SESSION

REAL-WORLD DATA MINING APPLICATIONS, CHALLENGES, AND PERSPECTIVES

Chair(s)

Drs. Mahmoud Abou-Nasr
Robert Stahlbock
Gary M. Weiss

Mechanical Property Classification of Vapor-Grown Carbon Nanofiber/Vinyl Ester Nanocomposites Using Support Vector Machines

O. AbuOmar[1,2], **S. Nouranian**[2], **R. King**[1,2], **T.M. Ricks**[3], **T.E. Lacy**[3]
[1]Department of Electrical and Computer Engineering, Mississippi State University, MS 39762, USA
[2]Center for Advanced Vehicular Systems (CAVS), Mississippi State University, MS 39762, USA
[3]Department of Aerospace Engineering, Mississippi State University, MS 39762, USA

Abstract — *In the context of materials informatics, the support vector machines technique was used to analyze and classify a large dataset of vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposites into three classes of desired mechanical properties, i.e., high storage modulus, high true ultimate strength, and high flexural modulus. Resubstitution and 3-folds repetitive cross validation techniques were implemented and the resulting classification information was compared and analyzed through sets of confusion matrices. This classification proves to be useful to materials designers and engineers, since a qualitative assessment of the expected nanocomposite mechanical response is given when suitable changes are made to the formulation, processing, and environmental conditions. This classification accelerates the lead time for the development of VGCNF/VE nanocomposites for a specific engineering application.*

Keywords: Support vector machines, materials informatics, vapor-grown carbon nanofiber/vinyl ester, confusion matrix, repetitive cross validation.

1 Introduction

The support vector machine (SVMs) technique [1] is considered one of the most widely used techniques in artificial intelligence community. This technique employs datasets of different sizes and dimensions and from different fields and domains. SVMs can be used for both supervised and unsupervised learning problems. Unsupervised learning ideally requires a large number of data vectors (points) within a particular dataset in order to adequately model a problem and avoid over-training (over-fitting). Supervised learning, however, can be utilized with a less number of data vectors, but some prior knowledge of the problem is needed in order to assist the SVMs model in generalizing and predicting the correct quantity given an unknown data vector [1]. SVMs can also assign linearly and nonlinearly separable data into two or more classes [1]. SVMs have recently been employed in the context of materials science and engineering to extend materials informatics [2-5]. This interdisciplinary study integrates computer and information science with other knowledge domains to facilitate knowledge discovery. For example, materials scientists can use materials informatics to interpret acquired experimental data through the use of SVMs and other machine learning approaches. It can also accelerate the research process and guide the development of new materials with desired mechanical properties. Materials informatics is being fueled by new and dynamic growth in the information technology sector and is driving the interest in SVMs, data mining, machine learning, information retrieval, and other knowledge representation or discovery schemes in the engineering disciplines [6]. AbuOmar, *et al*. [7] applied an artificial neural network (ANN) technique to a dataset associated with the viscoelastic response of a vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposite material system. The ANN was trained using the resubstitution method and the 3-fold repetitive cross validation (RCV) technique to provide a predictive model for these responses with minimal mean square error [7]. Roberts, *et al*. [8] presented a model that classifies different materials based on their microstructure. The core of the designed model is an SVMs classifier that identifies the appropriate class of given material sample based on microstructural characteristics such as Haralick variables, the Euler parameter, and the fractal dimension [8]. Swaddiwudhipong, *et al*. [9] utilized another important and efficient materials informatics technique, *i.e*., least squares support vector machines (LS-SVMs) [10]. Four LS-SVMs models simulated the relationship between the indentation load-displacement characteristics and elasto-plastic material properties, which are subject to the law of power hardening. No iterative approaches were used and it was found out that LS-SVMs approach was robust in determining the parameters in this relationship. Hu, *et al*. [11] used materials informatics to resolve the problem of materials science image data sharing. An ontology-based approach was employed to develop annotation for non-structured materials science data with the aid of semantic web technologies. Sabin, *et al*. [12] evaluated an alternative statistical Gaussian process model, which assumes a normal probability distribution over all of the training data and then interpolates to make predictions of microstructural evolution arising from static recrystallization in a non-uniform strain field. In this work, a specific class of advanced engineering materials was studied i.e., polymer nanocomposites [13]. These materials have multifunctional properties and are increasingly being used for aerospace, automotive, biomedical, fuel cell, catalysis, and other applications. For example, improved stiffness properties and energy

absorption characteristics are desired in automotive structural applications and nano-enhanced polymer composites meet these requirements [14]. They have been the subject of intensive research in recent years [15, 16]. AbuOmar, *et al*. [17] applied data mining and knowledge discovery techniques to a thermosetting VGCNF/VE nanocomposite material system [18-21] and include self-organizing maps (SOMs) [22, 23] and clustering techniques [24, 25]. The SOMs were used to recommend VGCNF/VE nanocomposite systems exhibiting the same storage and loss modulus responses in order to minimize the material preparation cost. A clustering technique (*i.e*., a fuzzy C-means algorithm) was also applied to discover any pattern in the nanocomposite behavior after using principal component analysis (PCA) as a dimensionality reduction technique [26].

This study seeks to expand the current knowledge of the influence of formulation, processing, and environmental factors on the mechanical behavior of VGCNF/VE nanocomposites by including a wider range of measured mechanical properties, *i.e*., viscoelastic property data [18], compressive and tensile property data, and flexural property data [27]. This new dataset provides a more general insight into the mechanical behavior of VGCNF/VE nanocomposites for data mining purposes. Application of data mining and knowledge discovery techniques to a comprehensive dataset of mechanical responses of polymer nanocomposites is unprecedented and novel. In the context of materials informatics, the results of this study serve as a guideline for materials scientists and engineers to efficiently design or optimize a material system for a certain application. The major contribution of this paper is to apply SVMs technique to separate VGCNF/VE nanocomposite test data into various desired mechanical property classes. As a result, an unknown VGNCF/VE specimen (*i.e*., a configuration not represented by the current dataset) can be easily characterized and classified into its corresponding VGCNF/VE class without the need to conduct expensive and time-consuming experiments. This quick qualitative assessment significantly reduces the lead time on developing a new material system for a desired application.

2 Materials and Methods

All data used in this work were generated using various statistical experimental designs, such as a general mixed level full factorial and central composite design, and are described in detail elsewhere [18-21, 27]. Different datasets were merged into a larger one incorporating 240 viscoelastic data points, 60 flexural data points, 172 compression data points, and 93 tension data points for variously formulated and processed VGCNF/VE nanocomposites. Therefore, the new larger dataset has a total of 565 data points. Each data point corresponds to combinations of nine input design factors and nine output responses. The input factors of the new VGCNF/VE dataset are curing environment (air vs. nitrogen), use or absence of a dispersing agent, strain rate, mixing method (ultrasonication, high-shear mixing, combination of both), VGCNF weight fraction, VGCNF and type (pristine vs. oxidized), high-shear mixing time, sonication time, and temperature. The output factors (*i.e*., measured properties) are true ultimate strength, true yield strength, engineering elastic modulus, engineering ultimate strength, flexural modulus, flexural strength, storage modulus, loss modulus, and tan delta. Therefore, the effectiveness of the SVMs technique implemented in this study is that materials scientists and engineers can select the optimal manufacturing combination of input factors that yield a desired mechanical property response; namely high storage modulus response, high true ultimate strength response, or high flexural modulus response. The choice of the optimal combination is based on several industrial measures, among which is the inputs' combination that has the minimum fabrication cost, the fastest or the most time-efficient combination, the combination that results in the best mechanical properties of the resulting VGCNF/VE nanocomposites, or a combination of two or more of these measures.

Different data interpolation techniques were used to replace some of the missing and unknown data fields in the new dataset [28]. These techniques include linear interpolation which is a method of curve-fitting using linear polynomials, and spline interpolation where the interpolant is a spline (piecewise polynomial). However, spline interpolation is more precise than regular polynomial interpolations because of its low interpolation error regardless of the polynomial degree used for the spline. In addition, spline interpolation avoids the problem of Runge's phenomenon, which occurs when using high degree polynomials for the interpolation process [28].

3 Theory/Calculation

As mentioned in Section 2, this study incorporates nine input and nine output design factors. Therefore, the dataset represents an eighteen-dimensional (18-D) analysis case. Since curing environment, use or absence of dispersing agent, mixing method, and VGCNF type are considered qualitative factors, they are represented by a numeric code for the analysis purposes. All quantitative values were normalized using standardized scores since the original value ranges were dissimilar.

Resubstitution and 3-fold repetitive cross validation techniques were used with the dataset to characterize the specimens that have desired VGCNF/VE properties. Each specimen was separated into an appropriate VGCNF/VE mechanical property class: specimens with high storage modulus (class 1), specimens with high true ultimate strength (class 2), and specimens with high flexural modulus (class 3). Before applying these techniques, a brief explanation of the SVMs operations, resubstitution, and repetitive cross validation techniques are introduced.

3.1 SVMs Operations

The goal of an SVMs classifier is to find a separating hyperplane between the points belonging to two distinct classes and maximize the distance between these points to the hyperplane. This maximum distance is referred to as the margin. This concept is illustrated in Figure 1 [1] for linearly separable data. For nonlinearly separable data, the

resulting hyperplane and margin has a complex, nonlinear form as shown in Figure 2 [1].

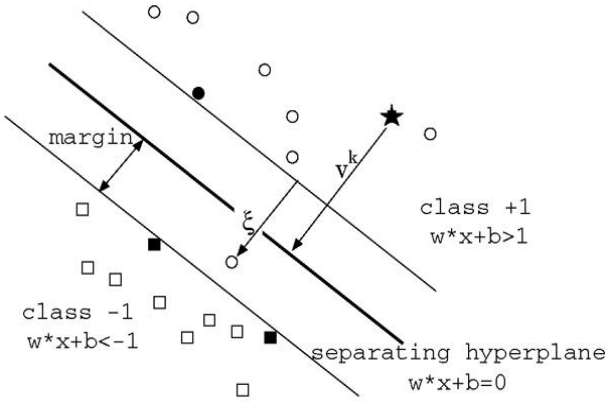


Figure 1. The SVMs model: the separating hyperplane along with the maximum margin for linearly separable data [1].

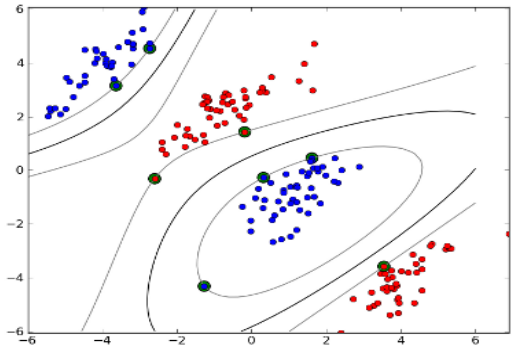


Figure 2. An example of the SVMs model for nonlinearly separable data [1].

3.2 Resubstitution method

The resubstitution method [29] is a computationally efficient technique in which the whole dataset is used to train the SVMs model, and the same dataset is used for testing (validation). This ensures that the SVMs model generalizes well when combinations of inputs and outputs are applied whose classes are not explicitly known. Good generalization is achieved when the apparent error (AE) is minimized [24].

The AE is defined as:

$$AE = \frac{1}{N}\sum_{i=1}^{N}\left|t_i - a_i\right| \qquad (1)$$

where N is the total number of samples, t_i is the targeted class of the sample in binary classification (i.e. 1 if the sample belongs to one class and 0 if it belongs to the other class), and a_i is the actual SVMs binary classification value (0 or 1).

Although several SVMs architectures and training algorithms are available, the SVMs classifier for two nonlinearly separable data is the most commonly used one and was utilized in this study [1]. However, since this study deals with separating the VGCNF/VE specimens into three different distinct property classes, the designed SVMs model was implemented in three stages using a one-against-all (OAA) strategy [30]. For example, in the first stage, specimens belonging to class 2 and class 3 were combined and compared against class 1. Finally, the classification information from these stages was combined in order to determine the three distinct property classes. This SVMs model assumed a non-linear relationship between the input-output variables and the corresponding class associated with each sample.

3.3 Repetitive cross validation technique

Repetitive cross validation (RCV) techniques can better train the SVMs model using available data. First, the available dataset is randomly partitioned into a training set and a test set. The training samples are further partitioned into two disjoint subsets: 1) the estimation subset, which is used to select the SVMs, and 2) the validation subset, which is used to test or validate the developed SVMs model [31]. In this way, the training samples can be used to assess the performance of various candidate SVMs models and thus the "best" one can be chosen [31]. Currently, there are four different RCV methods: holdout RCV, early-stopping method of training, multi-fold RCV, and leave-one-out RCV [32].

4 Results and discussion

The workflow of the classification process begins with applying the developed SVMs model to the VGCNF/VE data in which it was divided into training and test sets. Then two techniques were used for performance evaluation of the SVMs classifier; the resubstitution and the 3-folds RCV techniques. Finally, the results from both techniques were compared. In essence, the classifier ability to identify the percentage of test samples that belong to each of the desired mechanical properties (high storage modulus, high true ultimate strength, and high flexural modulus) was evaluated and analyzed.

In the SVMs analyses, classifications were compared and analyzed by using sets of confusion matrices (contingency tables) [33]. Additionally, a positive constant, *C*, was used to balance the margin size and the misclassification instances. The choice for *C* determines the number of support vectors and the overall performance of the SVMs model [1]. Three values of C were used in this study: 0.5, 10, and 100. Three kernel functions were also used in this study: a degree two polynomial, a dot product, and a hyperbolic tangent kernels.

For the resubstitution method, the SVMs model was generally able to correctly classify 100% of the VGCNF/VE specimens into three different distinct property classes for all kernel functions used in this study regardless of the constant *C*. Although a minor classification error (5%) resulted when the hyperbolic tangent kernel was used for class 3, this error was considered to be acceptable as it did not affect the overall classification accuracy of the model. These high classification rates are due to the fact that all samples were used for training and testing so the amount of misclassified information was minimal. For the 3-fold RCV technique, the chosen sizes of training and testing

sets were 80 and 40 data samples, respectively, for each of the three folds.

Following the standard practice of the SVMs analysis, the inputs and outputs were normalized using standardized scores, as their original value ranges were completely different from each other. The classification performance of the 3-folds RCV technique was inferior to that of the resubstitution method. Since the class 1, class 2, and class 3 sizes for the three folds in all stages were significantly lower than such classes when the resubstitution method was applied, some misclassification error is to be expected. Also, unlike the resubstitution method, the same samples were not used for training and testing in each fold resulting in some additional misclassification error. However, the resulted confusion matrices showed that the SVMs classifier performed well for fold 3 samples for all kernel functions at about 100% classification rate. In addition, reasonable classification rate was achieved for fold 2 when the hyperbolic tangent kernel was implemented at 75.00% and 58.33% was obtained for fold 2 samples when the polynomial kernel (degree 2) was implemented. The classification rates were lower for other cases. In addition, similar to the resubstitution method analyses, the classification results were independent of the value of the constant *C*. Another observation is that while 3-folds RCV technique was able to correctly classify specimens into class 1, mixed results were obtained when classifying specimens according to class 2 and class 3 and were observed to be dependent on the kernel function. For example, when a hyperbolic tangent kernel, the correct classification rate for class 3 was observed to be 81.67%. This value dropped to 33.33% when the degree two polynomial kernel was used. On average, the classification performance was the best when hyperbolic tangent kernel was implemented yielding a classification rate of 76.81%.

Table 1. Classification information of the SVMs model when a polynomial kernel of degree 2 was implemented using both the resubstitution and 3-fold RCV methods.

	Resubstitution method				**3-fold RCV method**			
	Class 1	Class 2	Class 3	Average	Class 1	Class 2	Class 3	Average
Correct Classification Rate	100%	100%	100%	100%	100%	75.00%	33.33%	69.44%
Apparent Error Rate	0%	0%	0%	0%	0%	25.00%	66.67%	30.56%

Table 2. Classification information of the SVMs model when a dot product kernel was implemented using both the resubstitution and 3-fold RCV methods.

	Resubstitution method				**3-fold RCV method**			
	Class 1	Class 2	Class 3	Average	Class 1	Class 2	Class 3	Average
Correct Classification Rate	100%	100%	100%	100%	100%	50.00%	47.50%	65.83%
Apparent Error Rate	0%	0%	0%	0%	0%	50.00%	52.50%	34.17%

Table 3. Classification information of the SVMs model when a hyperbolic tangent kernel was implemented using both the resubstitution and 3-folds RCV methods.

	Resubstitution method				**3-fold RCV method**			
	Class 1	Class 2	Class 3	Average	Class 1	Class 2	Class 3	Average
Correct Classification Rate	100%	100%	95.00%	98.33%	98.75%	50.00%	81.67%	76.81%
Apparent Error Rate	0%	0%	5.00%	1.67%	1.25%	50.00%	18.33%	23.19%

The overall classification rates and apparent error rates for the three different kernels using both the resubstitution and 3-fold RCV methods are shown in Tables 1-3. Based on these results, the SVMs model was able to more correctly classify specimens belonging to class 1 than class 2 and 3. Additionally, the resubstitution method was determined to be superior to the 3-fold RCV method for this specific problem. For example, when the resubstitution method was implemented using the hyperbolic tangent kernel, the SVMs model was able to identify all samples (100%) that have the highest storage modulus responses and all samples that have the highest true ultimate strength responses where it was able to identify 95% of samples that have the highest flexural modulus responses (Table 3). When the 3-fold RCV was implemented, the SVMs model was able to identify 98.75% of test samples that have the highest storage modulus responses, 50% of test samples that have the highest true ultimate strength responses, and 81.67% of test samples that have the highest flexural modulus responses (Table 3).

In addition, by choosing particular inputs' combination based on one of the industrial optimal measures mentioned in section 2, this SVMs model is able to identify the desired mechanical property (one of the three desired mechanical response classes of high storage modulus, high true ultimate strength, or high flexural modulus) that will be resulted from this combination based on the selected industrial measure. Section 5 will

elaborate more on the effectiveness of the developed SVMs model on the VGCNF industrial manufacturing process.

5 Application to Materials Informatics

Since this dataset encompasses a wide variety of mechanical testing methods, conditions, and material configurations, the resulting SVMs model can be used to effectively predict the mechanical response for previously untested material configurations. Such a capability reduces the need to perform further experiments and allows the materials scientists to quickly assess the viability of a new material configuration. For example, if a high storage modulus is desired, the optimal VGCNF weight fraction can be determined for given mixing conditions which are likely not located at one of the tested levels. Additionally, material and processing costs can likely be reduced by using material informatics principles. Since VGCNFs are often expensive and nanocomposite fabrication processes often take several hours, a range of VGCNF weight fractions and mixing times can be established over which adequate properties are obtained. A smaller amount of VGCNFs combined with shorter mixing times could ultimately reduce production costs by a significant amount.

6 Summary and Conclusions

A support vector machines (SVMs) technique was applied to a vapor-grown carbon nanofiber (VGCNF)/vinyl ester (VE) nanocomposite dataset as a proof of concept for materials informatics. This dataset consists of 565 different design points: 172 compression, 93 tension, 60 flexure, and 240 viscoelastic points. Each treatment combination consisted of eighteen feature dimensions corresponding to the nine input and nine output design factors. The nine input factors of the VGCNF/VE dataset were curing environment (air vs. nitrogen), use or absence of a dispersing agent, strain rate, mixing method (ultrasonication, high-shear mixing, and combination of both), VGCNF weight fraction, VGCNF type (pristine vs. oxidized), high-shear mixing time, sonication time, and temperature. The output factors (*i.e*., measured properties) were true ultimate strength, true yield strength, engineering elastic modulus, engineering ultimate strength, flexural modulus, flexural strength, storage modulus, loss modulus, and tan delta. The SVMs model was trained using the resubstitution method and the 3-fold repetitive cross validation (RCV) technique to classify each VGCNF/VE sample into one of three optimal property classes: high storage modulus, high true ultimate strength, high flexural modulus. The classifier was implemented in three stages using a one-against-all strategy. Three possible kernel functions were explored in this study: a polynomial kernel of degree two, a dot product kernel, and a hyperbolic tangent kernel. A set of confusion matrices was used to compare the different analysis methods.

In general, the SVMs model using the resubstitution method was able to predict the optimal property classes with a minimal apparent error (AE) rate irrespective of the kernel function that was used. While the SVMs model using the 3-fold RCV method was able to accurately predict which data points belonged to the high storage modulus class, in general, the SVMs model using this method had significant FARs.

Most importantly, the developed SVMs model is able to identify the desired mechanical property response value (high storage modulus, high true ultimate strength, or high flexural modulus) resulted from a chosen untested combination of the nine input factors mentioned in this study. The choice of the inputs' combinations commensurate with particular optimal industrial measure(s) selected by materials scientists and engineers. This includes but is not limited to, the inputs' combination that has the minimum industrial fabrication cost, the combination that yields the fastest manufacturing process, the combination that results in the optimal mechanical properties of the resulting VGCNF/VE polymer nanocomposites, or any two or more of these measures combined. In other words, if an inputs combination whose outputs responses are unknown is given to the developed SVMs model, then the desired mechanical property response will be easily retrieved based on this combination.

The model's ability to identify these desired mechanical property responses based on particular combination of input factors will result in faster VGCNF nanocomposites manufacturing lead time without the need to rely on intensive and time-consuming experiments. In addition, while this model only considers three classes, it can also be readily extended to include additional desirable material properties. This issue is the focus of ongoing research.

The SVMs classifier applied in this study demonstrates the usefulness of data mining and knowledge discovery techniques in materials science and engineering. It is expected that more such techniques will be employed within the rising field of materials informatics in near future.

References

[1] S. Theodoridis, K. Koutroumbas, Pattern Recognition, fourth ed., Academic Press, Massachusetts, 2008.

[2] K. Rajan "Materials informatics," *Materials Today*. Vol. 8, pp. 38-45, 2005.

[3] K. F. Ferris, L. M. Peurrung, J. M. Marder "Materials informatics: Fast track to new materials," *Advanced Materials and Processes*, Vol. 165, pp. 50-51, 2007.

[4] C. Suh, K. Rajan, B. M. Vogel, B. Narasimhan, S. K. Mallapragada "Informatics methods for combinatorial materials science," *Combinatorial Materials Science*, Vol. 5, pp. 109-119, 2006.

[5] Q. Song "A preliminary investigation on materials informatics," *Chinese Science Bulletin*, Vol. 49, pp. 210-214, 2004.

[6] J. Rodgers and D. Cebon (2006). Materials Informatics. *MRS Bulletin*, 31, pp 975-980. doi:10.1557/mrs2006.223.

[7] AbuOmar, O., Nouranian, S., & King, R. (2013). Artificial Neural Network Modeling of the Viscoelastic Properties of Vapor-Grown Carbon Nanofiber/Vinyl Ester Nanocomposites. *The 19th International Conference on Composite Materials (ICCM19)*, July 28-August 2, 2013. Montreal, Quebec, Canada.

[8] Roberts, K, Muchlich, F, Schenkel, R, Weikum, G, An Information System for Material Microstructures. *International Conference on Scientific and Statistical Database Management (SSDBM'04)*. 1099-3371/04, 2004 IEEE

[9] Swaddiwudhipong, S, Tho, K K, Liu, Z S, Hua, J, Ooi, N S B. Material Characterization via Least Squares Support Vector Machines. *Modelling Simul. Mater. Sci. Eng*. 13 (2005) 993-1004.

[10] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. DeMoor, J. Vandewalle, Least Squares Support Vector Machines, World Scientific, 2002 (ISBN 981-238-151-1)

[11] C. Hu, C. Ouyang, J. Wu, X. Zhang, C. Zhao "NON-structured materials science data sharing based on semantic annotation," *Data Science*. Vol. 8, pp. 52-61, 2009.

[12] T. Sabin, C. Bailer-Jones, P. Withers "Accelerated learning using Gaussian process models to predict static recrystallization in an Al-Mg alloy," *Modelling and Simulation in Materials Science and Engineering*, Vol. 8, pp. 687-706, 2000.

[13] J. H. Koo "*Polymer nanocomposites: Processing, characterization, and applications*," 1st edition, McGraw-Hill, New York, 2006.

[14] J. Garces, D. J. Moll, J. Bicerano, R. Fibiger, D. G. McLeod "Polymeric nanocomposites for automotive applications," *Advanced Materials*, Vol. 12, pp. 1835-1839, 2000.

[15] F. Hussain, M. Hojjati, M. Okamoto, R.E. Gorga, Review article: polymer-matrix nanocomposites, processing, manufacturing, and application: An overview, J. Compos. Mater., 40 (2006) 1511-1575.

[16] E.T. Thostenson, C. Li, T.W. Chou, Nanocomposites in context, Compos. Sci. Technol., 65 (2005) 491-516.

[17] O. Abuomar, S. Nouranian, R. King, J.L. Bouvard, H. Toghiani, T. E. Lacy, C. U. Pittman Jr "Data mining and knowledge discovery in materials science and engineering: A polymer nanocomposites case study," *Advanced Engineering Informatics* 27(4), 615–624, 2013. DOI:10.1016/j.aei.2013.08.002.

[18] S. Nouranian, Vapor-grown carbon nanofiber/vinyl ester nanocomposites: Designed experimental study of mechanical properties and molecular dynamics simulations, Mississippi State University, PhD Dissertation, Mississippi State, MS USA, 2011.

[19] S. Nouranian, H. Toghiani, T.E. Lacy, C.U. Pittman, J. Dubien, Dynamic mechanical analysis and optimization of vapor-grown carbon nanofiber/vinyl ester nanocomposites using design of experiments, J. Compos. Mater., 45 (2011) 1647-1657.

[20] S. Nouranian, T.E. Lacy, H. Toghiani, C.U. Pittman Jr, J.L. Dubien, Response Surface Predictions of the Viscoelastic Properties of Vapor-Grown Carbon Nanofiber/Vinyl Ester Nanocomposites, J. Appl. Polym. Sci., (2013) DOI: 10.1002/app.39041.

[21] G.W. Torres, S. Nouranian, T.E. Lacy, H. Toghiani, C.U. Pittman Jr, J. Dubien, Statistical Characterization of the Impact Strengths of Vapor-Grown Carbon Nanofiber/Vinyl Ester Nanocomposites Using a Central Composite Design, J. Appl. Polym. Sci., 128 (2013) 1070-1080.

[22] R.L. King, A. Rosenberger, L. Kanda, Artificial neural networks and three-dimensional digital morphology: a pilot study, Folia Primatol., 76 (2005) 303-324.

[23] T. Kohonen, Self-organization and associative memory, Springer-Verlag, 1988.

[24] S. Miyamoto, H. Ichihashi, K. Honda, Algorithms for Fuzzy Clustering: Methods in c-Means Clustering with Applications, Springer, 2008.

[25] J.C. Bezdek, R. Ehrlich, FCM: The fuzzy c-means clustering algorithm, Computers & Geosciences, 10 (1984) 191-203.

[26] I.T. Jolliffe, Principal Component Analysis, Springer, 2002.

[27] J. Lee, S. Nouranian, G.W. Torres, T. E. Lacy, H. Toghiani, C. U. Pittman, J. L. DuBien, Characterization, prediction, and optimization of flexural properties of vapor-grown carbon nanofiber/vinyl ester nanocomposites by response surface modeling. J. Appl. Polym. Sci., 130 (2013) 2087–2099. doi: 10.1002/app.39380

[28] MATLAB Mathematics and Interpolation, Release 2012a, The MathWorks, Inc., Natick, Massachusetts, United States.

[29] J. Twomey, A. Smith "Bias and variance of validation methods for function approximation neural networks under conditions of sparse data," *IEEE Transactions on Systems, Man., and Cybernetics*, Vol. 28, pp. 417-430, 1998.

[30] J. Milgram, M. Cheriet, and R. Sabourin, "One Against One" or "One Against All": Which One is Better for Handwriting Recognition with SVMs?." *Tenth International Workshop on Frontiers in Handwriting Recognition*. 2006.

[31] S. Haykin, Neural networks and learning machines, 3rd edition, Prentice-Hall, 2009.

[32] S. Hayken, Neural networks: A comprehensive foundation, 2nd edition, Englewood Cliffs (NJ): Prentice-Hall, 1999.

[33] R. Kovari, F. Provost, Glossary of terms, *Machine Learning*, 30 (1998), No. 2/3, 271-274.

APPES Maps as Tools for Quantifying Performance of Truck Drivers

Iulian Carpatorea and Slawomir Nowaczyk and Thorsteinn Rognvaldsson and Marcus Elmer

Abstract— **Understanding and quantifying drivers' influence on fuel consumption is an important and challenging problem. A number of commonly used approaches are based on collection of *Accelerator Pedal Position - Engine Speed* (APPES) maps. Up until now, however, most publicly available results are based on limited amounts of data collected in experiments performed under well-controlled conditions. Before APPES maps can be considered a reliable solution, there is a need to evaluate the usefulness of those models on a larger and more representative data.**

In this paper we present analysis of APPES maps that were collected, under actual operating conditions, on more than 1200 trips performed by a fleet of 5 Volvo trucks owned by a commercial transporter in Europe. We use Gaussian Mixture Models to identify areas of those maps that correspond to different types of driver behaviour, and investigate how the parameters of those models relate to variables of interest such as vehicle weight or fuel consumption.

I. Introduction

Road transportation is one of the ways most often used to move goods and people from one point to another in Europe. According to [1], almost 75% of all the cargo transported in 2011 in Europe was done with trucks, summing to approximately 520 billion tonne-kilometres. As a consequence the fuel burned by vehicles accounts for around 20% of the CO_2 emissions in the region.

In addition, fuel expense is one of the most important cost factors, accounting for approximately 30% of the total operating expenses of a heavy duty vehicle.Fuel efficiency is very important for modern vehicles for environmental as well as financial reasons.

There are many factors influencing fuel consumption of a vehicle. Some of them the transporters and vehicle manufacturers have little or no control over, such as weather or road topology. However, there are many others that they can affect, for example route planning, aerodynamics or tires. Furthermore, a factor that has been widely recognised as being very important is the driver.

One asset that is recently becoming available are large quantities of data collected over long time under real driving conditions. This data can generally be made available and processes either on-board or off-board the vehicle. Each of these two options have their own advantages and disadvantages. Storage and analysis of high frequency data consisting of hundreds of signals requires large amounts of memory and computation power, neither of which is commonly available or feasible to implement on commercial trucks. On the other hand, data transmission costs are currently too high to justify a fully off-board solution. Therefore, a promising approach is to investigate data representation abstractions that can be easily computed and stored on-board, but which also contain enough information to provide valuable knowledge when analysed off-board.

In order to find opportunities for reducing fuel consumption there is a need to analyse this newly available data. This paper is one step in this direction.

A. Related Work

Fuel consumption for heavy duty vehicles is an issue that affects our daily lives. The recently adopted Euro VI standard exemplifies the importance of fuel consumption reduction, as it directly affects particle emissions. The Euro VI standard enforces reduction of noxious emissions by 66% and of nitrogen oxide emissions (NOx) by 80%, compared to Euro V.

Liimatainen [2] has demonstrated how to use fuel consumption as an incentive for drivers to increase their fuel efficiency. He also points out that that it is difficult to take external factors into account when assessing drivers' performance. Ting et al [3] have, in a simulation study, shown the importance of driver and address the issue of driver fatigue and its effects on driving capabilities. Another study of driver performance and its classification is done in [4].

Rafael-Morales and de Gortar [5] conducted an extensive field study to determine the effects of so called "technical driving" for reducing fuel consumption. Authors investigate several means of using vehicle in an efficient manner looking from various perspectives, including analysis of relation between torque and engine speed.

In this paper we expand on the work of of Guo et al [6], who investigate using *Accelerator Pedal Position - Engine Speed* (APPES) map for evaluating drivers' performance. Guo et al identify different regions of this map that correspond to higher or lower fuel consumption. The weakest point of their work is the limited variety of data used to validate the method: only one vehicle was driven on the same route by the same driver. We show that APPES maps can be correlated with a number of relevant factors not only under controlled experimental conditions, but also during actual commercial operation.

II. Data

We use two large datasets that have been collected in research and development projects within Volvo Group Trucks

Iulian Carpatorea, Slawomir Nowaczyk and Thorsteinn Rognvaldsson are with Center for Applied Intelligent Systems Research, Halmstad University, Sweden (email: firstname.lastname@hh.se).

Marcus Elmer is with Volvo Group Trucks Technology, Advanced Technology & Research, Göteborg, Sweden (email: marcus.elmer@volvo.com).

Technology(VGTT). The first dataset comes from *European Field Operational Test* (EuroFOT) project [7], in which VGTT was a partner with the role of testing *Fuel Efficiency Advisor* functionality. The other is an internal Volvo project called *Customer Fuel Follow-up* (CuFF). In both projects, data from multiple trucks have been collected, covering a wide area in Europe and also spanning over a long period of time, offering a variety of both geographic and ambient conditions.

The subset of data that we base our results on consists of over 1200 trips, performed by five Volvo trucks. Each truck has an automatic gearbox with 12 gears and Cruise Control system. Each trip contains over one hundred signals that are logged from the vehicles' internal Controller Area Network(CAN) as well as additional sensors, at 10 Hz sampling frequency.

The complete database amounts to approximately 100 TB of data. Even with this large amount of information, we do not have complete knowledge of all the relevant circumstances. Among the most important factors that we are missing are traffic and weather conditions.

III. Methodology

The first step in quantifying a driver's behaviour is finding a good representation of it, one that is simple enough to reason about, but at the same captures all the important aspects. In this work we have decided to focus on *Accelerator Pedal Position - Engine Speed* (APPES) maps. It is a way to describe truck usage information that is commonly employed both by automotive engineers and in driver training, as those maps are easy to understand and have very intuitive interpretations.An example of APPES map for a single trip is presented in Figure 1.

In this work we focus on analysis of driver performance. Therefore, we are only considering the time where the cruise control has been disabled. The surface represents how much time has been spent in various combinations of accelerator pedal position and engine speed during a trip. The first of those signals can be thought of as the request from the driver, while the second as an overall response of the vehicle. Most important aspects of truck operation are directly reflected in this map. For example circumstances such as road gradient, engine power or vehicle weight all affect how fast the speed will be changing when the accelerator pedal is pressed to a given level.

Another benefit of APPES maps is that they can be easily obtained from commercial vehicles, for example using telematics technology. Computing them on-board is a smalll effort, and they can be efficiently stored in existing control units to be periodically transmitted, either via wireless networks or during garage visits.

In the past, APPES maps have been used to evaluate drivers, but mostly in controlled experiments, under well-known conditions. Their potential and usefulness in a realistic setting have not been fully explored yet. For example, in [6] the authors focus on identifying regions of the APPES map that can be correlated to fuel consumption. They show

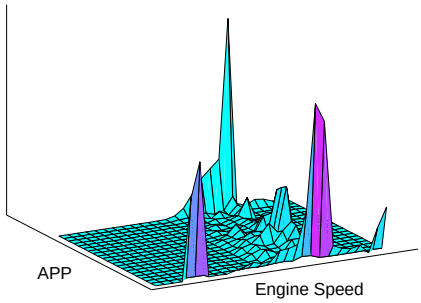


Fig. 1. APPES map for a single trip of heavy vehicle.

increased vehicle efficiency through better driving, but their results are based on a single truck being driven on a pre-specified route.

The contribution of our work is to demonstrate the usefulness of APPES maps in realistic scenarios, when the external conditions of driving vary in unpredictable ways. Actual situations on the road will put many constraints on what is possible for the driver to do, but we are interested in finding similarities and differences between different driving styles. We can use this to communicate to drivers information regarding their driving behaviour, for example give them advice on how to achieve better fuel efficiency.

One important aspect of vehicle operation that is not explicit in the APPES map is the gearbox. Different gears should be used at different speeds in order to maximise fuel efficiency. Modern truck engines always generate power by burning fuel, but some of this power is lost, as heat. The lost amount, however, varies depending on parameters such as torque and engine speed — each engine manufacturer designs their products with a specific *efficiency map* in mind. By changing gears, driver can optimise power output for any desired vehicle speed, by controlling the ratio between the engine and wheel speeds. This power is then used to overcome resisting forces, gravity and to maintain or increase velocity.

Figure 2 shows the APPES map for all the trips we are considering in this paper. We have identified four important regions in this distribution. We refer to them as *Neutral*, *Free Roll*, *Driving* and *Full Throttle*. Those names indicate the intuitive interpretation of the physical behaviour of the truck that corresponds to each of those regions in the APPES map.

The region we call *Neutral* corresponds to low engine speed, with the exact value depending on engine specification, and acceleration pedal being fully released. It can be seen in Figure 2 as the peak in the lower left corner. Those conditions can be achieved when the truck is in neutral gear, but it can be either stationary or moving.

The *Free Roll* region is characterised by the accelerator pedal being fully released, but engine is rotating above idle speed. This generally means that the vehicle is in gear other than neutral and that it is moving forward. In this mode

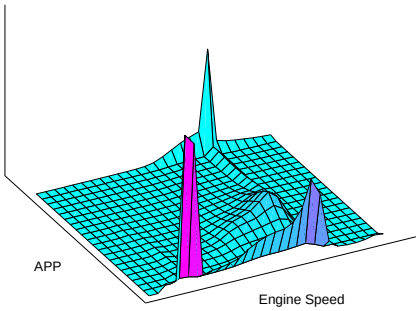


Fig. 2. Combined APPES map for all the trips.

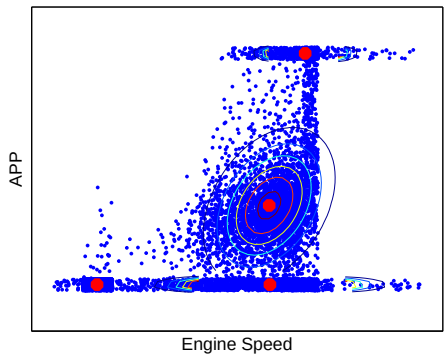


Fig. 3. Gaussian Mixture Model, with blue points corresponding to the data and red points corresponding to locations of four Gaussian means representing different regions. The two lower are Neutral and Free Roll. The top is Full Throttle. The middle one is Driving.

the engine is not using any fuel. This situation most often happens when travelling downhill, where the driver can use gravity to maintain the desired speed, or when they anticipate that a speed reduction will be needed in the near future, and use kinetic energy to propel the vehicle forward for a short time. In Figure 2, this region corresponds to the ridge along the bottom.

Full Throttle corresponds to the accelerator pedal being fully pressed. It can either indicate that the driver is attempting to reach the desired speed in the minimum amount of time or trying to maintain the highest speed allowed by the electronic limiter, which is set to 90 km/h. This region can be seen as the peak near the top right corner of Figure 2.

Finally, the *Driving* region captures the full spectrum of driver behaviour in between the three aforementioned extremes. It covers engine speeds that correspond to driving using different gears and at different velocities, as well as how much the accelerator pedal is pushed, either for acceleration or for compensating for road conditions such as hills. This region is probably the most interesting one, since the exact distribution of data within it can tell us, for example, when this particular driver changes gears, possibly leading to creation of driver profiles that can later be used, with long term observation, to track driver performance. In Figure 2, this region can be seen as the triangle-like shape in the middle of the plot.

Based on this rough classification, we have decided to model the data using a Gaussian Mixture Model. Figure 3 shows the obtained result. Blue dots correspond to the data that was extracted from the APPES map using uniform random sampling within each map cell. The four red dots denote the location of means of the four Gaussians that were fitted to this data. As can be seen, they correspond quite well to the four regions we identified above. The coloured ellipses around each mean visualise the covariance matrix of each model.

In the following section we will present the results of comparing the parameters of those Gaussians across different trips.

IV. Results

In this section we report the results of our experiments concerning the usefulness of APPES maps in a real setting, especially with respect to fuel consumption. We base our analysis on comparing parameters of the Gaussians Mixture Models corresponding to the four types of driving that we have introduced in the previous section. There are three important comments regarding our methodology that need to be explained before we present actual results.

First of all, the goal is to relate various factors of interest to the APPES regions that we have identified as the most important ones. In order to do that, it is useful to categorise those factors as either *cause* or *effect*, in particular from the driver's point of view. For example, traffic density requires drivers to change their desired behaviour and affects their decisions. In this paper we analyse a factor from each category, choosing ones that are easy to understand and using available expert knowledge. As a *cause* type we have selected gross vehicle weight, while as an *effect* type, fuel consumption. We have decided upon those two because it is interesting to analyse how weight affects driving style, while at the same time we are very much aware that it also heavily influences vehicle fuel usage.

Second, since our data comes from real commercial trips, we do not have access to any form of ground truth concerning actual performance level of drivers. Therefore, we are interested in a finding as few parameters describing each APPES map as possible, preferably ones that are easy to visualise and whose correlation to variables of interest can be discussed in this text. To this end, we have chosen to only take into consideration the proportion that each Gaussian occupies in the complete model, ignoring both the exact location of the mean and the covariance matrix.

Finally, many of the individual trips did not have enough data to build reliable Gaussian Mixture Models with all four regions of interest properly represented. Individual drivers rarely cover all of them in one trip, which is a direct effect of the length of those trips as well as high usage of cruise control. This often caused undesired effects of Gaussian

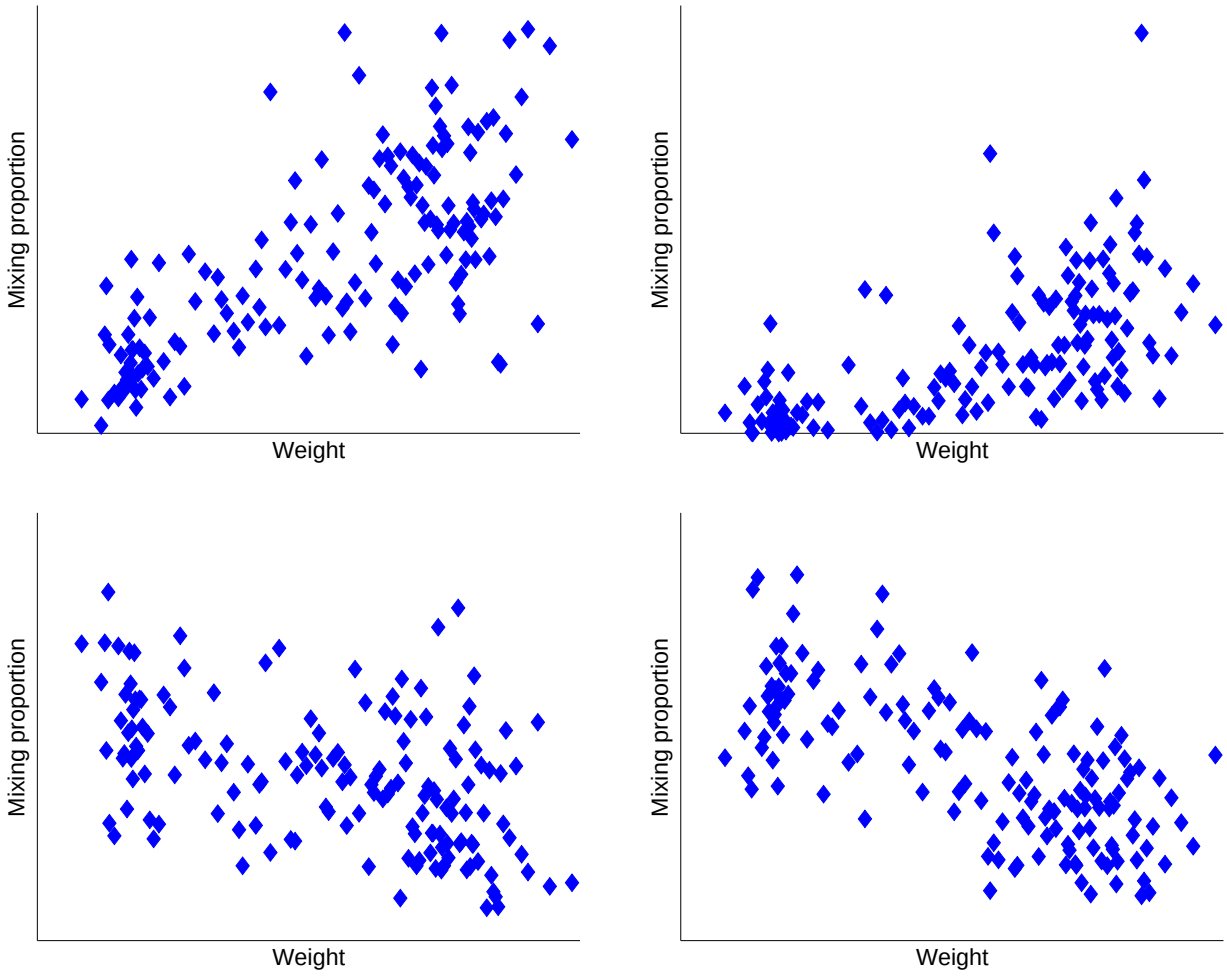


Fig. 4. Mixing proportions of *Free Roll* (top left), *Full Throttle* (top right), *Neutral* (lower left) and *Driving* Gaussians for groups of 8 trips based on weight.

means moving to different locations.

Those situations are not particularly interesting, since we believe they are usually caused by specifics of the individual mission. On the other hand, we do not want to ignore such data but rather evaluate the robustness of APPES maps. However, in order to evaluate it we need to perform experiments, e.g. test results with and without outliers. The immediate goal is to determine usefulness of APPES maps. In order to analyse how each factor correlates to selected regions of interest, we require all four of them to be present. Therefore, we did not fit Gaussian Mixture Models to individual trips, but to groups of several similar trips. Grouping of trips reduces the chance that any of the Gaussians is missing.

A. *Vehicle Weight*

Vehicle weight is a factor that is known to be very important for fuel consumption. However, there are no large scale, systematic studies of how a driver's behaviour is affected by vehicle weight. Therefore, in this section we present results that can be seen as a starting point towards this goal.

As explained before, APPES maps show a driver's behaviour for a given trip or a group of trips. We can describe each map by four numbers, the mixing proportions of the Gaussian models corresponding to each type of driving: *Free Roll*, *Neutral*, *Full Throttle* and *Driving*. Those proportions correspond to the amount of time that was spent in each of the regions. In order to analyse the relation between driver's performance and vehicle weight, we want to observe changes that happen when driving trucks with different loads. Therefore, we group similar trips and plot the mixing proportions of each APPES Gaussian, looking for interesting relations.

The clearest correlation, with value of 0.7, can be seen in Figure 4, top left. Each point on this plot corresponds to a group of 8 trips, with similar vehicle weights. The Y value represents the mixing proportion of the *Free Roll* Gaussian, while the X value represents the average weight within the group. As can be seen, the lightest vehicles very rarely use free roll, while the share of this driving type generally increases as the vehicle becomes heavier.

This is an anticipated result, since heavier vehicles have

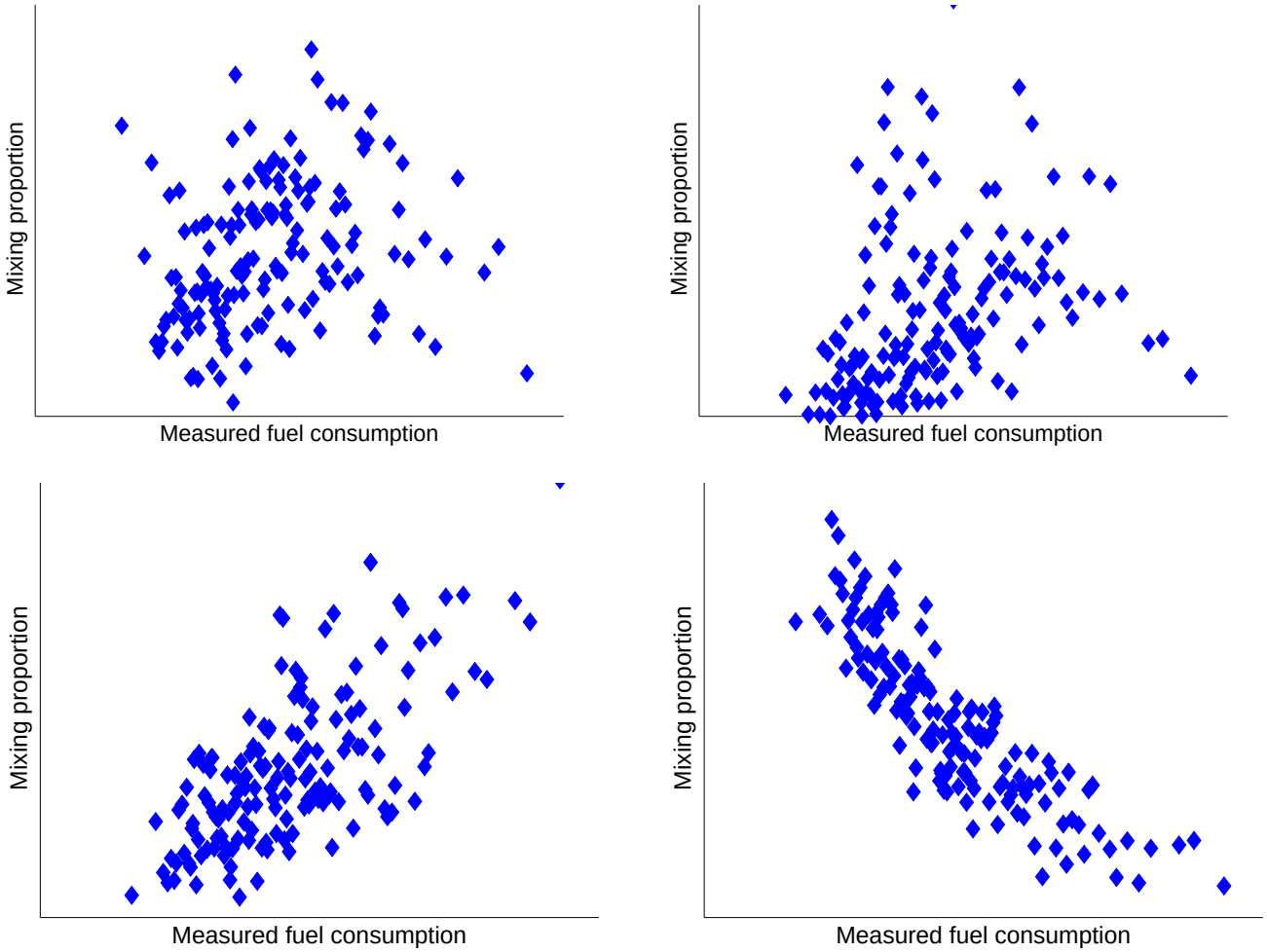


Fig. 5. Mixing proportions of *Free Roll* (top left), *Full Throttle* (top right), *Neutral* (lower left) and *Driving* Gaussians for groups of 8 trips, with respect to measured fuel consumption.

more inertia and therefore lose speed less rapidly. However, free rolling with a vehicle does not only depend on vehicle weight but also on a skill of the driver, mainly their ability to anticipate future situations and use ambient conditions to better use the vehicle. One circumstance that a good driver takes advantage of is coasting in gear. In such a case, if the road gradient is enough to keep the vehicle at desired speed, a *Free Roll* can reduce the fuel usage to zero.

However, the APPES map itself does not contain enough information to identify such situations, and finding them is a topic for future work. An anticipation situation is more often encountered in towns or in high traffic scenarios, where change of speed is more common. For example, driving towards a stop light and anticipating having red light upon arrival might prompt the driver to stop accelerating and let the vehicle roll. This situation could result, depending on the gear, in either *Free Roll* or *Neutral* behaviour.

Another aspect to consider is that heavier vehicles, having more inertia, also offer more opportunities to use free roll. For example, in anticipation of a steep downhill, the driver can choose to *Free Roll* for some time, on the flat road, and lose some speed which can then be regained during the downhill section.

Another aspect to consider for this Gaussian is what happens after the *Free Roll* ends. This, again, is not actually included in the APPES map, but *Free Roll* followed by, e.g., high acceleration is often an indication of insufficient anticipation. There are many possibilities and analysing them will reveal further information regarding drivers' behaviour in various scenarios.

Continuing to Figure 4, top right, the *Full Throttle* Gaussian, we can also notice a high correlation, at 0.62. This relation can be explained by the fact that engine power is a limiting factor for heavier trucks, and desired acceleration can often only be achieved by pressing the pedal fully. It could also be attributed to the fact that drivers are less inclined to try and optimise fuel usage for heavy loads, since in most cases their performance is measured in absolute terms, and therefore they know they cannot compete with lighter trucks.

If validated, this could be a strong argument towards introducing performance indicators that analyse fuel consumption and also consider vehicle and ambient conditions. As all trucks involved in this analysis are the same model, it is also

possible that lighter trucks do not require full power fron the engine, especially if there is an acceleration level whicl generally satisfies the drivers. Heavier trucks may never reacl said acceleration, even with full throttle, leading to highe amount of time spent in that region.

Correlation for Figure 4, bottom left, the *Neutral* region has the value of -0.47. Weight seems to have a mino influence, if any at all, for time spent in neutral gear. On possible explanation for slightly higher average values fo lighter trucks is lower inertia. Instead of using *Free Roll* where engine braking occurs, lighter trucks choose to us *Neutral*. This way they the keep vehicle rolling longer.

Figure 4, bottom right, *Driving*, shows a stronger negativ correlation of -0.66. One explanation can be the opposit of *Full Throttle*: since drivers rarely need to use full throttl to maintain the desired vehicle speed profile, they end up i the *Driving* region more often. However, it is important t remember that the mixing proportions for all the Gaussian always sum up to 1. Looking at Figure 4, top left and top right respectively, we notice that the time spent in those regions is very small, which means that it has to be distributed among the remaining two regions.

Finally, the four regions can be also seen as two groups of complementary driving styles. We can consider the first to be comprised of *Full Throttle* and *Driving* Gaussians, where light vehicles tend to spend more time in the *Driving* region while heavy vehicles end up in *Full Throttle* more often. The second group is formed by *Neutral* and *Free Roll*. Lower inertia and engine braking being relatively stronger make light vehicles use neutral gear more often, while heavier vehicles do not require it so much.

B. Fuel Consumption

As mentioned earlier, we consider fuel consumption to be an effect of driving style. Therefore, observing how different driving styles influence fuel consumption can be directly used to assess performance of drivers. We can estimate how beneficial it is to be in one region or another, ignoring other factors. However, the degree to which positioning oneself in the APPES map is up to a driver is unclear, as there are many conditions that constrain their decisions.

One of the surprising results is that there is no clear correlation, at a value of 0.22, between *Free Roll* and fuel consumption, as seen in Figure 5, top left. Expert knowledge tells us that if a vehicle is in gear and the acceleration pedal is released, the engine will not use any fuel. This has been verified directly with real world data and it strongly suggests that *Free Roll* is highly desirable behaviour.

One possible explanation could be that the beneficial effects are clouded by detrimental effects of other regions. Another would be that other factors, for example heavy traffic or hilly terrain, heavily influence when a driver can choose to *Free Roll*, and their own detrimental effect can outweigh the benefits. Further investigation is required before a more concrete conclusion can be reached, as this observation contradicts prior beliefs. However, it motivates our original

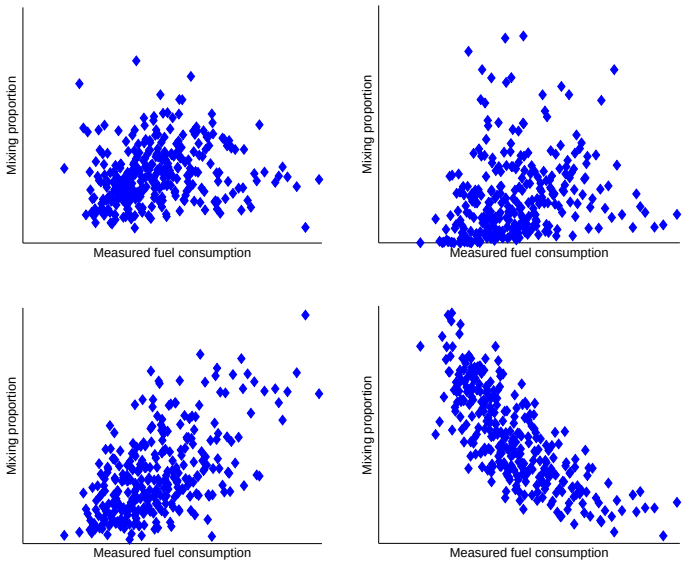


Fig. 6. Mixing proportions of *Free Roll* (top left), *Full Throttle* (top right), *Neutral* (lower left) and *Driving* Gaussians for groups of 4 trips.

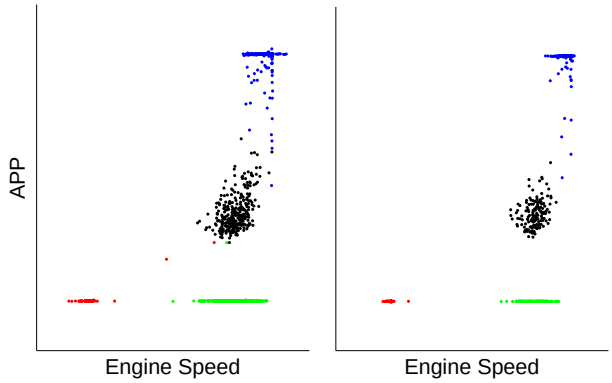


Fig. 7. Gaussian means for groups of 4 trips(left) and 8 trips(right)

thesis that APPES maps, due to their simplicity, can uncover interesting relations in the data.

It is also interesting to note that this region has higher correlation with weight, as depicted in previous subsection, Figure 4, top left. Consequently, further investigation should focus on how both weight and fuel consumption influence each Gaussian.

Figure 5, top right, corresponding to the *Full Throttle* region, shows 0.32 correlation. Even despite it being so low we can notice that there is a tendency for low fuel consumption to be associated with low *Full Throttle* proportion. One way of explaining this is by taking into account a cause, such as the previously discussed vehicle weight. We noticed a much higher usage of full throttle for heavier vehicles, which in turn can be associated with higher fuel consumption. This suggests that APPES maps can also be used to link relations between various factors.

On the other hand, a strong positive correlation of 0.68 can be seen in Figure 5, bottom left, for *Neutral* mixing proportion. This agrees with our expectations, since we measure fuel consumption in $L/100km$ and having a vehicle in neutral gear burns fuel but does not propel the truck forward. However, some drivers may choose to use *Neutral* instead of *Free Roll* under certain circumstances, as it means is no engine braking.

The final Gaussian, *Driving*, shown in Figure 5, bottom right, exhibits a very strong negative correlation of -0.88. We conclude that adapting the fuel demand as well as keeping an appropriate speed, both of whom can be done by active driving, uses the fuel most efficiently.

For comparison, Figure 6 presents the same relations as the previous four figures, except this time each point represents a group of 4 trips. We have chosen two different group sizes to show the amount of noise that is present when less data is available to fit each Gaussian Mixture Model. Overall, however, very similar relations can be found between fuel consumption and time spent in each region, but increasing the number of trips in each group makes them clearer.

Figure 8 shows the changes in correlation coefficients between fuel consumption and the Gaussians, for different group sizes. The largest decrease occurs for *Neutral* and *Driving* at 3 trips in each group. The other two regions display more stable relations.

We also present the positions of the four Gaussian means for all the groups, of both 4 and 8 trips, in Figure 7. It can be seen that there is significantly less variation in the data for the right plot. This observation agrees with Figure 8, where we see a decrease in correlation coefficient with lower number of trips per group.

It is important to stress once more that this data comes from real operation of commercial vehicles, and thus captures the actual situations that take place. In particular, there are individual trips with extremely unusual patterns — for example, over 50% of time spent in neutral. It is therefore important that any analysis method designed for real world use can handle such outliers and does not assume too much conformity to the "expected norm".

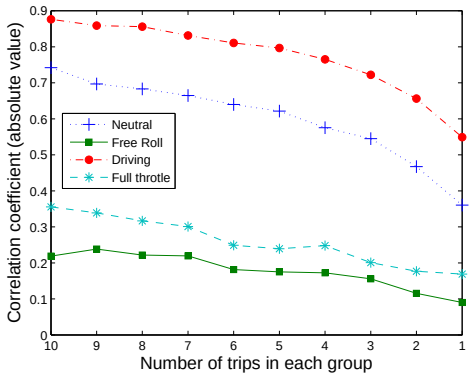


Fig. 8. Correlation of each gaussian vs number of trips in a group

V. Conclusions and Future Work

In this paper we have analysed the usefulness of *Accelerator Pedal Position - Engine Speed* (APPES) maps for evaluating drivers' behaviour from the point of view of fuel consumption. Such maps are an inciting tool, because they are very easy to calculate on-board and have very intuitive interpretations.

We have used data collected during commercial operation of a fleet consisting of five Volvo trucks used by multiple drivers, under the full range of operating conditions on over 1200 trips. This gives us confidence that the results are relevant for future products and services.

The goal of our work was to compare the performance of different driver, describe them using simple to understand features, and correlate those features to key performance indicators, such as fuel consumption. APPES maps can be easily used both for evaluating as well as for training drivers.

Our approach is based on fitting Gaussian Mixture Model to the data, and analysing the relative importance of the four regions we have identified as being of particular interest. We have shown that several of those are highly correlated with factors such as vehicle weight or fuel consumption.

The results presented in this paper are encouraging, but final conclusions have not yet been reached. We have identified both intuitive correlations, e.g., driving in neutral should be avoided, as well as counter-intuitive correlations, e.g., *Free Roll* leads to higher fuel consumption. We believe that that the next step is to find ways to identify external conditions that affect different trips in different ways: for example, it is plausible that *Free Roll* is more common in hilly areas, which would explain higher fuel consumption.

Future work ideas also include investigation of other ways to compare APPES maps. In the current approach, we only investigate the proportion between various Gaussians, but the position of individual means is likely to also contain interesting information. In addition, more complex Gaussian Mixture Models, as well as other formalisms, should be explored in the future.

References

[1] WWW, "http://epp.eurostat.ec.europa.eu/statistics_explained/index.php/freight_transport_statistics_-_modal_split."

[2] H. Liimatainen, "Utilization of fuel consumption data in an ecodriving incentive system for heavy-duty vehicle drivers," *IEEE Transactions on Intelligent Transportation Systems*, 2011.

[3] P. Ting, J. Hwang, J. Doong, and M. Jeng, "Driver fatigue and highway driving: A simulator study," *Physiology & Behavior*, 2008.

[4] M. Irmscher, M. Ehmann, J. Cafeo, and B. Thacker, "Driver classification using ve-dyna advanced driver," *Polyurethane*, 2004.

[5] M. Rafael-Morales and J. C. de Gortar, "Reduced consumption and environment pollution in mexico by optimal technical driving of heavy motor vehicles," *Energy*, 2002.

[6] P. Guo, Z. Li, Z. Zhang, J. Chi, S. Lu, Y. Lin, Z. Shi, and J. Shi, "Improve fuel economy of commercial vehicles through the correct driving," in *Fisita 2012 World Automotive Congress*. Springer-Verlag Berlin Heidelberg, 2013.

[7] WWW, "http://www.eurofot-ip.eu/."

Distributed Evolutionary Algorithm for Clustering Multi-Characteristic Social Networks

Mustafa H. Hajeer
Department of computer science
The University of Memphis
mhhajeer@memphis.edu

Dipankar Dasgupta
Department of computer science
The University of Memphis
ddasgupta@memphis.edu

King-Ip Lin
Department of computer science
The University of Memphis
davidlin@memphis.edu

***Abstract*— In this information era, data from different sources (online activities) are in abundance. Social media are increasingly providing activities and data, relations and interactions (audio, video and texting) among social actors (people), due to increasing capabilities of mobile devices and the ease access to the Internet. More than a billion people are now involved in online social media, and analyzing these interactive structures is a huge data-analytic problem. The primary focus of this work is to develop a clustering algorithm for multi-characteristic and dynamic online social networks. This work uses a combination of multi-objective evolutionary algorithms, distributed file systems and nested hybrid-indexing techniques to cluster the multi-characteristic dynamic social networks. Empirical results demonstrate that this adaptive clustering of dynamic social interactions can also provide a reliable distributed framework for BIG data analysis.**

***Index Terms*— Social Network, Clustering, Graph, Evolutionary, Genetic Algorithm, HDFS, Hadoop Distributed File System, Distributed Fuzzy Clustering.**

1. INTRODUCTION

Social network data can be stored as graphs, which dynamically change with time either by expanding or shrinking. The topology of the graph also changes along with the relationship among nodes. Several algorithms were proposed for community clustering, but only a few of these deal with multi-characteristic and dynamic networks. In particular, most of the existing algorithms work for static and small networks. Very few algorithms can be extended to work on large and dynamic networks. In this work, we developed an evolutionary clustering framework using MapReduce programming paradigm. The framework runs over an HDFS (Hadoop distributed file system). The combination of evolutionary algorithms and HDFS allows us to reduce the computational cost and efficiently perform parameter-less clustering of big and dynamic social network data.

1.1 Data Clustering

Data clustering is the process where a set of elements is divided into a set of Groups/Classes contains these elements; each Group/Class contains data that shares similar characteristics or properties. Similarity measures that are used to cluster these data differ based on the nature of data to be clustered. These measures control the clustering process and the formation of the Classes/Groups; it can be the distance measure in some problems or connectivity in other problems, etc.

1.2 The Problem Statement

Social media and online social networks have become a massive data source and represent virtual relations among people. These social media and online social networks contain important information, which helps in studies such as social behavior, online marketing, and studies about web usage. Recently, these have attracted more attention among the research field and research groups.

Communities can be defined as a group of individuals who interact within a group more frequently than outside the group. Studies on these communities can not only contribute in the above-mentioned areas, but also in addressing security issues. Understanding how these groups are formed and how they change over time, by classifying nodes in a network based on some characteristics, can help in applying theories and techniques to improve data analytic field.

Graph or network clustering has been proven to be NP-complete problem [1], which means there is no known efficient way to find an optimal solution, also the time required to solve this problem increases relatively with the size of the dataset. However, social networks grow fast, and they are dynamic in nature, this means by the time the network is clustered, the network has already changed, and the newly formed network may be different than the recent clusters.

In addition to the above data issues, social

networks are multi-characteristic in nature. There are multiple ways online social communication can occur. For example, Facebook creates a node (a person/page…etc.) and can add another node as a friend when agreed to share information, which constitutes a connection. A node can also send a message to another node, and this is different type of connection, and it can be combined together to form links with values for each characteristic.

Combining different types of connections with multiple characteristics results in large and complex datasets, making it difficult for clustering algorithms to cluster in a reasonable time.

1.3 Defining Network Clusters

In this work, we referred to graph of vertices V and edges E as *G (V, E)*, as an undirected graph. Let number of vertices $|V|=m$, number of edges $|E|=n$ and clustering $C = (C_1, C_2, C_3, \ldots, C_j)$ as a partition of V as disjoint sets. We call *C,* a clustering of *G* containing *j* clusters. The number of clusters *j* has a minimum of $j=1$, when *C* contains only one subset $C_1 = V$, and a maximum of $j=m$ when every cluster C_k contains only one vertex. We identify the cluster C_k as a sub-graph of *G*. The graph $G[C_k]:=(C_k,E(C_k))$, where $E(C_k)=\{\{V,W\} \in E:V,W \in C_k\}$. Then $E(C) = \bigcup_{K=1}^{j} E(Ck)$ is the set of intra-cluster edges and $E\backslash E(C)$ is the set of inter-cluster edges. The number of intra-cluster edges denoted by $m(C)$ and $\bar{m}(C)$ is the number of inter-cluster edges.

As an input, a social network is represented as a set of graphs $SNG=(G_1,G_2,G_3,\ldots,G_Z)$, and the set of graphs-clustering $SGC=(C_{G1},C_{G2},\ldots,C_{GZ})$ where each graph G_i has its own clustering C_{Gi} satisfying conditions mentioned for G and C respectively. Let Z is the number of characteristics in a given dataset. Now, graphs $[G_1,G_2,G_3,\ldots,G_Z]$ have the same set of V but have a different set of E and each C_{GI} is an objective to achieve.

The goal is to find SGCs using a multi-objective optimization and combine them into one clustering $SNC=(SNC_1, SNC_2, SNC_3, \ldots, SNC_X)$, where $SNC:= \bigcup_{L=1}^{Z} C_{GL}$. The set of clustering for social network SNC is not necessarily disjointed, but it is a union of sets where each set is a group of disjointed subsets.

A social network representation with all of its characteristics can lead to a dataset of a huge graph, however, we represented the social network as a set of graphs rather than one graph, each graph represents one characteristic The proposed algorithm takes the social network as a multi-characteristic dataset, then partitions it into set of graphs SGC, where each graph contains edges for only one characteristic. Then each graph G_I is clustered individually by an edge removal algorithm to produce disconnected graph represented by clustering C_{GI}, then by measuring the strength of these clusters. After clustering each graph G_I, we combine elements of each clustering in SGC into one clustering SNC, to produce an overlapped clustering where clusters SNC_1 to SNC_X are not necessarily disjointed.

During the clustering process of each graph, an evolutionary algorithm has been used and Hadoop distributed file system (HDFS) has been used to provide improved performance and speed by partitioning the large datasets into smaller logical blocks.

1.4 Related work

Several algorithms have been applied for data clustering problems; we surveyed these approaches and classified them based on their scalability issues in handling large data.

The list below shows different clustering and data mining techniques along with their advantages and drawbacks.

- In "basic concept of data mining, clustering and genetic algorithm" [24], Tsai-Yang Jea reviewed basic evolutionary algorithms, whose concepts have been analyzed with the following results:

Advantages: Fast results as a clustering algorithm, since it doesn't search the whole space of solutions.

Drawbacks: Final clusters don't show global optimization, the chromosomes represent the whole space each time, and need parameters as inputs, like a number of clusters to start. Also the search for node's similarities itself is time consuming process, and it has to be done in C*N2 time, where N is the number of nodes and C the number of chromosomes. The amount of time taken to produce the result makes it inapplicable for huge datasets.

- Petra Kudová developed (CGA) an evolutionary algorithm for clustering in his paper "Genetic algorithm clustering" [20], published from Academy of Sciences of the Czech Republic, ETID, on 2007. After a deep analysis the advantages and drawbacks for his research have been summarized as follows:

Advantages: faster than regular search approach, and looks for global optimization.

Drawbacks: Similar to Tsai-Yang Jea [4], the search for node's similarities itself is a time consuming process, and it has to be done in C*N2 time, where N is number of nodes and C number of chromosomes. Also each chromosome copies the whole search space as a list, and that is C*S where S is the search space size (billions of nodes and connections in real life). That makes the execution time and space for this algorithm impossible for huge dataset processing.

There are many other related works (listed below) which demonstrate different advantages of clustering algorithms; nevertheless, these approaches have almost similar drawbacks as discussed in reference to the algorithms mentioned in above section. The major drawback with these approaches is they needed some parameters to be fed. For example, a number of clusters, size of clusters and/or number of generations are needed by

evolutionary approaches. It is suggested that these parameters should be found from the dataset and not given as an input parameters. Also these inputs aren't available; hence, it changes the solutions and can result false solutions.

Some other critical drawback is that these approaches copy the search space many times "each solution encoded in a way that represent the whole network", thus resulting in high demands for processing space, which is an impractical approach in real life. On the other hand, if the results are produced slowly, the network has already changed because of its dynamic behavior. Below is another list of papers that share the same disadvantages:

- Evolutionary Clustering and Analysis of Bibliographic Networks [15]
- Multi-objective Evolutionary Algorithms for Dynamic Social Network Clustering [13]
- A Multi-objective Hybrid Evolutionary Algorithm for Clustering in Social Networks [6]
- A framework for analysis of dynamic social networks. [23]
- An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. [21]
- Genetic algorithm and graph partitioning. [22]
- Multi-objective evolutionary clustering of web user sessions. [18]
- Dynamic algorithm for graph clustering using minimum cut tree. [15]
- Community detection in complex networks using genetic algorithm. [2]
- A new graph-based evolutionary approach to sequence clustering. [16]

None of the previous works show interest in a combination of evolutionary and distributed or parallel approach, thus resulting in clustering and search overhead. The approach in this study uses the Hadoop distributed file system and a combination of hybrid hashing and evolutionary algorithms; hence, resulting in a fast, robust, and practical solution according to the dataset size.

1.5 Large "Social Networks" Datasets Clustering

The major problem in traditional combine and test, for search algorithms (the sub problem of clustering), is that the search space is too large and for larger networks, it is impossible to apply. It was observed that the previous works, including evolutionary clustering algorithms, need to search for connections, and this process takes place several times for each node in the network, either for the clustering process or for the evaluation process. On the other hand, traversing the network at each node and looking for all neighbors in a huge dataset is an overhead itself.

For the above reasons, a new distributed evolutionary algorithm was developed for very fast clustering.

The algorithm takes the social network data as an input and then transforms it into multiple graphs, each graph clustered separately to reduce the clustering overhead for one large dataset, and all clusters combined together as final clusters for the social network, thus can lead to overlapped clusters based on the characteristic.

1.5.1 The Evolutionary Clustering

Since the large datasets clustering problem have a huge search space, and the clustering problem is proved to be an NP-Complete problem, as per Jiri Sima and Satu Elisa Schaeffer proved in their work "On the NP-Completeness of Some Graph Cluster Measures". We chose evolutionary algorithms to find an approximation or a close to optimal solution and because of the dynamic nature of the social networks, it was decided to develop an evolutionary algorithm that clusters the network in a fast way and uses the metrics above as a fitness function and evaluation for solutions. Since social networks are full of noise in terms of data, the chromosome encoding developed as a list of weak and noisy edges to be removed, "edge removal and cut based algorithm".

Most traditional evolutionary algorithms were developed in a way that the user would provide some parameters for the network, and the algorithms would process the data based on these parameters. In this study, it is believed that the user shouldn't provide these parameters, but they should be extracted from the network itself. Hence an algorithm is developed in such a way that the network– edges list and its characteristics- has to be the only input, and noisy data was read from the user. During the execution of the algorithm, the framework receives the changes in the network and reflects on the algorithm inputs-the edges list- then the algorithm produces results and creates output solutions based on the most recent network inputs.

jMetal 4.3 is a powerful object-oriented Java-based framework aimed at multi-objective optimization by using metaheuristics. jMetal provides a rich set of classes which can be used as the building blocks of multi-objective techniques. As per Antonio J. Nebro, Juan J. Durillo "jMetal 4.3 User Manual" [18], jMetal is used to develop the evolutionary algorithm.

1.5.2 The Job Distribution and Parallelism

The clustering problem causes overhead, and there are a lot of searches during the process. To make the process faster and less memory demanding, a parallel-distributed evolutionary algorithm is developed. Such algorithms need a synchronization mechanism so the solutions can be produced. In this study, the algorithm is synchronized on a population level (discussed in architecture section): each population will move to the next one only after a complete evaluation.

Evolutionary algorithms make the adopted approach work faster and distributed evolutionary computing makes it possible to process even faster;

resulting in less clustering overhead and increasing of practicality of clustering huge dynamic datasets.

Hadoop distributed file system (HDFS) provides a robust platform for our algorithm where a dataset is distributed among multiple computers and each computer works on the data it has, based on the job it received from the master computer.

2. Hadoop Distributed File System (HDFS)

2.1 HDFS Architecture

Hadoop distributed file system (HDFS) is an open source file system that can combine multiple computers and show them as one system. It also allows users to query and process large datasets in a short time. Additionally the capabilities of the system increase with the number of computers added to HDFS cluster. It can work with unstructured data as well as a collection of structured data. The main idea behind this file system is to split large datasets into smaller ones and spread them over the HDFS cluster with some redundancy mechanism to provide a strong reliability in the cluster.

HDFS cluster provides a collection of services. The primary service among them is MapReduce, where any process (also called job) can be submitted to the master computer (called the master node), and the master computer, in turn, splits the job into tasks for each computer in the HDFS cluster (called data nodes). Each computer performs mapping and reducing for the task assigned to it on the data it has. After mapping and reducing, the result is returned to the master node, and the master node returns result to the HDFS client as files.

2.2 MapReduce

MapReduce is a programming paradigm that can run on HDFS. MapReduce can be used to query from serial files distributed on HDFS cluster. This process made it easier to query huge datasets (petabytes of data) in a faster way than normal indexed databases.

After the file is uploaded to HDFS, it becomes ready to use. MapReduce operations can occur based on RPC (remote procedure calls) for the user to the Job Tracker. The user defines a MapReduce functions and passes them to the Job Tracker. Then the Job Tracker spreads the map function as tasks to Task Tracker. Each Task Tracker works only on the data blocks it has on its DataNode. The map function reads the data and maps it to pair off <Key, value> and passes that to the reducer function. Before the reducer function performs the reduce operation, a shuffle operation exchanges the pairs between TaskTracker, where each reducer takes one Key or more to work on. The reducer then starts reducing the collection of <key, value> pairs that came from the map function into a new single pair of <NewKey, NewValue>. The reduce operation is user defined which writes these results into files and saves it in the HDFS.

3. Evo-Distributed Clustering using HDFS

The proposed framework consists primarily of two main components, the evolutionary component, and the distributed file system. The two components overlap to provide one framework that takes a dynamically changing dataset as an input and splits it into blocks over the distributed file system. The evolutionary part creates chromosomes and is responsible for extracting the parameters, create clustering and evaluating jobs, sending jobs to HDFS cluster, getting the result back to generate new solutions and create new jobs again.

Figure 1 illustrates the proposed framework architecture on a very high level. The components' purposes are listed after the figure.

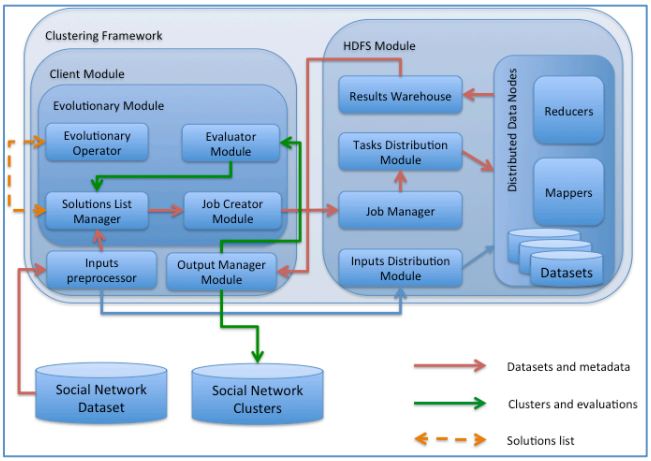


Figure 1: The Proposed framework

The framework composed of two main components:

1. Client Module:
   a. Inputs Preprocessor: processes the social network dataset and transforms it into a multi-dimensional dataset ready for uploading into HDFS.
   b. Output Module: gets evaluation results and sends it to evolutionary module and saves the most recent clusters as clustering results.
   c. Evolutionary Module: processes and executes the evolutionary algorithm and its operator to find best clustering solutions, it also creates clustering and evaluating jobs for HDFS.
2. HDFS module:
   a. Inputs Distribution Module: receives the processed dataset and distributes it over HDFS DataNodes.
   b. Job Manager: receives jobs from evolutionary module and transform it into tasks of Map and Reduce.
   c. Tasks Distribution module: distribute Map and Reduce tasks to TaskTrackers.

d. Results Warehouse Module: saves clustering results of solutions and its evaluations to be read by output manager module on client module.

Every distributed algorithm needs synchronization. The approach adopted in this study, especially the evolutionary algorithm component, needs synchronization because no solutions can be generated if previous solutions (parents) aren't evaluated. To reduce the overhead in job generation, submission, and on HDFS calls, the synchronization is generalized to the simplest level.

The simplest level to which synchronization can be generalized is new population level. The approach can be preceded with the population itself but cannot be moved before evaluating solutions. The algorithm class in jMetal is modified to evaluate the population at once rather than solve on each level. Each group of solutions is sent at once as a clustering and evaluating job to HDFS. The next generation of solutions can only be created after the previous one is already clustered and evaluated.

The pseudo code of the algorithm after the file is uploaded in the HDFS and the changes are continuously being uploaded in parallel with the execution of the algorithm and can be further described as:

1. The problem class generates the inputs for the evolutionary algorithm.
2. The User program runs the algorithm by issuing the command execute for the modified algorithm class, and sends the problem object with its values to the algorithm object by the execute method.
3. The algorithm class on HDFS client generates the first population chromosomes, and keeps it without fitness values ready for evaluation.
4. The algorithm class running on HDFS client contains the unevaluated population into a job along with number of dataset characteristics, and sends the job as a MapReduce job to the JobTracker on the NameNode.
5. The JobTracker distributes the job to TaskTrakers as tasks.
6. A Map and Reduce operation carried out by TaskTrackers writes the solutions as groups with its fitness in results files into HDFS.
7. JobTracker triggers the HDFS client running the user program that the job is done and the solutions with evaluations are ready along with group description for each solution.
8. The HDFS client pulls the results form HDFS results files and writes the groups of the best solutions into network file as the most recent clustering solution, on the other hand the algorithm takes only the fitnesses along with the solutions as an evaluated population. Then, ready to do GA operations like crossover and mutation, it generates a new unevaluated population. Step 4 is repeated while the program is still under execution.

While the system is running, any changes to the dataset is immediately uploaded and merged with the input dataset on the HDFS. In parallel to the running algorithm, the changes are immediately reflected in new solutions.

3.1 Evo-Distributed Solution Space

The primary idea of the encoding scheme for the chromosome (called solution) is an array of integers that represent the noise edges in the network. We want to find and remove them, to create a network of distinct groups without any noisy edges and then find how strong these groups are as an evaluation for each solution. Each TaskTracker works only on the parts of the chromosome that it has in its block and marks them as removed edges. Each integer is an ID to an edge in the dataset, these IDs are created uniquely for each edge before uploading into HDFS.

3.2 Evo-Distributed Objective Functions

The algorithm is configured to be parameter-less, while previous work required the user to enter parameters such as number of clusters, clusters size gap, clusters modularity etc. It is believed that these inputs should be derived from the dataset, to make solutions realistic. So these parameters are made as objective functions, and they are added to the problem class in jMetal framework. The main objective function is composed of an equation, which contains a number of groups, a number of noise edges removed, and the values of each characteristic of the edges itself. Another objective has been added to represent the groups' strengths and uses these values as multiple fitnesses to evaluate the solution. The formula below shows the main objective:

$fc = \left(\sum_{groups} \frac{\sum_N Vn}{Gn}\right) - Er$, Where:

- Fc: objective for characteristic C.
- N: edge N.
- Vn: value of characteristic C on edge N.
- Gn: the group size.
- Er: number of edges removed.

N number of objective functions were created based on number of characteristics the network have, and each one of them reflects a different Fc and results in a different Fc value. Each one of these values is considered to have the objective to maximize in the problem class in addition to the modularity objective. This formula is developed to remove noisy edges. If any edge other than noise is removed, it will result in a lower fitness as a penalty, which lowers the number of edges removed and keep the network in the same topology, it also prevents groups of one node from being created.

3.3 *Tasks at TaskTracker Level*

Tasks on the TaskTracker's level receives a copy of the solutions list (population) and then maps the data read from data blocks into pairs of <Key, Value>, by comparing the data with the solution list received. Each TaskTracker works only on the parts of the solution contained in its data blocks. We call these parts "active parts" and the rest parts of the solutions are called "inactive parts". During the reduce step, values for each solution are collected from all mappers and each solution is made fully active.

The green dashed line is the active part of the solution, and its data is available in the data block on DataNode where the TaskTracker runs. The red dashed line indicates that this data is available in another data block on another DataNode. The reduce step combines all available solutions for one or more Keys, and the main Keys become active because of the data shuffle process that collects all the data for each solution. After the reduce step is carried out, the result for each solution (NewKey) is a data structure we developed. Its content is the list of solutions with its distinct groups and final fitness ready to be written on HDFS, so that HDFS can read it and proceed to the next generation in the algorithm.

4. SYNTHETIC MAPREDUCE JOB ILLUSTRATION

In this section, a MapReduce job is illustrated on the proposed framework to explain exactly what is happening in each step of the Job on TaskTrackers.

After uploading the dataset to HDFS cluster, it gets divided into data blocks each on DataNode, multiple blocks can be on the same DataNode. For illustration purposes, the file is divided into three data blocks, each on separate DataNode as shown in figure 2.

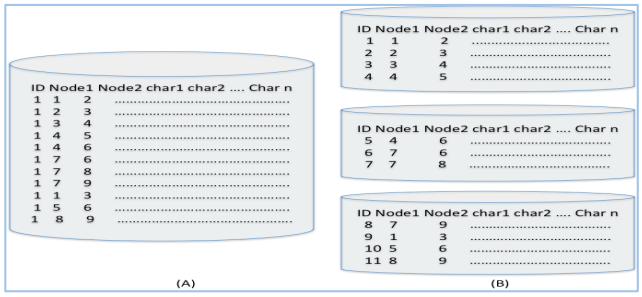


Figure 2: (A) File before uploading to HDFS, (B) Data blocks after uploading to HDFS

Each TaskTracker receives a copy of the Keys-solutions- to the mapper as mentioned in section 4.4, and maps the data into a <Key, Value> pair for the active part of the solutions. These <Key, Value> pairs are written into intermediate files using the custom write-ables we have specially designed. Figure 3 shows the map task on a single TaskTracker on one DataNode.

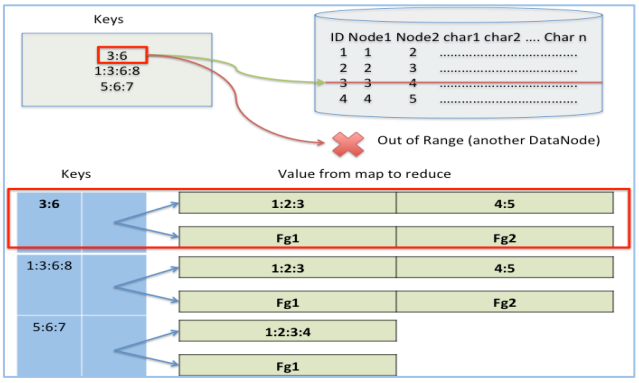


Figure 3: map operation on single DataNode.

Figure 3 shows a data block, which contains edges with IDs {1, 2, 3} and {4}, and the Keys list, which is input for the mapper and consists of three solutions. The first one is responsible for removing noise edges 3 and 6, here edge 3 is considered an active part because the data blocks on this DataNode (only one block available) contains information about it; on the other hand, 6 is inactive since no information about it is in this DataNode. The mapper maps these values after removing the edge 3 for the first solution and writes the values shown in figure 13 as an array list. In figure 14, nodes {1, 2, 3} and {4, 5} are grouped after their fitness is calculated. Other nodes will be combined from other mappers in the reduce phase, and groups can be merged together when there is connections and fitness is recalculated. The same process is continued for solution 2 and 3. Their keys and values list from this mapper is then sent to the reducer.

The reducers shuffle the files so that each Key gets assigned to one reducer along with collection of values as an array, making the whole key active at this stage. The reducer looks for connections between groups, combines groups where there is connection into a single group, then combines multiple array elements into one element and recalculates the fitness by combining subgroups' fitness values in the same formula. Figure 4 (A) shows the result received by a single reducer for the first solution, and (B) shows the solution after reducing and merging groups that have connections.

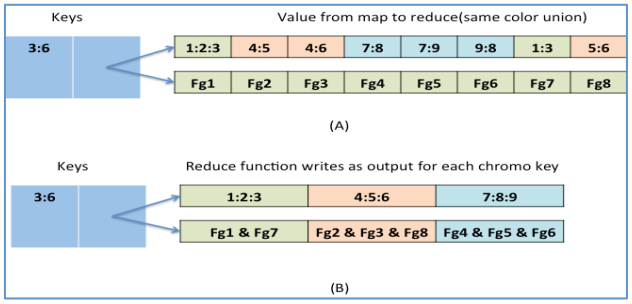


Figure 4: (A) Values list for first solution from mapper collected to reducer. (B) Groups merged after reduce process.

In figure 4 (A) each color should be reduced into one group. The reducer merges these groups using the hybrid HashMap we developed.

After all reducers finish their assigned tasks, all keys are finalized with values and written into HDFS ready to be downloaded to the HDFS client for most recent results files. These values are for the algorithm to generate the next generation (population cycle).

5. Experiments and Analysis of Results

5.1 Large Scale Real World Dataset

A multi-dimensional dataset available online from Youtube servers uses an open source API called Youtube API. The dataset contains 15,088 nodes and 5,574,249 edges; it also contains the following characteristics:

- Number of shared subscribers between two users.
- Number of shared favorite videos
- Number of shared friends between two users excluding the original nodes.
- Number of shared subscriptions between two users.

These datasets were merged together in on dataset and uploaded into HDFS of 4 nodes, three DataNode and one NameNode, and then the following experiments were performed. The first experiment consisted of 100 edges, population size 100 chromosomes. Each generation execution time was around 6000 ms. After ~50 generations the solutions started to form steady groups and 17 groups were found. During the run of the algorithm, the dataset was modified. Five arbitrary groups were added (totally unconnected to any of the previous groups), and the results were immediately reflected. After 3 groups were joined together with some noisy edges, the algorithm took around 10 generations to find those edges and to separate the groups again and go back to steady results, which was expected. The same experiment was done with larger datasets, 200, 400, 800, 1600, 3000 and 10000 edges; table 1 shows the results of these experiments.

Table 1

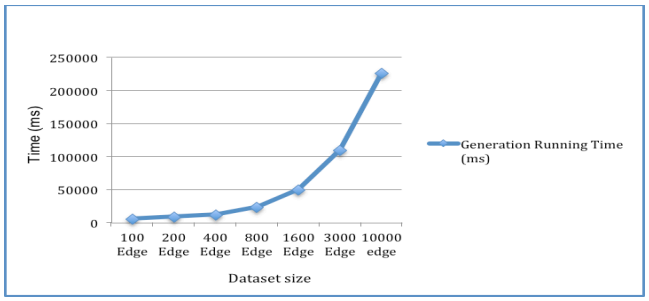
Experimental Results (2 Nodes HDFS Cluster)

Dataset Size	Average Generation execution time (ms)	Number of groups	Number of generations for steady results
100	~6000	17	~50
200	~9000	26	~130
400	~12000	50	~280
800	~24000	90	~740
1600	~50000	236	~2000
3000	~110000	479	~3200
10000	~270000	4932	~7100

5.2 Real World Dataset Experiments Analysis

As results are shown in table 2, there was almost a polynomial relation between the dataset size and the generation execution time. The time difference was caused by the dataset size and the communication latency. More time was needed for shuffling data between the extra cluster components which increased the clustering algorithm time.

Graph 1 illustrates the average generation running time vs. dataset size on the same HDFS cluster.



Graph 1: Average generation running time Vs. dataset size

The small curve at the end of the graph is caused by the difference between the last two datasets in size, and the light curve at the left side of the graph is caused by the communication latency at shuffle step between the maps and reduces. The algorithm doesn't affect the number of groups found. The numbers of groups are totally related to the dataset.

Another positive impact of the approach is that it extracts parameters like the optimal number of groups and group sizes, considering them as objectives where previous approaches consider these values as input parameters, and gives our approach an advantage of a lower number of runs to get the correct inputs, which differ from dataset to another.

A further analysis of results files produced by the framework showed that after making changes in the dataset, the changes immediately reflected on the dataset of the changes do not result in adding groups, or splitting groups. However, if the changes add groups or split current groups into new groups it takes very little time to cluster new changes. Groups that are not affected do not need to be re-clustered and that is because of the evolutionary part of the framework, which keeps a copy of the best solutions and passes it on to the next generation.

5.3 HDFS Experiments

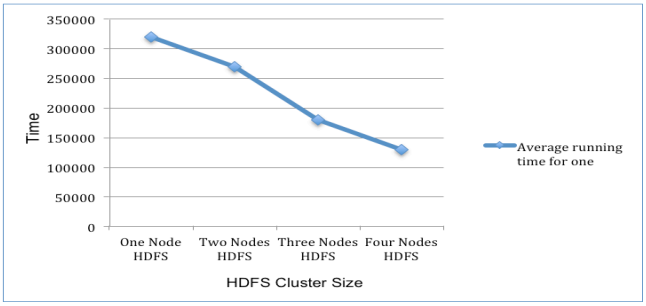
The HDFS components are tested on the same YouTube dataset of 10000 vertices. The number of DataNodes involved in the HDFS cluster are changed and then tested on a single node cluster, 2 Nodes HDFS cluster, 3nodes HDFS cluster and 4 nodes HDFS cluster. Table 2 illustrates the results on multiple clusters for 10000-edge dataset and 100-solution population size.

Table 2

Running Time on Different HDFS Cluster Size

HDFS cluster size	Average Generation running time (ms)	Number of generations for steady results
Stand alone one node	~320000	~7100
Two nodes HDFS cluster	~270000	~7110
Three nodes HDFS cluster	~180000	~7120
Four nodes HDFS cluster	~130000	~7110

Table 2 shows that the HDFS cluster size had negligible effect on the number of generations needed for steady results to start being produced, thus only the size of HDFS cluster affected the running time and had no effect on the results of clustering the dataset. Graph 2 illustrates the relationship between the sizes of the HDFS cluster vs. the average running time for each generation.

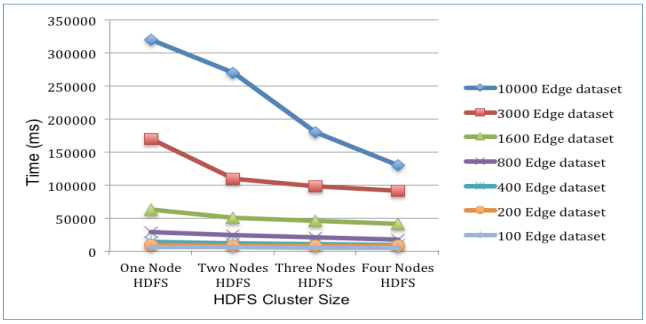


Graph 2: Average generation running time vs HDFS cluster size

The average running time does not decrease in polynomial form since increasing the number of nodes in HDFS cluster reduces clustering work. Conversely, the communication and shuffling latency time increases and communication between nodes does take more time.

5.4 Comparative Results

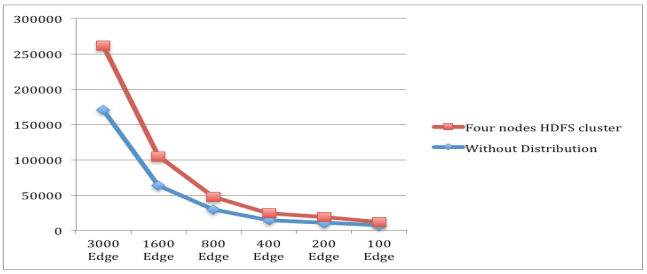
For comparison between average running times on different sizes of HDFS clusters and different dataset sizes, graph 3 illustrates the performance of HDFS cluster and its effect on clustering time. Graph 3 also explains the effect of the HDFS size on clustering performance over time variable. Results on one node HDFS cluster illustrate the execution of the evolutionary algorithm without distribution.



Graph 3: average generation running time vs. different dataset size on different HDFS cluster size

By comparing results on graph 3, the results shows that HDFS cluster size have a big effect on the running time, and this effect decreases with the dataset size, on a very small dataset the HDFS size starts to lose it affect, since communication time between HDFS components and the shuffling latency takes more time than clustering on one node HDFS cluster. The almost steady change in average running time for 100,200 and 400 datasets size clearly proves the analysis above.

Graph 4 illustrates the time performance to cluster 3000-edge network, to compare the difference of executing evolutionary clustering algorithm without distribution, with executing the algorithm on HDFS cluster of 4 nodes.



Graph 4: Single algorithm vs distributed

Graph 4 illustrates the difference for evolutionary clustering without HDFS distribution, in comparison with four node HDFS distribution. Results show that there is a slight difference on small datasets; however, the distribution of the algorithm execution provided a noticeable difference for larger datasets. Distributed evolutionary clustering algorithm improved the performance by influencing the computation performance on time variable. The same execution steps with almost half the time. However on small datasets, results show that there is slight difference or almost no difference in execution time. Deep analysis showed that communication and shuffling step on four Nodes HDFS cluster consumes almost the same time difference the distribution saves in map and reduces steps.

6. Concluding Remarks and Future Work

Clustering social network and huge datasets is an NP-complete problem, optimization techniques proved efficient in solving such problems in the past, however combining such algorithms with the new distributed systems, lead to noticeable improvement. Also Defining multi-Characteristics social network dataset as a collection of multiple layers graphs, where each have the same set of nodes but different set of edges based on the characteristics, improves the computation and makes less overhead in computation aspects. On the other hand, considering the social network graph as a single layer represents all links and characteristics, leads to more complex graph with multiple links between same end nodes thus resulting in higher complexity for the same algorithm.

Distributed systems do not affect the solutions as much as the primary algorithm does; however, it has a big influence on the performance of the algorithm used, speeds up the process, and reduce work load and memory usage. Distributed computing opens new directions for algorithms to be expanded, and distributed file systems allow computing power to be able to work on a larger scale.

References:

[1]. Šíma, J., & Schaeffer, S. E. (2006). On the NP-completeness of some graph cluster measures. In SOFSEM 2006: Theory and Practice of Computer Science (pp. 530-537). Springer Berlin Heidelberg.

[2]. A. Sima Uyar & Sule Gunduz Oguducu. A new graph-based evolutionary approach to sequence clustering. In ICMLA '05: Proceedings of the Fourth International Conference on Machine Learning and Applications, pages 273–278. IEEE, 2005.

[3]. A. Sima Uyar & Sule Gunduz Oguducu. A new graph-based evolutionary approach to sequence clustering. In ICMLA '05: Proceedings of the Fourth International Conference on Machine Learning and Applications, pages 273–278. IEEE, 2005.

[4]. Antonio J. Nebro & Juan J. Durillo, jMetal 4.3 User Manual, January 3, 2013

[5]. B. Saha & P. Mitra. Dynamic algorithm for graph clustering using minimum cut tree. In Proceedings of ICDM Workshops, pages 667–671, 2006.

[6]. Babak Amiri, Liaquat Hossain & John Crawford, Multiobjective Hybrid Evolutionary Algorithm for Clustering in Social Networks,

[7]. C. A. C. Coello, G. B. Lamont & D. A. V. Veldhuizen. Evolutionary Algorithms for Solving Multi-Objective Problems (Genetic and Evolutionary Computation). Springer-Verlag, 2006.

[8]. C.-K. Cheng & Y.-C. Wei. An improved two-way partitioning algorithm with stable performance [VLSI]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 10(12): 1502–1511, 1991.

[9]. Clara Pizzuti. Ga-net: A genetic algorithm for community detection in social networks. In PPSN, volume 5199 of Lecture Notes in Computer Science, pages 1081–1090. Springer, 2008.

[10]. Community Detection and Mining in Social Media. Lei Tang and Huan Liu, Morgan & Claypool, September 2010.

[11]. E. Zitzler & L. Thiele. Multiobjective optimization using evolutionary algorithms - a comparative case study. In PPSN V: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature, pages 292–304, 1998.

[12]. Jianbo Shi & J. Malik. Normalized cuts and image segmentation. IEEE Trans. on Pattern Analysis and Machine Intelligence, 22(8):888–905, 2000.

[13]. Keehyung Kim, RI (Bob) McKay& Byung-Ro Moon, Multiobjective Evolutionary Algorithms for Dynamic Social Network Clustering

[14]. M E Newman & M Girvan. Finding and evaluating community structure in networks. Phys Rev E Stat Nonlin Soft Matter Phys, 69(2):026113.1–15, 2004.

[15]. Manish Gupta, Charu C. Aggarwal, Jiawei Han & Yizhou Sun, Evolutionary Clustering and Analysis of Bibliographic Networks,

[16]. Mursel Tasgin & Haluk Bingol. Community detection in complex networks using genetic algorithm. In ECCS '06: Proc. of the European Conference on Complex Systems, Apr 2006.

[17]. Mursel Tasgin & Haluk Bingol. Community detection in complex networks using genetic algorithm. In ECCS '06: Proc. of the European Conference on Complex Systems, Apr 2006.

[18]. Pasi Fränti & Olli Virmajoki, POLYNOMIAL TIME CLUSTERING ALGORITHMS DERIVED FROM BRANCH-AND-BOUND TECHNIQUE, University of Joensuu, Department of Computer Science, P.O. Box 111, FIN-80101 Joensuu, FINLAND

[19]. Peter Rousseeuw. Silhouettes: a graphical aid to the interpretation & validation of cluster analysis. J. Comput. Appl. Math, 20(1):53–65, 1987.

[20]. Petra Kudová, Clustering Genetic Algorithm, Department of Theoretical Computer Science, Institute of Computer Science, Academy of Sciences of the Czech Republic, ETID 2007

[21]. R. Breiger, S. Boorman, & P. Arabie. An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling* 1. Journal of Mathematical Psychology, 12(3):328–383, 1975.

[22]. T. N. Bui & B. R. Moon. Genetic algorithm and graph partitioning. IEEE Transactions on Computers, 45(7): 841–855, 1996.

[23]. T. Y. Berger-Wolf & J. Saia. A framework for analysis of dynamic social networks. In KDD '06, Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 523–528, 2006.

[24]. Tsai-Yang Jea, Basic concepts of Data Mining, Clustering and Genetic Algorithms, Department of Computer Science and Engineering, SUNY at Buffalo.

[25]. X. Cheng, C. Dale, & J. Liu. Statistics and social network of YouTube videos. In IWQoS '08: Proceedings of the 16th International Workshop on Quality of Service, pages 229–238, 2008.

[26]. Peng Gang Sun, Lin Gao & Shan Shan Han, Identification of overlapping and non-overlapping community structure by fuzzy clustering in complex networks, Information Sciences: an International Journal, Volume 181 Issue 6, March, 2011, Pages 1060-1071.

[27]. Jianzhi Jin,Yuhua Liu, Laurence T. Yang & Naixue Xiong Fang Hu, An Efficient Detecting Communities Algorithm with Self-Adapted Fuzzy C-Means Clustering in Complex Networks, TRUSTCOM '12 Proceedings of the 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.

[28]. Qun Liu, Zhiming Peng, Yi Gao & Qian Liu, A new K-means algorithm for community structures detection based on fuzzy clustering, GRC '12 Proceedings of the 2012 IEEE International Conference on Granular Computing (GrC-2012).

Implementation of Artificial Neural Networks in MapReduce Optimization

Changlong Li[1], Xuehai Zhou[1], Kun Lu[1], Chao Wang[1], Dong Dai[2]
[1] *University of Science and Technology of China,*
[2] *Texas Tech University, USA*
Email: {liclong, xhzhou, local, saint}@mail.ustc.edu.cn, dong.dai@ttu.edu

***Abstract*—The ability to handle large datasets has become a critical consideration for the success and ability of industrial organizations such as Microsoft, Amazon, Yahoo! and Facebook. As an important cloud computing framework for data processing, MapReduce is widely used by these organizations. However, its performance has been seriously limited by its stiff configuration strategy. In practice, even for a single simple job in a MapReduce framework, a large number of tuning parameters have to be set by end users, who often run into performance problems since they do not know how to configure them. Besides, once set, most parameters will never be changed again. This may easily lead to performance loss due to some misconfigurations. In this paper, we present a soft computing technique: Artificial Neural Network(ANN) to achieve the automatic configuration of parameters for MapReduce. Given a cluster and MapReduce job, frameworks can adapt the hardware and software configurations to the system dynamically and drive the system to an optimal configuration in acceptable time with the help of ANN. Experimental results show that ANN has a great contribution to optimize system performance and let the system at the speedup of 9x.**

***Keywords*-Neural Network; MapReduce; Automatic Configuration; Optimization; Big Data;**

I. Introduction

Since its inception, MapReduce [1] has frequently been associated with cloud computing and large-scale datasets. Widely deployment and application at industry organizations have thrust this programming framework to the forefront of cloud computing and data processing application domain. There are growing interests in deploying such a framework in the Cloud to harness the unlimited availability of virtualized resources of cloud computing. For example, Amazon's Elastic MapReduce provides data processing services by using Hadoop on top of their compute cloud EC2. However, the performance of MapReduce need to be further improved: a MapReduce program can run 2-50x slower than a similar relational query run on an RDBMS with identical hardware [2], Anderson also showed that Hadoop which is an implementation of a MapReduce framework performed bulk data processing at a rate of less than 5 megabytes per node per second [6]. Current technologies mainly achieve MapReduce optimization through the way of data locality, cluster heterogeneity and scheduling strategy. However, pushing the responsibility for performance optimization into underlying code will make the code complex and hard to maintain. Besides, these technologies are always limited by their scalability and operability.

Existing programming environments for running MapReduce jobs in a cloud platform aim to remove the burden of hardware and software setup from end users. However, they expect end users to provide appropriate parameters for running a job. In the absence of automatic configuration scheme, users are forced to make job provisioning decisions manually using best practices. As a result, customers may suffer from a lack of performance guarantee. The difficulty of setting up those parameters contains two folders. First, empirical evidence suggests that the performance of submitted job is some complex function of the configuration parameter settings. Second, the features of cluster hardware (memory capacity, network bandwidth) as well as the characteristics of MapReduce program (data-intensive, compute-intensive) also have significant impact on the performance.

As the performance of MapReduce is seriously limited by its configuration, therefore many techniques that evaluate and configure system parameters for traditional cluster have been proposed [7]. However, the effectiveness of these approaches are often inter-dependent, some of them focus on a single element only and hence are not able to address the complex, high-dimensional configuration problem. Others such as Minerva [3] are extremely complex for non-expert users, requiring expertise with advanced tools and a large number of experiments. Kambatla et al. [5] found this problem and proposed to select optimal configuration parameters using a given set of resources, but there is no guidance on deciding the appropriate number and type of resources to be allocated. So it is critical and necessary to propose an auto-configuration scheme that integrates various aspects of factors as well as parameters to provide optimal configuration in acceptable time for MapReduce.

There are some complex functions between configuration parameters, hardware features, program characteristics, and MapReduce performance. In this paper, we propose ANN, an artificial neural network, to optimize system performance through learning the internal relationship between impact factors. Given an application to run on a given platform, ANN automatically searches for optimized system configurations from candidate settings. We analyze the relationship between influential configurations and system performance with the implementation of the network and then dynam-

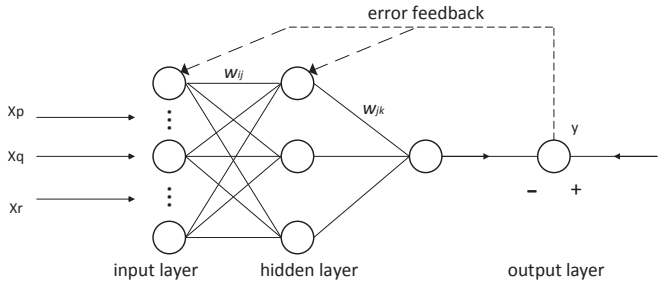


Figure 1. ANN: x_p represents the setting of configuration parameters, x_q represents program characteristics and x_r represents hardware features.

ically adjust the parameters to achieve performance optimization. After trained on the target platform, ANN learns the internal relationship and makes predictions accordingly. Based on the prediction, the network could recommend an optimized configuration and adjust them dynamically. New collected data will be learned by ANN to improve the accuracy of recommendation. Experimental results show that MapReduce framework is able to adapt the configurations to the system dynamically and drive the system to an optimal configuration in acceptable time.

This paper is organized as follows. In Section II, we describe the design of ANN. Section III shows evaluation based on real-world deployment. We introduce the related work in Section IV, and finally, Section V presents our conclusion and the future work.

II. Design

The empirical evidence from Section III suggests that the performance of a MapReduce job J is some complex function of the job configuration parameter settings. In additional, the features of the hardware as well as program characteristics will impact its performance. There exists some function F_J such that:

$$y = F_J(\vec{p} \in \vec{P}, \vec{q} \in \vec{Q}, \vec{r} \in \vec{R}) \quad (1)$$

Here, y represents a metric of system performance(e.g., CPU usage), $\vec{P} = \{x_1, ..., x_p\}$ represents the setting of configuration parameters which have significant impact on performance (parameters have little impact on performance are ignored), $\vec{Q} = \{x_{p+1}, ..., x_{p+q}\}$ represents the characteristics of program and $\vec{R} = \{x_{p+q+1}, ..., x_{p+q+r}\}$ represents the features of hardware. Since these parameters display strong performance interactions with other parameters, the function relationship is complex and unpredictable. Thus we design an artificial neural network to learn this function.

ANN, the artificial neural network, consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation. Each neuron is shown as a circle in the diagram, and the lines connecting them are known as weights. As illustrated in Figure 1, the $\vec{P}$, $\vec{Q}$ and $\vec{R}$ is the input and performance metric y is the output. Input variables are applied to the input units at the left of the diagram. The input layer send data via synapses to the hidden layer, and then via more synapses to the output layer. If the output $\hat{y}$ were inconsistent with y, the target value of performance metric (the actual output in samples), error messages will be back propagated and values of weights will be adaptively modified during the process of network training. Hence, the network provides a direct mapping from system parameters onto weight values associated with the best fit function. Although the training of neural networks is computationally intensive, the trained networks can process new data very rapidly. The more comprehensive the training set, the more representative the trained neural network becomes.

Given a cluster and MapReduce job, we can think of $\vec{P}$ which represents the setting of configuration parameters as the only factors that affect system performance. We provide a table to log all these parameters and their possible values. Once training complete and the relationship learned, our recommendation system will traverse the table and let each parametric combination as input for prediction, then recommends an optimized configuration and adjust them dynamically. Since the table size is not very large, we can quickly lock the optimal solution and adjust them dynamically. On the other hand, new configurations and their corresponding performance will be collected as ANN's feedback information to help improving its accuracy and efficiency through self-learning.

The problem of optimizing the parameters of a given functional form to fit experimental data points is frequently encountered in data analysis. In this Section we have shown that the artificial neural network can provide a direct mapping from the measured data onto the parameter values associated with the best fit function.

III. Evaluation

A. Experimental Setup

In this experiment, we choose Hadoop as our platform: as an open source implementation of a MapReduce framework, Hadoop is widely used in production deployments for applications such as log file analysis, scientific simulation, Web indexing, report generation and genome sequencing [8]. All our experiments were performed using local cluster running on 9 nodes, with 1 master and 8 worker nodes: all machines have Xeon dual-core 2.53GHz processor with 6GB memory. We choose the IOR [4] synthetic benchmark as it is generic, open-source, and highly configurable. ANN carries out the initial training by running the benchmarks on Hadoop. For each training run, it collects the performance metric with the candidate configurations. We choose dstat tool on each node to measure CPU usage (in terms of User, system, Idle and Wait percentages), disk usage (number of blocks in and out) and network usage (bytes/second into and out of network card) every second.

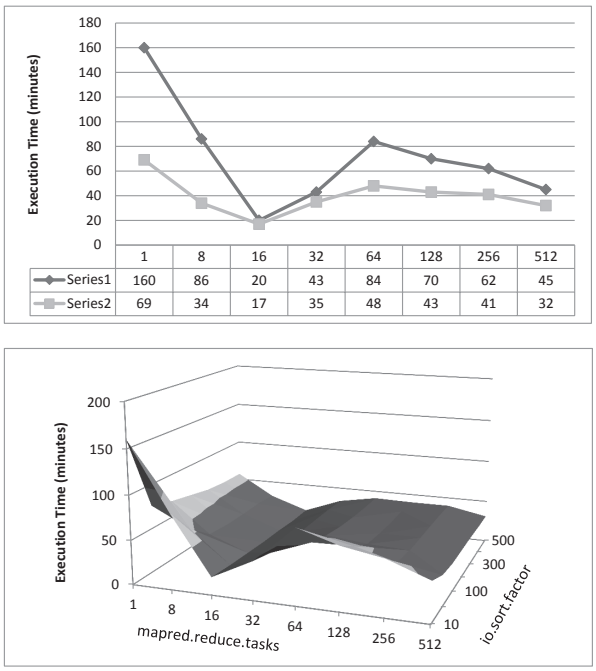


Figure 2. Execution time of TeraSort benchmark with 100 GB datasets. There are some complex relationship between performance and parameters *mapred.reduce.tasks* + *io.sort.factor*. Series1 means *io.sort.factor = 10*, and Series2 means *io.sort.factor = 100*.

B. Parameter Analyze

There are 200+ parameters are specified to control the behavior of MapReduce programs in Hadoop-2.2.0 and more than 30 of these parameters have significant impact on job performance. Here we divide the parameters into three categories. (i) Most parameters are configured for normal operation and have no impact on performance. For example, *dfs.datanode.dns.nameserver* determines IP address and *dfs.datanode.address* configures namenode's port information. (ii) Some parameters can be set when job is submitted. To run the program in Hadoop, the system will create a job configuration object based on some given parameters from users. This job configuration object usually is highly relevant with the performance. (iii) Apart from the job configuration parameters whose values are specified explicitly by users, there are a large number of parameters whose values are specified implicitly by the system.

All these parameters except the first type control various aspects of job behavior during execution such as memory allocation, I/O optimization and network bandwidth usage. For example, *mapred.reduce.tasks* determines the number of reducer tasks and *dfs.block.size* denotes HDFS block size. Although different parameters affect performance in different ways, they are not independent. On the contrary, they have strong performance interactions with one or more other parameters. Here we present some empirical evidence to demonstrate differences in job running times between good and bad parameter settings in Hadoop. Figure 2 shows the execution time of TeraSort on cluster for 100GB datasets. The two parameters, *mapred.reduce.tasks* and *io.sort.factor*, are varied in these figures while all other job configuration parameters are kept constant. In Figure 2-a, the function relationship between execution time and *mapred.reduce.tasks* is changed when the value of *io.sort.factor* is different. From the comparison we know that: (i) Configuration parameters have significant impact on system performance. (ii) A number of instances of inter-parameter interactions were seen in our experiments. (iii) There are some complex and high-dimensional function relationship between parameters and system performance.

C. Performance Comparison

Statistics show that there are more than 200 parameters are specified to control the behavior of submitted job in Hadoop-2.2.0, and more than 30 of these parameters have significant impact on job performance. Figure 3 (a), (b), (c) and (d) compare the impact of using the default Hadoop configuration with ANN's auto-tuned configuration on the job execution time. It compares the Hadoop configurable parameter values due to ANN's auto-configuration with the default Hadoop values. The network provides different configuration scenarios for running TeraSort, WordCount and PiEstimator benchmark with input data of 1GB, 10GB, 50GB and 100GB. As shown in Figure 3, with the help of ANN, Hadoop is able to adapt the hardware and software configurations based on the system dynamically and drive the system to an optimal configuration in an acceptable time. We know from the comparison that the running time of benchmarks are about 9 times faster than default configuration. Moreover, the effect of ANN is more obvious when datasets scale increased. In particular, parameters *io.sort.factor* and *mapred.reduce.tasks* have significant different values. Hadoop TeraSort benchmark sets the default value of *mapred.reduce.tasks* to about 0.9 times the total number of reduce slots in the cluster.

D. Fault Tolerance

Other than performance, we also measured the fault tolerance of the network. As we know, the success of neural network's machine learning depends to a great extent on the sample. So if the sample size is insufficient or the quality is not high, ANN cannot learn the relationship accurately. Here we consider an extreme case, since the hardware environment changed (or any other reasons), the sample will be out of date as a result. In this case, the configuration scenarios putted by ANN may be even worse than the default because the real relationship may have changed. Our feedback mechanism solves this challenge. In the process of self-learning, the proportion of sample is gradually replaced

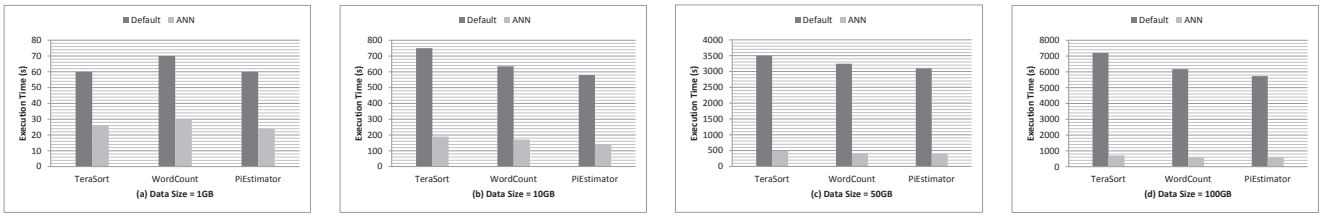


Figure 3. Execution time of TeraSort, WordCount and PiEstimator benchmark with 1GB, 10GB, 50GB and 100GB datasets. The performance of MapReduce is greatly improved with the help of ANN. Besides, the effect of optimization is more obvious when datasets scale increased.

by subsequent data. In our evaluation, the error or deficiency of sample can be covered by the measurement data.

IV. Related Work

Recently MapReduce infrastructures and its open source implementation Hadoop have gained much attention, and different approaches have been developed to automatically and efficiently optimize configurations. W. Zheng [10] presented an approach to achieve automated configuration. Although this heuristics approach efficient in time consuming, sometimes it cannot reach global optimality. The method of eliminate database tuning knobs through code rewrites have been used in Hadoop, but setting the parameter through code modifying can make the code complex and hard to maintain. There are also many other schemes proposed to achieve automatic configuration, Gideon et al. study the impact of different data sharing options for scientific workflows on Amazon EC2, Elastisizer selects the proper cluster size and instance types for MapReduce workloads running in the cloud. Most of these existing efforts assume certain knowledge on the application/middleware internals, but they did not solve the essential problem: finding the internal relationship. We solve the problems above by ANN. It achieves system-wide parameters configuration automatically through training and self-learning. Also, ANN offers the expandability and flexibility that allow it to work across cloud platforms and across hardware updates.

V. Conclusion and Future Work

Optimization through the way of automatic configuration is becoming an increasingly important concern for MapReduce frameworks. In this paper, the technology of Artificial Neural Network is implemented to learn the internal and high-dimensional function relationship. It accurately fits the relationship between performance, parameters and other factors through training and further learning. We also present a recommend scheme to MapReduce with the help of ANN, allowing for parameters to be automatically adjusted between tasks without the need for the framework to be shutdown or restarted. The experiment results demonstrate the performance of our approach.

In the near future, we plan to keep on investigating the factors that affect system performance, and consider to improve the speed and accuracy of the neural network.

Acknowledgment

We deeply appreciate the anonymous reviewers for their insightful comments and suggestions. This work was supported by the National Science Foundation of China under grants No. 61379040, No. 61272131 and No. 61202053, Jiangsu Provincial Natural Science Foundation grant No. SBK201240198.

References

[1] J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *Proc. of ISDI*, 2004.

[2] A. Pavlo, E. Paulson, A. Rasin, D. J. Abadi, D. J. Dewitt, S. Madden, and M. Stonebraker. A comparison of approaches to large-scale data analysis. In *SIGMOD Conference*, pages 165-178, 2009.

[3] G. Alvarez, E. Borowsky, and S. e. a. Go. Minerva: An Automated Resource Provisioning Tool for Large-scale Storage Systems. *ACM Transactions on Computer Systems (TOCS)*, 19(4):483-518, 2001.

[4] H. Shan, K. Antypas, and J. Shalf. Characterizing and Predicting the I/O Performance of HPC Applications Using a Parameterized Synthetic Benchmark. In *SC. IEEE*, 2008.

[5] K. Kambatla, A. Pathak, and H. Pucha. Towards optimizing hadoop proisioning in the cloud. In *HotCloud Workshop in conjunction with USENIX Annual Technical Conference*, 2009.

[6] E. Anderson and J. Tucek. Efficiency matters. In *SIGOPS Workshop*, pages 40-45, 2010.

[7] A. Verma, L. Cherkasova, and R. Campbell. ARIA: automatic resource inference and allocation for MapReduce environments. In *Proc. IEEE/ACM Int'l Conference on Autonomic Computing(ICAC)*, 2011.

[8] Chao Wang, Xi Li, Xuehai Zhou, Jim Martin, Ray C. C. Cheung: Genome sequencing using mapreduce on FPGA with multiple hardware accelerators, 2013.

[9] Z. Liu, H. Li, and G. Miao. MapReduce-based backpropagation neural network over large scale mobile data, in *ICNC10*, 2010.

[10] W. Zheng, R. Bianchini, and T. D. Nguyen. Automatic configuration of internet services. In *Proc. of ACM European Conference on Computer Sytems (EuroSys)*, 2007.

A New Effective Information Decomposition Approach for Missing Data Recovery

Shigang Liu[1], and Honghua Dai[1]
[1]School of Information Technology, Deakin University 221, Burwood Highway, VIC 3125, Australia

Abstract - *It is well recognized that missing data could cause severe problem in data mining. Due to its importance lots of work has been done in the past. Several algorithms [5-8] are proposed for missing data recovery. This paper presents a new 1-dimensional linear information decomposition (1-DLID) approach which is easier for use in missing data recovery. In this article, we study one particular problem, in which 1-dimensional data set is given and certain percentage of data are missing without any other additional information. Then the proposed 1-DLID method is used for creating the complete data set from both the generated data set and real-world data set. Comparatively, our experiments showed that the proposed method is reliable and can be used for the recovery of data set with missing values. The advantages of the proposed method are: 1) Will not change the distribution of the data set. 2) Easy to use for 1-dimensional dataset. 3) Have a higher accuracy, especially there is 10%~30% data missing. 4) No need to provide the historical data set.*

Keywords: Information decomposition, Missing data recovery, 1-dimensional data

1 Introduction

In recent decades, missing data recovery have been broadly studied and applied in various domains in order to solve many complicated and important real-world problems, such as pattern recognition, natural language processing, medical diagnosis, and so on[1,2], in the hope of improving performance. Meanwhile missing values recovery imputation is an existing yet challenging problem in both machine leaning and data mining [3]. On the other hand, Missing values in real-world data cause severe problem for the learning and knowledge discovery. In most cases, missing data problem is caused by data logging procedure and systems. Let's take a manufacturing line for example, it is impossible to record all the line variables of all the products at any time. That is to say, variables which are recorded are only certain kind of products, which can be regarded as incomplete measurement data values. In addition, the topic of missing data has attracted considerable attention in the last decade, as evidenced by several recent trends. First, many graduating PhDs in statistics and computer science are now claiming "missing data" as an area of research. Second, it has become difficult to publish empirical work in sociology without discussion of how missing data was handled. Thirdly, several methods for handling missing data has sprouted-up over the last few years, which will be discussed later. Missing data is important to consider, because they may lead to substantial biases in analyses [4] and in sometimes result in incorrect decision making. On the other hand, missing data could be harmless except reducing statistical power.

The approach discussed in this paper is good at processing data set with data missing at random (which is called MAR) and helpful for analyzing the incomplete data set. Before our approach is presented, we would like to discuss the identification scenarios for missing values pointed out by Little and Rubin (1987) [5]. Based on the values of attributes and the missingness of attributes, the categories of missing data include missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR). If data are MAR or MCAR, they can also be referred to as "ignorable" data while those MNAR are "non-ignorable" [6]. There are various methods that have been proposed to deal with missing data with each of these methods premised on a specific missing data mechanism [7-8], which will be discussed in the following section.

This paper is structured as follows: Section 2 is a brief overview of previous methods. Section 3 describes terminologies. That is to say we give some general knowledge used in the algorithm. Section 4 is devoted to the introduction of our 1-dimensional linear information decomposition approach. Section 5 presents experiment results explanation based on generated data set. The real-world values experiments are organized in section6. Section 7 concludes this paper and section 8 describes our future work.

2 Current techniques and existing problems

There are many methods that used in missing data recovery [7-8]. However every method has its own problems. For example, filling manually is very time consuming when the missing data set is very large, it is impossible to make good use of this approach; MI algorithm is flexible and time expensive. Another drawback is that this method is adequate for statistics better than data mining. Moreover, it is widely used in multivariate normal data. Last but not least is that some distribution for the stochasticity must be assumed, which can be problematic as well [7] etc.. In this paper, regarding to 1-Dimensional data set without any historical dataset

provided, we summarize the most commonly used methods as follows:

1) Listwise deletion[7-8]. By far most times researchers would like to simply omit those instances with missing attribute-values and run the analyses only on the complete instances.

The major problems of this method are that when parametric model based on the attribute-values are not MAR this approach does not work well. Moreover, this method may lead to a large amount of data being thrown away, miss some important information.

2) Filling Manually[7-8]. This method based on the experience of the experts and used in some of statistical area with small missing data set.

The major problem of this method is that it is time consuming particularly when the missing data set is very large. It is impossible to make good use of this approach.

3) Mean/Mode Imputation[7-8]. It means Replacing missing values with the sample mean. In fact, this method is simple and save time when the missing data is numerical rather than non-numerical.

The major problems of this approach are it will make the distribution more peaked around the mean and assumes all the missing data should be MCAR.

4) The Expectation Maximization(EM) algorithm. The EM algorithm is an elaborate technique for incomplete data or data set with missing values. The EM algorithm[9-10] is an approach used for finding the maximum-likelihood estimate of the parameters based on the assumption of the distribution for a given incomplete data set. Usually the EM algorithm is used for the following two situations, first there are indeed missing values, because of limitation of observation process. The second situation is when optimizing the likelihood of a function, it is analytically intractable while the likelihood function could be simplified by assuming the existence values for additional, however, missing or latent parameters. There are two steps in EM algorithm, E-step (expectation step) to compute the expectation of the expected value of the complete data log-likelihood with respect to the unknown data given the observed data and current parameter estimates. The second step (M-step) is to maximize the expectation which was computed in the E-step. That is to say, we find out the new parameter $\theta^{(i+1)} = \arg\max_{\theta} Q(\theta, \theta^i)$. These two steps are repeated until $|\theta^{(i+1)} - \theta^i| \le error$, the error is the values we fixed before the two steps are repeated.

For this paper, during the experiments the initial values of the missing data are produced by computer in random. Then we use it to finish the E-step, after that we can maximize the likelihood function and get the θ, then with the help of θ we can renew the missing data, and then go to M-step.

Problems: Firstly, it is time consuming towards multi-dimensional dataset. Secondly, the algorithm doesn't produce standard errors for the parameters. Thirdly, it may converge to a local maximum of the observed data likelihood function, and this depends on starting values.

Overall, every methods have its own problems, that is to say it is hard to find an algorithm that suitable for all kind of problems. For example, it is said handling missing data by eliminating cases with missing data ("listwise deletion" or "complete case analysis") will lead to the predicted results away from the reality when the remaining data cannot be representative of the whole data set. Moreover, the Expectation Maximization (EM) algorithm is also one method that is used for data mining, however, it can be regarded as an auxiliary method such as bootstrapping when obtaining standard errors.

The major contribution of this paper is to propose a new method which is 1-dimensional linear information decomposition (1-DLID) approach used for missing data recovery. The 1-DLID method is useful in two aspects, one is that it can be used for the recovery of missing data; on the other hand, it can create or generate data for the incomplete data. Compared with other algorithms, 1-DLID has its own advantages. For example, unlike EM algorithm, 1-DLID approach do not need to set the latent variables even more, we do not have to know any kind of probability distribution. Therefore, 1-DLID approach is easy to use and can be used in any kind of one dimension numerical data set with missing values.

3 Basic terminologies used in missing data recovery

3.1 Information Distribution [11]

Let $X = \{x_1, x_2, \cdots, x_n\}$ be a sample observed from an experiment, and $U = \{u_1, u_2, \cdots, u_m\}$ be the discrete universe of X.

A mapping from $X \times U$ to [0,1],

$\mu : X \times U \to [0,1]$,

$(x,u) \to \mu(x,u)$

is called an information distribution of X on U, if $\mu(x,u)$ has the following properties:

1)Reflexive. $\forall x \in X$, if $\exists u \in U$, such that $x = u$, then $\mu(x,u) = 1$.

2)Decreasing. For $x \in X$, $\forall u', u'' \in U$,

if $\|u' - x\| \le \|u'' - x\|$, then $\mu(x,u') \ge \mu(x,u'')$.

3)Conserved. That is to say $\sum_{j=1}^{m} \mu(x_i, u_j) = 1$,

$i = 1, 2, \cdots, n$.

3.2 1-Dimension Linear Information Distribution[11]

Let $X = \{x_i \mid i = 1, 2, \cdots, n\}$ be a given sample, R is the universe of discourse of X, and $U = \{u_1, u_2, \cdots, u_m\}$ is the discrete universe of X, where $u_j - u_{j-1} \equiv h, j = 2, 3, \cdots, m$. For $x_i \in X$, and $u_j \in U$, the following formula is called 1-dimensional linear-information distribution:

$$\mu(x_i,u_j)=\begin{cases}1-\left|x_i-u_j\right|/h & if\ \left|x_i-u_j\right|\le h\\ 0 & if\ \left|x_i-u_j\right|>h\end{cases} \quad (1)$$

Where h is called step length and μ is called linear distribution.

Obviously, μ satisfies all properties of an information distribution function.

For example, let $X=\{5,6\}$. What we want to do is to calculate their relative frequency between 3.4 and 8.2 in order to get the soft-histogram [11].That is $X\subset[3.4,8.6]$. Assume we would like to have three intervals between 3.4 and 8.2. That is $h=\frac{8.2-3.4}{3}=1.6,$ therefore we can get three intervals $[3.4,5)\cup[5,6.6)\cup[6.6,8.2).$

We chose u_i as the center of each intervals. Accordingly $u_1=(3.4+5)/2=4.2$, $u_2=(5+6.6)/2=5.8$ and $u_3=7.4$, Thus $U=\{4.2,5.8,7.4\}$. Given $x_1=5,x_2=6$, we can get:

$\mu(x_1,u_1)=1-|5-4.2|/1.6=0.5$, $\mu(x_1,u_2)=1-|5-5.8|/1.6=0.5$,

$\mu(x_1,u_3)=0$, $\mu(x_2,u_1)=0$, $\mu(x_2,u_2)=1-|6-5.8|/1.6=0.875$

$\mu(x_2,u_2)=1-|6-7.4|/1.6=0.125$, $\mu(x_2,u_3)=0$.

3.3 1-Dimension Linear Information decomposition

Let $X=\{x_i\mid i=1,2,\cdots,n\}$ be a given sample, R is the universe of discourse of X, $A=[a,b]$, where $a=min\{x_i\mid i=1,2,\cdots,n\}$ and $b=max\{x_i\mid i=1,2,\cdots,n\}$. t is the settled number of the intervals that $A=[a,b]$ is being divided, usually t is the number of missing values. That is the step length $h=(b-a)/t$, $A=\bigcup_{j=1}^{t}A_j$ and $A_j=[a+(j-1)*h,a+j*h]$. $U=\{u_1,u_2,\cdots,u_t\}$ is the discrete universe of R where $u_j-u_{j-1}\equiv h, j=2,3,\cdots,t$ and $u_j=(a+(j-1)*h+a+j*h)/2$, that is to say u_j is the center of A_j. For $\mu(x_i,u_j)$ is obtained from formula (1), $x_i\in X$, and $u_j\in U$, m_{ij} obtained from formula (2) is called 1-dimensional linear information decomposition from x_i to A_j.

$$m_{ij}=\mu(x_i,u_j)*x_i \quad (2)$$

Where h is called step length and μ is called linear distribution.

For example, $X=\{3.4,5,6,8.2\},t=3$, then we get $x_1=3.4,x_2=5,x_3=6,x_4=8.2$, $u_1=4.2$, $u_2=5.8,u_3=7.4$ $h=1.6$ and $A_1=[3.4,5),A_2=[5,6.6),A_3=[6.6,8.2)$.

Therefore, 1-dimensional linear information decomposition from x_2 to A_1 is:

$m_{21}=\mu(x_2,u_1)*x_2=(1-|5-4.2|/1.6)*5=2.5$

Similarly, we can get:

$m_{22}=\mu(x_2,u_2)*x_2=(1-|5-5.8|/1.6)*5=2.5$

$m_{31}=\mu(x_3,u_1)*x_3=0$

$m_{32}=\mu(x_3,u_2)*x_3=(1-|6-5.8|/1.6)*6=5.25$ etc..

4 1-DLID approach for missing data recovery

In the following discussion, the detailed steps about how 1-dimensional linear information decomposition method used in missing data recovery will be introduced. First of all, we would like to note that this paper focus on numerical missing data.

Let $X=\{x_i\mid i=1,2,\cdots,n\}$ be a missing data (incomplete data) set; the number of missing values is t, the missed values denoted as $\{m_k\mid k=1,2,\cdots,t\}$.

Let $a=min\{x_i\mid i=1,2,\cdots,n\}; b=max\{x_i\mid i=1,2,\cdots,n\}$. Then we get an interval $[a,b]$. Bear that if we let $c=a\pm0.5$ or $c=a\pm1$ and $d=b\pm0.5$ or $d=b\pm1$, we can get another interval that is $[c,d]$ and will help us get another recovery missing data, thus we can choose the average values of all the missing data.

Let $h=\frac{b-a}{t}$ $A_i=[a+(i-1)*h,a+i*h),i=1,2,\cdots,t.$

And we get $(a+(i-1)*h+a+i*h)/2$, then we find out the number of $x_i\in A_i$ and we get $\{x_i\}=A_i\cap X$.

To clarify, we denote $Y_i=\{y_l\mid l=1,2,\cdots,s\}=\{x_i\}=A_i\cap X$, then we get $\sum_{l=1}^{s}y_l/s$, $i=1,2,\cdots,t.$

In the 1-dimensional linear information decomposition approach, we choose the linear distribution as:

$$\mu(y_j,u_i)=\begin{cases}1-\left|y_j-u_i\right|/h & if\ \left|y_j-u_i\right|\le h\\ 0 & if\ \left|y_j-u_i\right|>h\end{cases}$$

Then we can calculate the following values:

$f_{A_i}(\tilde{y}_i,u_i)$, $f_{A_i}(\tilde{y}_{i+1},u_i)$ and $f_{A_i}(\tilde{y}_{i-1},u_i)$, finally we get the i^{th} missing data value, which is $m_i=(f_{A_i}(\tilde{y}_i,u_i)+f_{A_i}(\tilde{y}_{i+1},u_i)+f_{A_i}(\tilde{y}_{i-1},u_i))/3$. If one of them is 0, for example, $f_{A_i}(\tilde{y}_{i+1},u_i)=0$, we get $m_i=(f_{A_i}(\tilde{y}_i,u_i)+f_{A_i}(\tilde{y}_{i-1},u_i))/2$, once two of them are 0, for example, $f_{A_i}(\tilde{y}_{i+1},u_i)=f_{A_i}(\tilde{y}_{i-1},u_i)=0$, then $m_i=f_{A_i}(\tilde{y}_i,u_i)$.

The following steps are used to generate the missing values for the data set with missing values:

1. Given the incomplete data set X and the number of missing data values.
2. Compute A_i and u_i.

3. Compute $\tilde{y}_i, i = 1, 2, \cdots, t.$ Where t is the number of missing values.
4. For each i compute $f_{A_i}(\tilde{y}_i, u_i)$, $f_{A_i}(\tilde{y}_{i+1}, u_i)$ and $f_{A_i}(\tilde{y}_{i-1}, u_i)$.
5. Compute m_i , if $f_{A_i}(\tilde{y}_i, u_i) = f_{A_i}(\tilde{y}_{i+1}, u_i)$ $= f_{A_i}(\tilde{y}_{i-1}, u_i) = 0$ then $m_i = 0$, then $m_i = mean(\mathrm{X})$. Otherwise, $m_i = (f_{A_i}(\tilde{y}_i, u_i) + f_{A_i}(\tilde{y}_{i+1}, u_i) + f_{A_i}(\tilde{y}_{i-1}, u_i))/3$.

If one of them is 0, for example, $f_{A_i}(\tilde{y}_{i+1}, u_i) = 0$, we get $m_i = (f_{A_i}(\tilde{y}_i, u_i) + f_{A_i}(\tilde{y}_{i-1}, u_i))/2$, once two of them are 0, for example, $f_{A_i}(\tilde{y}_{i+1}, u_i) = f_{A_i}(\tilde{y}_{i-1}, u_i) = 0$, then $m_i = f_{A_i}(\tilde{y}_i, u_i)$.

The 1-dimensional linear information decomposition approach is easy to use; because it doesn't need any restrictions as long as we know the data set with missing values or incomplete data set and the number of missing values. In the following sections, the experiments and explanation will be discussed.

5 Experiments and results analyze with generated data set

5.1 Experimental Data

In order to make sure whether the proposed algorithm works well or not, we generate data from the Gaussian distribution and Gamma distribution. That is to say the generated data set $\{x_i\} \sim N(\mu, \sigma^2)$ or $\{x_{ij}\} \sim N(\mu, \sigma^2)$ and $\{x_i\} \sim \Gamma(\alpha, \beta)$ or $\{x_{ij}\} \sim \Gamma(\alpha, \beta)$ which can be regard as a matrix, either $1 \times n$ or $p \times n$. Then we get rid of some of the data values randomly with the help of computer (matlab). That is to say, we create the data set with missing values and ready for using in the experiments. With every data set with missing values we used increasing levels of 'missingness': 5%, 10%, 20%, 30% and 50%.

The reason why we choose Gaussian distribution and Gamma distribution is that because Gaussian distribution is widely used in research area, which can be regarded the data values is distribute averagely beside the means of the data set. However, not every data set that with missing values can follow Gaussian distribution in daily life. In order to show that 1-dimensional linear information decomposition method is good at processing any kind of data set rather than Gaussian distribution. We used data from Gamma distribution for experiment, because the property of data sets is totally different from each other. We use the following data sets:

Table 1: Information about the datasets used in this paper

Dataset Number	Instances	Data from $X \sim N(10, 5^2)$ Missing values (%)	Data from $X \sim \Gamma(10, 5)$ Missing values (%)
1	100	5%	
2	100	10%	
3	100	20%	
4	100	30%	
5	100	50%	
6	1000	5%	
7	1000	10%	
8	1000	20%	
9	1000	30%	
10	1000	50%	
11	1000		5%
12	1000		10%
13	1000		20%
14	1000		30%
15	1000		50%

In the following table the results arrived on a Window 8 laptop equipped with Core i7-2600 CPU at 3.40 GHz and 8.00 GB RAM is presented. And the matlab 7.0 is use for evaluation.

5.2 Experimental Strategy

Because 1-DLID method only require the condition of the incomplete data set and the number of missing values without any more information such as probability distribution or the incomplete data should meet the need of Bayesian estimation, mean/mode imputation method and listwise deletion method can be used in the data sets. However, in order to show 1-DLID approach works well and can achieve good results most of times. We would like to do experiment with one of most popular used algorithm which is EM algorithm. Overall, the experiments are based on four approaches that is 1-DLID method, Mean imputation method[6-9], listwise deletion method[6-9] and EM algorithm[6-9].

To ensure this is not the case, we performed the following for each experimental run:

1). Generate a data set with matlab 7.0 and save it into a file. We would like to generate a $1 \times n$ data set. And we denote this data set as F_t.

2). We get rid a certain percentage of the data from F_t, and we get the missing data set X , which we mentioned before.

3). Then we come to the proposed steps in section 3.

5.3 Evaluation Criteria

Before the results were presented, we would like to give a brief explanation of the errors of all the parameters. First of all the predicted parameters are calculated by the complete data, which is the total of data set with missing values and the recovered data. Then we compare the predicted parameters and the original ones and give the following definition:

μ error is defined as $|\tilde{\mu} - 10|/10$, where $\tilde{\mu}$ is the predicted parameter.

σ error is defined as $|\tilde{\sigma} - 5|/5$, where $\tilde{\sigma}$ is the predicted parameter.

α error is defined as $|\tilde{\alpha}-10|/10$, where $\tilde{\alpha}$ is the predicted parameter.

β error is defined as $|\tilde{\beta}-5|/5$, where $\tilde{\beta}$ is the predicted parameter.

5.4 Results

After choosing different intervals $[a,b]$ or $[c,d]$ etc., choosing of a good interval is very important, not only it helps to achieve a good results but also save time. Most times the intervals are chosen as $a=min(X)\pm 0.5$, $b=max(X)\pm 0.5$. However, sometimes are chosen as $a=min(X)\pm 1$, $b=max(X)\pm 1$ or others. Because it is difficult to choose a perfect interval, we will discuss in our future papers. In case we can get better results, we chosen three different intervals and use the average results, which can be regarded better than only choose one interval. After the experiments, we got the following results:

Table 2: Comparison of the results of dataset $X \sim N(10,5^2)$

	μ					σ				
Dataset number	Original	Proposed method	Deletion method	Mean method	EM algorithm	Original	Proposed method	Deletion method	Mean method	EM algorithm
1	10	9.9718	9.9292	9.9292	9.8325	5	5.0851	5.1300	4.9987	5.1056
2	10	10.1883	10.2293	10.2293	10.3433	5	5.1695	5.2450	4.9730	5.1308
3	10	10.1675	10.2267	10.2267	10.2139	5	5.0831	5.1697	4.8706	5.1101
4	10	10.0391	9.9273	9.9273	9.8525	5	4.9853	4.7080	4.7080	5.1760
5	10	10.5420	10.5867	10.5867	10.9646	5	5.0015	5.0551	4.6068	5.3577
6	10	9.8100	9.7690	9.7690	9.8215	5	4.7211	4.6920	4.5731	4.9697
7	10	9.8499	9.8372	9.8372	10.2400	5	4.7624	4.7619	4.5173	5.5702
8	10	9.6138	9.7825	9.7825	10.1881	5	4.7439	4.6884	4.1929	5.2286
9	10	9.4849	9.7904	9.7904	10.6379	5	4.3620	4.6613	3.8991	5.8313
10	10	9.3260	9.8733	9.8733	11.7081	5	4.1955	4.7951	3.3889	6.8376

Table 3: Comparison of the results of the dataset $X \sim \Gamma(10,5)$

	α					β				
Dataset number	Original	Proposed method	Deletion method	Mean method	EM algorithm	Original	Proposed method	Deletion method	Mean method	EM algorithm
11	10	9.4530	9.6000	10.0968	9.6469	5	5.4397	5.3265	5.0644	5.2971
12	10	9.9587	9.8964	10.9781	9.7294	5	5.1642	5.1517	4.6619	5.2484
13	10	10.0140	9.9374	13.3813	9.9384	5	5.1065	5.1647	4.1452	5.1584
14	10	10.1487	9.3651	13.3096	9.5682	5	5.0231	5.5433	3.9005	5.3870
15	10	9.7572	9.4903	18.8185	10.3844	5	4.9736	5.3379	2.6919	4.8208

In order to make the results clear to understand, we have presented them in the following pictures:

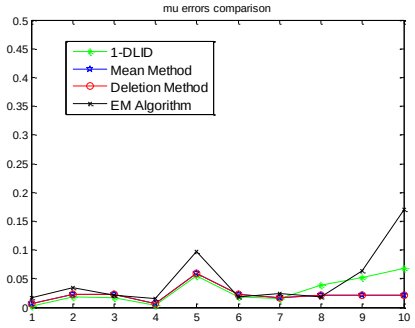


Fig. 1: μ error comparison of each method from data 1 to data 10 of $X \sim N(10,5^2)$

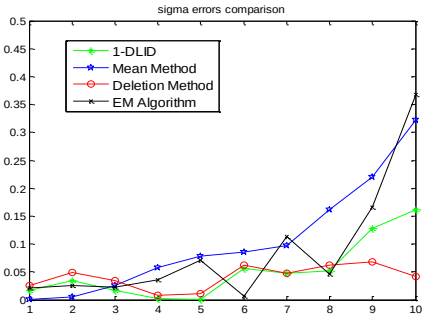


Fig. 2: σ error comparison of each method from data 1 to data 10 of $X \sim N(10,5^2)$

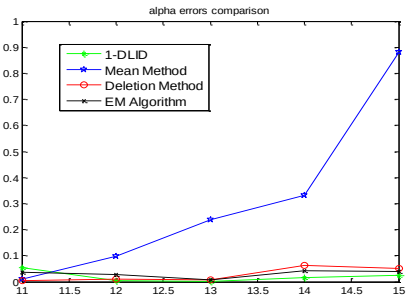


Fig. 3: α error comparison from data 10 to data 15 of $X \sim \Gamma(10,5)$

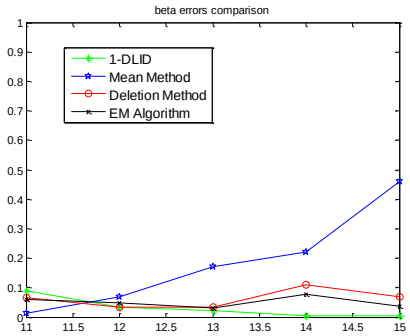


Fig. 4: β error comparison of each method from data 10 to data 15 of $X \sim \Gamma(10,5)$

As can be seen from figure 1, 1-DLID approach works well especially for dataset from 1 to 7, even when there are 50% of data lost, the μ error is still within 10% which is smaller than EM method. For the reason that delete method and mean imputation method achieved better results for μ from dataset 8 to dataset 10, we think the normal distribution dataset made them works well towards this. However, the results predicted by 1-DLID method are acceptable. Moreover, the σ errors showed in figure 2 also proved that 1-DLID approach perform better than the other three methods, especially from dataset 1 to dataset 8. While it is illustrated that the 1-DLID method also achieves a better results compared with EM method and mean imputation method. Again because normal distribution data set that delete method works well even though 50% of data values lost. And

this is can be seen from $X \sim \Gamma(10,5)$ datasets. Figure 3 told us that though 1-DLID method may not achieve a good result towards α when there are 5% data lost, it indeed works very well on any other datasets except dataset 1. We have to say, this error is acceptable, because it is only about 5.43%. Similarly, the trend in figure 4 seems the same as figure 3 towards the errors of β. Figure 4 described that 1-DLID approach presents a much better results which the errors decreased from 8.79% to 0.53% gradually while the errors of other methods are bigger than 1-DLID method, particularly, take the mean imputation method for example, its error reached nearly 50% when there are 50% data lost, which is unacceptable.

6 Implementation and evaluation with real-world data set

To evaluate the proposed method, a suitable and standard data set is needed. In this paper, the data set from Wisconsin Diagnostic Breast Cancer (WDBC) was chosen for our experiments. The original data set was provided by Dr.Willian H. Wolberg, W. Nick Street and Olvi L. Mangasarian of University of Wisconsin. We chose this kind of dataset because:

1. It has sufficiently large number of attributes and records, which is not only make sense for this paper, but also helpful for our future experiments, which will based on large numbers of attributes.
2. Except the class attributes, all the other data are numerical data, which suit for the proposed approach.
3. The dataset is from the UCI website, which is reliable and can be downloaded.

Information about the dataset:
Number of the instances: 198
Number of attributes:34 input real-valued features(ID, outcome, 32 real-valued input features). Based on our method is good at processing one-dimensional data set, we randomly chose one attributes for our experiments.
Experiment attribute feature: field 4, which is the Mean Radius of the cell nucleus.
Experiment data Missing attribute values: None
Missing attribute values of the whole data: Lymph node status is missing in 4 cases.

The mean, standard error, and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 4 is Mean Radius, field 14 is Radius SE, and field 24 is Worst Radius.

The chosen dataset is selected for evaluating four missing data imputation approaches because it is suitable for doing experiment on computer for the Delete method (listwise deletion), Mean imputation method, EM imputation and 1-DLID approach. While Filling Manually and Hot Decking Imputation is too time consuming and is not good at processing large number data in data mining. Multiple Imputation works well in high-dimensional data set, and we will do experiment based on MI method in our future experiment once we explored our method works in multi-dimensional data set.

Missing data were deleted randomly by the computer, and then recovered with the help of the four method separately and then compared the index of cluster: Rank index and Silhouette index.

The following tables and figures show the performance of the proposed method and other approaches. Precisely, the performance for the reconstructing of the WDBC dataset is based on the performance of the classification measures. That is to say, the higher rate or the better classification performance means better imputation of the missing values.

Table 4: Results of Rand Index

Missing values (%)	Clustering Accuracy			
	Listwise deletion	Mean imputation	EM imputation	Proposed method
10	64.34%	66.99%	65.26%	63.61%
20	62.58%	68.2%	61.56%	85.07%
30	58.11%	72.76%	56.39%	82.56%
40	61.15%	73.46%	81.57%	85.93%
50	66.23%	80.94%	68.2%	92.21%

The following figure 5 illustrates the performance of the methods (in terms of Rand Index) towards different percentage of missing values. It can be seen from the chart that the proposed 1-DLID method shows a better results than the other three methods.

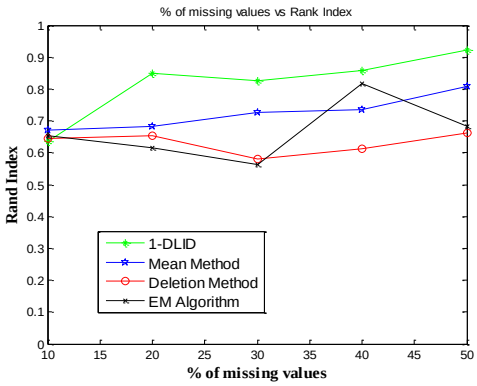


Fig. 5: Percentage of Missing Values vs. Rank Index

The following table, the results in terms of Silhouette index. The higher rate or better classification performance means the better imputation of missing values.

Table 5: Results of Silhouette index

Missing values (%)	Clustering Accuracy			
	Listwise deletion	Mean imputation	EM imputation	Proposed method
10	74.58%	76.75%	74.96%	75.39%
20	75.05%	79.01%	76.8%	89.75%
30	73.8%	82.51%	75.31%	87.17%
40	70.85%	80.76%	85.74%	90.21%
50	75.85%	87.1%	77.8%	94.76%

The following figure 6 illustrates the performance of the methods (in terms of Silhouette index) towards different percentage of missing values. It can be seen from the chart that the proposed 1-DLID method shows a better results than the other three methods.

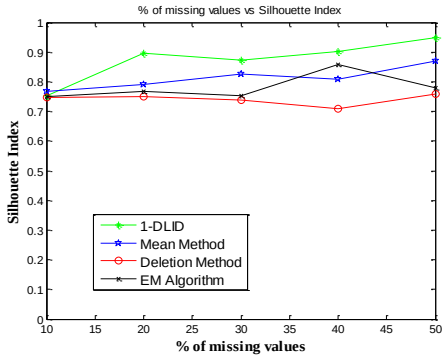


Fig. 6: Percentage of Missing Values vs. Silhouette index

Just as we discussed in the first section, delete method (or listwise deletion) and mean imputation method usually cause bias when the data is not normal distribution. For example, several noise data or leverage data can change the mean of the data set, and delete some data usually causes some important information got lost. For EM algorithm, it may converge to a local maximum of the observed data likelihood function, and this depending on starting values. However, 1-DLID approach doesn't have such problems. It can make good use of the existing data values. One problem is that how to choose a good interval for the 1-DLID method is very important, because it can reflect the recovery data set. We will do more research and talk this in one particular paper.

7 Conclusions

This paper presented an 1-DLID approach for missing data recovery. In the experiments, data are generated based on Gaussian distribution and Gamma distribution while the missing data is created by computer, that is to say the missing data were chosen randomly from computer and removed them from the related complete data set to get the test data sets we need. Regarding to the 1-dimensional data set, we compare our method with deletion method (or listwise deletion), mean imputation method and EM algorithm and compare our results with the other approaches. The experimental results showed that our approach has a precise results, especially when missing values between 10% and 30%. More importantly, the proposed method is easy to use. If needed researchers can use the 1-DLID algorithm several times based on different interval, and then choose the average values of each data, the results would be improved and more reliable. From the generated experiments, we can see that the proposed approach does not change the distribution of the data set. And from the real-world data set, a higher accuracy is achieved compared with the other methods.

8 Future Works

This paper has presented an 1-DLID approach which is used for data recovery for the analysis of incomplete data set. Like most method, 1-DLID method can create data for the data set with missing values while 1-DLID method do not need to provide the estimation distribution, which is easier to use. However, this method works well in 1-dimensional data set. Our future work is to do more exploration and make it work in 2-dimensional data set and then multidimensional data set and compare with MI algorithm. What is more, how to choose a proper interval that used for data recovery is of vital important, and it becomes one of the problems that we should overcome in the future. Lastly, we would like to develop it into a sophisticated software which will contribute to the society and help people recover the missing data.

9 References

[1] A. Ragab, S. Yacout, Ouali and S. Mohamed, Intelligent Data Mining For Automatic Face Recognition, Turkish Online Journal of Science & Technology, Apr. 2013, Vol. 3, Issue 2, pp. 92-96.

[2] K. Lokanayaki and A. Malathi, Exploring on Various Prediction Model in Data Mining Techniques for Disease Diagnosis, International Journal of Computer Applications, Sep. 2013, Vol. 77, pp. 26-29.

[3] R. Somasundaram and R. Nedunchezhian, " Evaluation on Three Simple Imputation Methods for Enhancing Preprocessing of Data with Missing Values", International Journal of Computer Applications, Vol 21, No. 10, May 2011, pp. 14-19.

[4] J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2006.

[5] R. Little and D. Rubin. Statistical Analysis with Missing Data, Wiley Series in Probability and Statistics, 1987.

[6] E. Foster and G. Fang: Alternative methods for handling attrition: an illustration using data from the Fast Track evaluation, Eval Rev 2004, Vol.28, pp. 434–464.

[7] S. Lynch. Missing data (Soc 504), Princeton University Sociology 504 Class Notes, 2003.

[8] P. Allison: Missing data. Thousand Oaks, CA: Sage; 2000.

[9] A.P.Dempster, N.M. Laird, and D.B. Rubin, Maximum-likelihood from incomplete data via the EM algorithm. J. Royal Statist. Soc. Ser. B., 39, 1977.

[10] C.F.J. Wu. On the convergence properties of the EM algorithm, The Annals of Statistics 11, pp. 95–103, 1983.

[11] C.F. Huang, Demonstration of benefit of information distribution for probability estimation, Signal Processing 80.6, pp.1037-1048, 2000.

Predicting Causes of Traffic Road Accidents Using Multi-class Support Vector Machines

Elfadil A. Mohamed

Department of Management Information Systems, College of Business Administration, Al Ain University of Science and Technology, Al Ain, United Arab Emirates

Abstract *- Road traffic accidents have caused a myriad of problems for many countries, ranging from untimely loss of loved ones to disability and disruption of work. In many cases, when a road traffic accident occurs that results in the death of both drivers of the vehicles involved in the accident, there are some difficulties in identifying the cause of the accident and the driver who committed the accident. There is a need for methods to identify the cause of road traffic accidents in the absence of eyewitnesses or when there is a dispute between those who are involved in the accident. This paper attempts to predict the causes of road accidents based on real data collected from the police department in Dubai, United Arab Emirates. Data mining techniques were used to predict the causes of road accidents. Results obtained have shown that the model can predict the cause of road accidents with accuracy greater than 75%.*

Keywords—component; road traffic accident, data mining, multi-class SVMs

1 Introduction

Road traffic accidents (RTAs) are currently ranked ninth globally among the leading causes of disability-adjusted life years lost and the ranking is projected to rise to third by 2020 [1]. A study conducted by Bener et al. [2] indicates that road traffic fatalities are second only to cardiovascular disease in the list of major causes of death. About 90% of the disability-adjusted life years lost worldwide due to road traffic injuries occur in developing countries. In recent years, high rates of serious RTAs have been reported in several Arabian Gulf countries, including the United Arab Emirates. UAE is a young, wealthy country with a number of vehicles on the road that is continuously increasing. The rate of RTAs is relatively high in the UAE and generally causes more serious trauma than other accidents, which is reflected in a high number of fatal and serious injuries.

RTA prediction is an essential problem in traffic safety control. It is acknowledged that the success of traffic safety and highway improvement programs hinges on the analysis of accurate and reliable traffic accident data. For the successful completion of traffic safety controls, robust computational methods for predicting RTA are seriously needed; therefore, the subject was intensively studied by researchers around the globe. RTAs that lead to the death of all on board the vehicles involved in the accident leave the traffic police without eyewitnesses to question in order to determine the cause of the accident. Even when an accident does not result in death, there might be disputes between those who are involved in the accident to know who is the victim and who is the offender. The police department might experience some difficulties in identifying the real offender. Methods are needed to predict the cause of the accident and the offender in RTAs. On the other hand, preventing accidents from happening is a major challenge. Computation and data mining methods are greatly needed to predict the possible causes of RTAs.

The main objective of this study is to design effective data mining methods to investigate and predict the cause of RTAs in one of the Gulf countries; namely, United Arab Emirates (UAE). Real data were obtained from the Dubai police department and were used for building a multi-class support vector machine for predicting the possible cause of TRAs. The remainder of the paper is organized as follows: Section II discusses the literature related to the prediction of number of RTAs and the forecasting of the cause thereof. Section III explains the methodology used. Section IV discusses the experimental work and results. Finally, Section V provides a conclusion of the work.

2 Related Works

Most of the literature related to RTAs is centered on the prediction of the number of road accidents; for example, Huilin and Yucai [3] used neural networks for the prediction of traffic accidents. Yisheng et al. [4] have investigated the use of the k-nearest neighbor method for identifying the more likely traffic conditions leading to traffic accidents, while considering the joint effects of accident precursors on traffic accident occurrences and controlling the geometry and environmental factors. For forecasting of RTA, Qing-wei et al. [6] used a method that combine support vector regression (SVR) and particle swarm optimization (PSO). The experimental results indicated that the proposed PSO-SVR method has better performance accuracy than back propagation neural network in traffic accident forecasting. Mathematical models have been used for the estimation of the number of RTAs. A novel composite grey back-propagation neural network model was proposed by Zhu [7] for the estimation of the number of RTAs. The proposed model

showed an improvement in the forecasting accuracy of the number of RTAs.

With the availability of data in an electronic form, data mining techniques have widely been used in road traffic accident analyses. Classification and clustering techniques were used in [8] for the prediction of traffic incidents. Spatial data mining for the analysis of traffic accidents is introduced in [9]. Recent study that addresses issues related to the use of data mining techniques for predicting the likelihood occurrence of road traffic accident on highways, the likely cause of the accident, and accident-prone locations can be found in [5].

The development of data mining models that predict the number and cause of RTAs has been studied extensively (see, for example, [3, 6, 7] for predicting the number of RTAs and [5] for predicting the cause of RTAs). However, most of the above-mentioned data-mining models suffer from low prediction accuracy of the prediction and, in most of the cases, it is due to a poor pre-processing step. New data mining methods are needed with the aim of improving the accuracy of predictions through the understanding of the fuzziness of the datasets, solid pre-processing (handling of missing entries, unbalance data issues, wrong entries, evaluating attributes related to RTAs, etc.), and usage of powerful machine learning techniques such as support vector machines (SVMs).

Research work studying and predicting the cause of RTAs in the Gulf region is quite limited. In Saudi Arabia, for instance, Ageli and Zaidan [10] used an autoregressive distributed lag ADRL model for the analysis of RTAs. Bener et al. [1] explored the pattern of RTAs and their causes in the state of Qatar. Bener and Crundal [2] have investigated data concerning RTAs and road user behavior in UAE. However, no published research work focuses on studying and addressing issues related to the prediction of either the number of RTAs or the cause of RTAs in the UAE.

Data pre-processing is an important step in data mining. The main purpose of data preprocessing is to improve the quality of the data, which leads to the improvement of the mining results. According to Han et al. [11], there are several data pre-processing techniques. Data cleaning can be applied to remove noise and correct inconsistencies in data. Data integration merges data from multiple sources into a coherent data store such as a data warehouse. Data reduction can reduce data size by, for instance, aggregating, eliminating redundant features, or clustering. Finally, data transformation may be applied, where data are scaled to fall within a smaller range. Researchers have long recognized the importance of data pre-processing; for example, Kotsintis et al. [12]'s work address issues of data pre-processing that can have significant impact on the generalization performance of machine learning algorithms. For improving the efficiency of data preprocessing, Chen and Liu [14] proposed an improved data cleaning method.

Missing values can be handled by either ignoring the record, using global constants to fill the missing value, using measures of central tendency for the attributes, or by using the most probable value to fill in the missing values [11]. Kumar and Kalia [13] used the average value to fill the missing values. For the handling of missing values, Higashijima et al. [15] proposed the use of a regression tree imputation method. The proposed method achieved high accuracy compared to the not-pre-processed and linear interpolation method. In this paper, several valid pre-processing methods are used to improve the quality of RTA data before the development of the multi-class SVM step.

3 Method

In this case, our solution will follow the typical data-mining framework, which consists of three main steps: preparing the data (pre-processing), mining patterns, and post-processing. These steps will be described in the following sections. Figure 1 illustrates the main components of the method we used in this research.

3.1 Preparing and preprocessing the accident data

3.1.1 Data collection:

The data set used in this study is collected and retrieved from the Dubai Police Department, UAE. The traffic police department gathers and records the accident data using a traffic accident information system. The data consists of 7,048 entries and seven different attributes related to the drivers (age, nationality, and license) and the vehicles (type, year of make, etc.) involved in the accidents besides the location where the accidents took place.

The seven attributes used are locations (914 different locations), DEGINJ (4 different types), gender (M/F), age (67 different ages), country the driver belongs to (31 different countries), vehicle type (9 different types), and year the vehicle is made (33 different years of made).

3.1.2 Data preprocessing:

The issue of data quality has a direct relevance on the quality of the data mining results. Although almost everyone accepts the importance of data quality (please refer to section II), in reality, it is not always rigorously controlled. Traffic data is no exception and it suffers from unknown or missing entries, consistency, completeness, redundancy, etc. In this particular case, our data has missing entries with respect to the car's manufacturing year and the driver's gender and should be handled before the development of the data mining model. To handle the missing values, we adopted the "ReplaceMissingValues" method available in Weka machine learning software, which uses modes and means to identify the missing entries.

To ensure that all our attributes are meaningful, a feature selection is used to assess the relevance of each attribute. In this case, we focus on using a feature selection method that is based on filtering. Filtering algorithms use independent search and evaluation methods to determine the relevance of features variables to the data-mining task. In this case, we used the

"GainRatioAttributeEval" method available in Weka to evaluate the worth of an attribute by measuring the gain ratio with respect to the class.

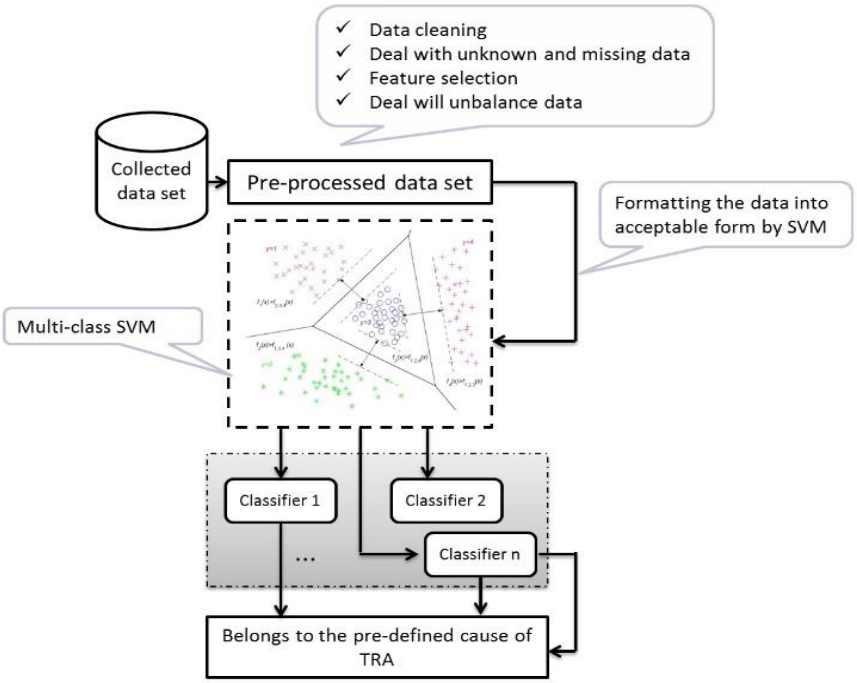


Fig. 1. The framework of predicting the cause of road traffic accident using multi-class SVMs.

3.1.3 Mining patterns

Once the data is pre-processed, a sensible data-mining task must be designed to comply with the objectives of predicting the 21 causes of road accidents. This problem can be handled by utilizing a multi-classification technique; therefore, Support Vector Machines (SVM) [20, 21] was selected. The idea of the SVM algorithm is to map the given training set into a possibly high-dimensional feature space and attempting to locate in that space a hyperplane that maximizes the distance separating the positive from the negative examples [16, 17].

The SVM algorithm addresses the general problem of learning to discriminate between positive and negative examples of a given class of n-dimensional vectors [18, 22]. In order to discriminate among the 21 causes of road accidents, the SVM learns a classification function from a set of positive

Examples $\mu+$ and set of negative examples $\mu-$. The classification function takes the form:

$$f(x) = \sum_{i:x_i \in \mu+} \lambda_i K(x, x_i) - \sum_{i:x_i \in \mu-} \lambda_i K(x, x_i) \quad (1)$$

where the non-negative weights λ_i are computed during training by maximizing a quadratic objective function and the function $K(x, x_i)$ is called a kernel function. Any accident case x is then predicted to be positive if the function $f(x)$ is positive. More details about how the weights λ_i are computed and the theory of SVM can be found in [16, 17].

3.1.4 Post-processing patterns

Following the classification step, it is important to evaluate the patterns detected by the SVM. Several evaluation measures are used in this study, such as Precision (Pr = $\frac{TP}{TP+FP}$), Recall (Re = $\frac{TP}{TP+FN}$), F1 measure ($2*\frac{\Pr \cdot \mathrm{Re}}{\Pr + \mathrm{Re}}$) and Accuracy (Ac = $\frac{TP+TN}{All}$, where TP, TN, FP, FN, and All are defined as:

- TP: related cause of road accident classified as "related."
- TN: unrelated cause of road accident classified as "unrelated."
- FP: related cause of road accident classified as "unrelated."
- FN: unrelated cause of road accident classified as "related."
- All: total number of causes of road accidents.

4 Experimental Work and Results

The experimental work began with the exploration and the preparation of the dataset. Several missing entries were found, particularly under the car's manufacturing year (2.23%) and gender (18.25%) attributes. The ReplaceMissingValues method was applied without referring to a particular class. Once the missing data entries were handled, the seven attributes were analyzed and the GainRatioAttributeEval method reveals that the location, vehicle type, and the driver's country have a strong relationship with the cause of the accident. Similarly, there is no evidence suggesting that gender has a link to the cause of accidents. Details of the attribute evaluation are summarized in Table 1.

TABLE 1

ATTRIBUTE EVALUATION

Attribute	Rank	Gain ratio
Location	1	0.1932
Vehicle type	2	0.0704
Country the driver belongs to	3	0.0494
Age	4	0.0353
DEGINJ (level of injury)	5	0.0317
Year the vehicle was made	6	0.0172
Gender	7	0

One other observation inferred from the data exploration is the fact that the data is unbalanced. The distribution of causes of accidents is shown in Figure 2. It is quite obvious to see that most of the accidents took place in the UAE due to an absence of attention/consideration for other drivers or excessively speeding.

From a data-mining point of view, this data requires balancing; therefore, a resampling method with a random seed equal to 1 was used. The resampling in this case produces a random subsample of a dataset using replacements.

Once the preprocessing step is completed and the dataset is prepared, a multiclass SVM was used. The Library for Support Vector Machines (LibSVM) implemented by Chih-Chung Chang and Chih-Jen Lin [23] was used. One of the significant parameters needed to tune the SVM is the choice of the kernel function. The kernel function allows the SVM to locate the hyperplane in a highly dimensional space that effectively separates the training data [16, 17]. The Gaussian Radial Basis function is used as it allows pockets of data to be classified, which is more powerful than just using a linear dot product [16].

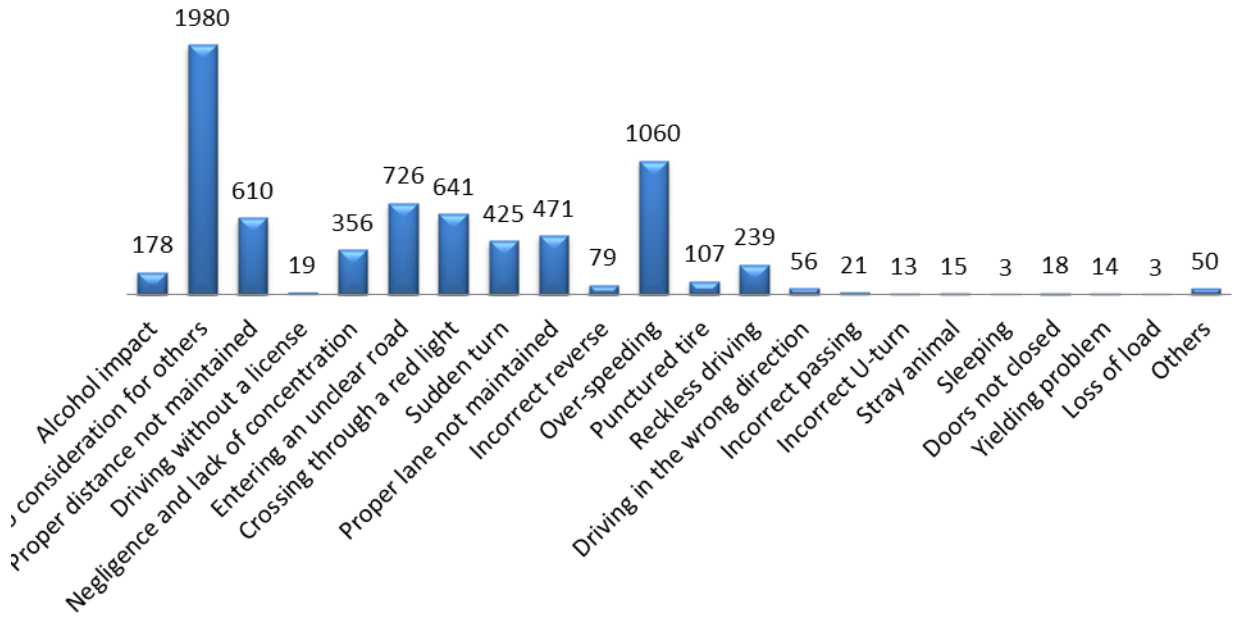


Fig. 2. Distribution of the causes of accidents in the UAE

Using 10-fold cross-validation, the detailed classification results are shown in Table 2. The overall Precision, Recall, F1 measure, and Accuracy are 0.767, 0.754, 0.752 and 75.395, respectively. Apart from the "Stray animal," "Yielding problem" and "Fall of the load" causes, the remaining 18 causes were detected with high accuracy. The three mentioned causes were had a very limited number of samples—15, 14, and 3, respectively.

TABLE 2

CLASSIFICATION DETAILS OF CAUSE OF ACCIDENTS

Cause of accident	Precision	Recall	F1-Measure
Alcohol impact	0.807	0.669	0.731
No consideration for others	0.693	0.896	0.782
Proper distance not maintained	0.718	0.636	0.674
Driving without a license	0.848	0.824	0.836
Negligence and lack of concentration	0.8	0.682	0.736
Entering an unclear road	0.858	0.749	0.799
Crossing through a red light	0.899	0.762	0.825
Sudden turn	0.84	0.642	0.728
Proper lane not maintained	0.781	0.647	0.708
Incorrect reverse	0.839	0.635	0.723
Over-speeding	0.687	0.77	0.726
Punctured tire	0.724	0.598	0.655
Reckless driving	0.901	0.628	0.74
Driving in the wrong direction	0.977	0.764	0.857
Incorrect passing	0.923	0.75	0.828
Incorrect U-turn	1	0.81	0.895
Stray animal	0	0	0
Sleeping	0.8	1	0.889
Doors not closed	0.5	0.5	0.5
Yielding problem	0.5	0.3	0.375
Loss of load	0	0	0
Others	0.722	0.433	0.542
Average	**0.767**	**0.754**	**0.752**

5 Conclusions

In this paper, a multi-class support vector machine model was developed to enable the prediction of the cause of RTAs in UAE. Several preprocessing methods are used to improve the quality of the traffic data. The accuracy achieved by the developed model is approximately 75%, which is quite acceptable despite the fact that further accuracy improvements are required. The developed model can be used by the UAE traffic police department as a tool for predicting the future causes of RTAs, as well as the offending driver, in the case of the absence of eyewitnesses or when there is a dispute between those who are involved in the accident. The model can assist in avoiding accidents. The good accuracy achieved in this study suggested that there are strong combinations (patterns) of attributes that could lead to possible common causes of accidents. Taking these combinations in consideration, the traffic authorities could communicate and warn drivers to be more alert. Moreover, the analysis of the RTA data has revealed that the dominant causes of RTA in the UAE are typically due to neglecting other vehicles on the road or over-speeding. These causes of RTA are more related to the drivers' behavior. Traffic police authorities could conduct campaigns and awareness sessions to educate drivers in how to avoid these two causes of RTAs. Other methods of controlling over-speeding should be implemented. In the current study, only seven features/attributes were used. In the future, more valuable features describing the cause of RTA should be collected and analyzed.

6 Acknowledgment

The author would like to thank Mr. Omar Alnakbi, an employee at Roods & Transport Authority, Dubai, for his effort in providing the data which were used for testing the model.

7 References

[1] BENER, "The neglected epidemic: Road traffic accidents in a developing country, State of Qatar," International Journal of Injury Control and Safety Promotion, Vol. 12, No. 1, March 2005, pp. 45 – 47.

[2] Bener and D. Crundall, "Road traffic accidents in the United Arab Emirates compared to Western countries," Advances in Transportation Studies in an international Journal Section A 6, 2005.

[3] F. Huilin and Z. Yucai, "The Traffic Accident Prediction Based on Neural Network," Second International Conference on Digital Manufacturing & Automation, 2011.

[4] L. Yisheng, T. Shuming , and Z. Hongxia, "Real-time Highway Traffic Accident Prediction Based on the k-Nearest Neighbor Method," International Conference on Measuring Technology and Mechatronics Automation, 2009.

[5] T. Akomolafe and A. Olutayo, "Using Data Mining Technique to Predict Cause of Accident and Accident Prone Locations on Highways," American Journal of Database Theory and Application 2012, 1(3): 26-38.

[6] Z. Qing-wei, F. Ai-Ying, and X. Zhi-Hai, "Application of Support Vector Regression and Particle Swarm Optimization inTraffic Accident Forecasting," International Conference on Information Management, Innovation Management and Industrial Engineering, 2009.

[7] X. Zhu, "Application of Composite Grey BP Neural Network Forecasting Model to Motor Vehicle Fatality Risk," Second International Conference on Computer Modeling and Simulation, 2010.

[8] Pan, U. Demiryurek, C. Shahabi, and C. Gupta, "Forecasting Spatiotemporal Impact of Traffic Incidents on Road Networks," IEEE 13th International Conference on Data Mining, 2013.

[9] W. Jinlin, C. Xi, Z. Kefa, W. Wei, and Z. Dan, "Application of Spatial Data Mining in Accident Analysis System," International Workshop on Education Technology and Training & 2008 International Workshop on Geoscience and Remote Sensing, 2008.

[10] M. Ageli and A. Zaidan, "Road Traffic Accidents in Saudi Arabia: An ADRL Approach and Multivariate Granger Causality," International Journal of Economics and Finance, Vol. 5, No. 7, 2013.

[11] J. Han, M. Kamber, and J. Pei, Data Mining Concepts and Techniques. Morgan Kaufmann: Waltham, MA 02451, USA, 2012.

[12] S. B. Kotsiantis, D. Kanellopoulos, and P. E. Pintelas, " Data Preprocessing for Supervised Leaning," International Journal of Computer Science, Volume 1, No. 2, 2006.

[13] M. Kumar and A. Kalia, "Preprocessing and Symbolic Representation of Stock Data," Second International Conference on Advanced Computing & Communication Technologies, 2012.

[14] X. Cheng and H. Liu, "Research on Data Preprocessing Technology in Safety Equipment Linkage System," International Conference on Computational and Information Sciences, 2013.

[15] Y. Higashijima, A. Yamamoto, T. Nakamura, M. Nakamura, and M. Matsuo, "Missing Data Imputation using Regression Tree Model for Sparse Data collected viaWide Area Ubiquitous Network," 10th Annual International Symposium on Applications and the Internet, 2010.

[16] N. M . Zaki, S. Deris, and R. Illias, "Feature Extraction for Protein Homology Detection using Hidden Markov Model combining Scores," International J. of Computational Intelligence and Applications. Vol. 4, pp: 1-12, 2004.

[17] A. Abdalla, S. Deris, and N. M. Zaki, "Breast Cancer Detection Based on Statistical Textural Features Classification," International Conference on Innovations in Information Technology, 728-730, UAE, 2007.

[18] N. M. Zaki, and W. El-Hajj, "Predicting Membrane Proteins Type Using Inter-domain Linker Knowledge," The 2010 International Conference on Bioinformatics and Computational Biology (BIOCOMP'10), 12-15 July 2010, Las Vegas, US.

[19] N. M. Zaki, S. Deris, and H. Alashwal, "Protein-Protein Interaction Detection Based on Substring Sensitivity Measure," International J. of Biomedical Sciences, Vol. 1, pp: 148-154, 2006.

[20] V. N. Vapnik, Statistical Learning Theory. Wiley, 1998.

[21] N. Cristianini, and J. Shawe-Taylor, An introduction to Support Vector Machines. Cambridge, UK: CambridgeUniversity Press, 2000.

[22] N. M. Zaki, S. Deris, and R. M. Illias, "Simple Representation of Protein Sequence for detecting homology," International Conference on Artificial Intelligence. Las Vegas, USA, 27-30 June 2005.

[23] R. E. Fan, P. H. Chen, and C. J. Lin, "Working set selection using second order information for training SVM," Journal of Machine Learning Research 6, 1889-1918, 2005.

Gold, Oil and the S&P 500 Index: Calm to Crisis and Back

A.G. Malliaris[1], **and M. Malliaris**[2]
[1]Economics & Finance Depts., Loyola University Chicago, Chicago, IL, USA
[21]Information Systems & Operations Management, Loyola University Chicago, Chicago, IL, USA

Abstract - *Numerous studies have analyzed the impact of gold and oil price movements on the S&P 500 Index. The theoretical motivation of studying gold, oil and equity markets is driven by the role of energy prices as a major contributing factor to inflation as measured by both the consumer and wholesale price index, and inflation by gold. Most existing studies do not confirm robust relationships among these three assets. In this paper we take a different methodological approach. We theorize that during the past 14 years, the U.S. economy has transitioned through four regimes: the pre-crisis calm period, the bubble period, the crisis period, and the return to average growth. For each regime we study the relationships among gold, oil and equity price changes and their lagged movements. We confirm that these relationships are regime dependent. This finding confirms the empirical difficulty in finding robust rules over a large sample.*

Keywords: Gold, Oil, S&P 500, Forecasting, Crisis

1 Introduction

Numerous studies have analyzed the impact of gold and oil price movements on the U.S. stock market as represented in the S&P 500 Index. The theoretical motivation of studying gold, oil and equity markets is driven by the role of energy prices as a major contributing factor to inflation as measured by both the consumer and wholesale price indices. Inflation in turn may be hedged by gold. These relationships are affected by the specifics of the samples used in empirical investigations. Most of the existing studies do not confirm robust relationships among these three assets.

In this paper we take a different methodological approach. We theorize that during the past 14 years, the U.S. economy has transitioned through four regimes: an initial period of calm, the pre-crisis booming period, the crisis period, and the return to average growth. For each regime we study, using decision rule analytics, the relationships among gold, oil and equity price changes and their lagged movements. We confirm that these relationships are regime dependent. This finding confirms the empirical difficulty in finding robust rules over large sample.

2 Literature Review

Barro and Misra [1] have studied the behavior of gold returns by developing a theoretical model motivated by the Lucas-tree model. Gold has been dominant in the financial markets during the past several centuries and is usually regarded as a hedge against inflation, political upheavals and physical disasters. Barro and Misra find from 1836 to 2011, the average real rate of price change for gold in the United States is 1.1% per year and the standard deviation is 13.1%, implying a one-standard-deviation confidence band for the mean of (0.1%, 2.1%). These authors also study the relationship of gold to other economic variables. In particular they find that the co-variances of gold's real rate of price change with consumption and GDP growth rates are small and statistically insignificantly different from zero. The implications of these negligible co-variances suggest that gold's expected real rate of return would be close to the risk-free rate, estimated to be around 1%. The volatility of the growth rate of real gold prices is small under the classical gold standard from 1880 to 1913 but high—comparable to that on stocks in other periods, including 1975 to 2011. Changes in the volatility of gold prices can be partially explained by the shifting role of gold in monetary policies and changes in expected inflation.

Baur and Lucey [2] also study the behavior of gold prices and distinguish between gold as a hedge versus gold as a safe haven. Gold as a hedge is defined as a security that is uncorrelated with stocks or bonds or inflation on average. This allows a typical investor to use gold as a hedge. Gold can also be used as a safe haven. A security is defined as a safe haven when it is uncorrelated with stocks or bonds or inflation in a market crash. These authors find that gold is a hedge against stocks on average and a safe haven in extreme stock market conditions.

Reboredo [3] asks a much more focused question than Baur and Lucey. Reborelo asks if gold is a hedge or safe haven against oil movements. Reboredo uses an approach based on copulas to analyze the dependence structure between these two markets. Using weekly data from from January 2000 to September 2011, the author obtains the following results:. First, there is positive and significant average dependence between gold and oil, which would indicate that gold cannot hedge against oil price movements. Second, there is tail independence between the two markets, indicating that gold can act as an effective safe haven against extreme oil price movements.

In a related study, Cinera, Gurdgievb and Luceyb [4] investigate the return relations between major asset classes

using data from both the US and the UK. The authors first examine time variation in conditional correlations to determine when these variables act as a hedge against each other. The authors provide evidence on whether the dependencies between the asset classes differ during extreme price movements by using quantile regressions. Their analysis provides evidence on whether these asset classes can be considered as safe havens for each other. A key finding is that gold can be regarded as a safe haven against exchange rates in both countries, highlighting its property as monetary currency. Sari, Hammoudeh and Soytas [5] introduce the euro as an exchange rate to study its impact on gold and oil markets. These authors find that investors may diversify at least a portion of their portfolio risk away by investing in precious metals, oil, and the euro.

Malliaris and Malliaris [6] investigate inter-relationships among the price behavior of oil, gold and the euro using time series and neural network methodologies. Traditionally gold is a leading indicator of future inflation. Both the demand and supply of oil as key global commodities are impacted by inflationary expectations and such expectations determine current spot prices. Inflation influences both short and long-term interest rates that in turn influence the value of the dollar measured in terms of the euro. Certain hypotheses are formulated in this paper and tested. The authors find that the markets for oil, gold and the euro are efficient but have limited inter-relationships among themselves.

Narayan [7] examines the long-run relationship between gold and oil for both spot and futures markets. The author draws on the conceptual framework that when oil prices rise, they create inflationary pressures, which encourage investments in gold as a hedge against inflation. Furthermore, this paper tests for the long-run relationship between gold and oil futures prices at different maturities and finds evidence of cointegration. This implies that investors use the gold market as a hedge against inflation and also, the oil market can be used to predict the gold market prices and vice versa, thus these two markets are jointly inefficient, at least for the sample period considered in this study.

Fan and Xu [8] focus on oil prices. They use endogenously determined break tests that allow for changes in both level and trend to identify four regimes. This paper has inspired the four regimes investigated in detail in our study.

3 Data

This study investigates patterns in movement of London Gold, Brent Oil, and the S&P 500 from January 2000 through January 2014. Data for the S&P 500 were downloaded from finance.yahoo.com, the London Gold prices were downloaded from www.lbma.org.uk, and Brent oil prices were obtained from www.eia.gov. After downloading, derived columns created for each series included a value standardized over the 14 year period, the percent change from yesterday to today, a 5 day moving average, the direction the series moved from yesterday to today, the number of times the series moved Up in the most recent 5 days, and concatenated directional strings of movement for 2, 3, 4, and 5 days. In addition, a column reflecting values for tomorrow was created, tomorrow's direction. An example of each of these values for the Brent Oil series is shown in Table 1.

The data was divided into 4 periods, denoted as Calm, Bubble, Crisis, and After. This selection of subperiods is taken from Fan and Xu [6] and is supported from numerous analyses of the Global Financial Crisis as reported by Evanoff, Kaufman and Malliaris [9]. For each period we built separate models to investigate what movements happened together and to attempt to understand what affects tomorrow's direction for each of the three series. The dates included in each series are shown in Table 2.

Table 1. Examples of Variables Names and Values for the Brent Oil Series

Variable Name	Sample Value
BStd	0.08349027
BPerChg	-0.73126143
B5DayMA	-0.01478882
Bdir	D
BNumDaysUp	2
B2Day	UD
B3Day	UUD
B4Day	DUUD
B5Day	DDUUD
BDirTp1	U

Table 2. Dates for each data set.

Set	Begins	Ends
Calm	1/4/2000	3/31/2005
Bubble	4/1/2005	12/31/2007
Crisis	1/2/2008	3/13/2009
After	3/16/2009	1/24/2014

4 Models And Results

The first step in investigating these series was to look simply at the movement of each series today using Association Analysis to investigate which, if any, series typically moved in the same broad patterns within a day. Next, we look at today's movements to see if rules appear predicting tomorrow's movement. Finally, we will use both movement and numeric data based on the three series in a decision tree to forecast tomorrow's direction for each series.

There were two possible outcomes for each series (Up or Down) per day. The Association Analysis model generates a set of rules that meet specific conditions. The models here were set to look for antecedents that occurred in at least 7 percent of the rows, and Consequents that were true at least 55% of the time when the Antecedent occurred. There were 6 possible

single series antecedents and 4 possible consequents per antecedent. For antecedents using two series, there were 12 possible combinations, each with 2 possible consequents. Of these possible rules, only 6 occurred in three of the periods over time. Table 3 shows all of the rules that occurred in at least three out of these four data periods. Notice that when a rule failed to apply to all four periods, they failed to work in either the Calm period or the Crisis period. But if they occurred in Calm, they also appeared in Bubble and After sets. If they did not appear in Calm, but began in Bubble, then they have continued on. The Table 3 rules can be read in the following way: "If the Antecedent is True, then the Consequent is also True at least [Confidence] percent of the time." The first rule would imply "If Brent Oil moved Up today, then Gold moved up today 62% of the time during the Bubble period, 58% of the time during the Crisis period, and 63% of the time in the After period. Four of the 6 rules have Brent Oil in the Antecedent. Only one rule has a consequent with the S&P series, and indication that the direction of the S&P is less likely to move with the other series in a regular pattern in the same day.

Table 3. Rules appearing in three of the data sets, and the associated confidence.

		Confidence			
Ante.	Conseq.	Calm	Bub.	Cris.	Aft.
BDir = U	GDir = U		62	58	63
GDir = D	BDir = D		55	59	58
BDir = U and SPDir = U	GDir = U		62	56	66
GDir = U	BDir = U	55	59		63
GDir = D and BDir = U	SPDir = U	56	56		59
SPDir = D and BDir = U	GDir = U	60	63		57

The next run of the Association Analysis algorithm used today's direction of Brent Oil, Gold, and the S&P 500 as possible antecedents, and tomorrow's direction of each series as possible consequents. This is a beginning attempt to see whether or not we can forecast future direction. There were 26 possible antecedents with one, two or three series represented, and 6 possible consequents, or 26*6 = 156 possible rules. The number of rules generated by association analysis methodology were 8 for the Calm period, 39 for the Bubble period, 40 during Crisis, and 32 in the After period. Reducing the results tables to rules that occurred in at least 2 periods, we see the resulting 15 rules in Table 4.

Notice that there is only one rule that repeats from the Calm period to the After period. So, directional forecasts using only knowledge of today's movements of these three series changed from the Calm to the later periods, and only one rule reappeared in the After period. Two rules carried over from the Bubble period to the Chaos period then disappeared. But, 10 pattern rules from the Bubble period are also active in the After period. That is, many relationships disappeared during the Crisis period only to return after the crisis diminished. There are two rules which appeared first in the Bubble period have continued in both of the following periods. These two longer-run rules are "If the S&P moved Up today, the Gold will move Up tomorrow, and if the S&P moved Down today then it will change direction and move Up tomorrow. These two rules had the highest confidence in the Bubble period, but occurred at least 55% of the time in all later periods. Notice also that all the rules that appeared in multiple periods had consequents referring to Up movements.

Table 4. Common Rules in Forecasting

Antec	Conseq	Calm	Bub	Cris	Aft
BDir = D and GDir = D and SPDir = U	BDirTp1 = U	58			59
BDir = U and GDir = U	BDirTp1 = U		56	56	
GDir = U and SPDir = U	BDirTp1 = U		56		58
SPDir = D and GDir = D and BDir = U	BDirTp1 = U		58		57
SPDir = U	BDirTp1 = U		55		56
BDir = D and SPDir = U	GDirTp1 = U		58		60
GDir = D and SPDir = U	GDirTp1 = U		62		60
SPDir = U	GDirTp1 = U		62	56	56
BDir = D	SPDirTp1 = U		57		55
GDir = D	SPDirTp1 = U		58		57
SPDir = D	SPDirTp1 = U		61	55	56
SPDir = D and BDir = D	SPDirTp1 = U		62		55
SPDir = D and BDir = U	SPDirTp1 = U		60		57
SPDir = D and GDir = D	SPDirTp1 = U		64		57
SPDir = D and GDir = U	SPDirTp1 = U		59	58	

We will next build a C5.0 decision tree model with tomorrow's direction as target. A C5.0 decision tree uses a non-numeric target, in this case, tomorrow's direction, and inputs can be either numeric or non-numeric. At this point, we can use many more variables as inputs because we are not restricted to only nominal values as we are with Apriori. The number of inputs for each model was 27, each of the variables listed in Table 1 for each of the series, minus the variables reflecting tomorrow's values. We built 12 of these trees, one for each of the four time periods and for each of the three data series. We used IBM's SPSS Modeler program to run each of them. The C5.0 decision tree identifies for us the variables that are most important in the decision by generating a relative importance score for each variable used in the model.

Relative importance scores sum to 1 for each model and indicate the relative impact each variable has on determining the final model output value. Tables 5a, 5b, and 5c show these scores for each period and each Target.

Table 5a. Brent Oil Tp1 target

Input	Calm	Bubble	Crisis	After
B2Day		0.05		
B3Day				
B4Day				
B5Day			0.24	
B5DayMA				
Bdir	0.06			
BNumDaysUp				
BPerChg				
BStd	0.03			
G2Day				
G3Day				
G4Day			0.04	
G5Day	0.61		0.02	
G5DayMA	0.09			
Gdir			0.07	
GNumDaysUp			0.06	
GPerChg	0.01			
GStd			0.17	
SP2Day				
SP3Day				
SP4Day	0.06			
SP5Day		0.95	0.33	
SP5DayMA	0.05			
SPDir	0.02			
SPNumDaysUp				
SPPerChg	0.07		0.06	
SPStd				

Columns with no values indicate that the model was not able to find any useful variables and always predicted a single direction for every day. Of the 12 models, three failed to generate discriminating models. These were Gold in the Calm period, and the S&P 500 in both the Bubble and After periods. This leaves 9 models with information about variable importance. In these models, we see that eight of the variables are used in at least three of the models. Of this group, three variables are based on Brent Oil (the Brent standardized value, the Brent percent change and the Brent 4-day directional pattern), one is derived from Gold (the 5-day directional pattern of Gold), and the remaining four are derived from the S&P closing value (the S&P percent change, the S&P standardized value, the S&P 5-day moving average, and the S&P 5-day directional pattern).

The variable most used, that was identified as important in 6 of the models, was the 5-day pattern of directional movement in the S&P, with a relative impact ranging from .33 to .95. That is, the model looks at a week's pattern of S&P movement and this accounts for at least one-third of the variable importance in two-thirds of the models, two from the Oil group, three in the Gold group, and one in the S&P group. This is an interesting variable because it is longer term (or, older) information and its influence occurs more often in oil and gold than in the S&P.

In addition to the impact of each of the variables, we are interested in the accuracy of forecast for each of the three models. Table 6 contains the counts for each Target variable during each of the periods. Here we have the predicted Tp1 (tomorrow) variable direction in the columns and the actual variable direction in the rows. We ask whether the correct predicted directions outnumber the incorrect ones for each target in each period. Looking, for example, at tomorrow's Brent Oil direction as the target, BDirTp1, we see that, in the Calm period the model predicted Down correctly 285 times and incorrectly 169 times. When predicting up, there were 334 incorrect predictions, and 503 correct ones.

Table 5b. Gold Tp1 target.

Input	Calm	Bubble	Crisis	After
B2Day			0.06	
B3Day			0.07	
B4Day			0.03	0.1
B5Day				
B5DayMA			0.11	
Bdir		0.21		
BNumDaysUp				
BPerChg			0.02	
BStd			0.14	0.05
G2Day			0.04	
G3Day				
G4Day				
G5Day		0.3		
G5DayMA			0.07	
Gdir				
GNumDaysUp				
GPerChg			0.09	
GStd				0.16
SP2Day		0.01		
SP3Day		0.19		
SP4Day				
SP5Day	0.95		0.36	0.42
SP5DayMA		0.21		0.06
SPDir				
SPNumDaysUp				
SPPerChg	0.05			0.06
SPStd		0.08		0.15

Table 5c. S&P 500 TP1 target.

Input	Calm	Bubble	Crisis	After
B2Day				
B3Day	0.01			
B4Day	0.12			
B5Day			0.44	
B5DayMA			0.08	
Bdir				
BNumDaysUp	0.08			
BPerChg	0.02		0.02	
BStd	0.1		0.02	
G2Day				
G3Day				
G4Day			0.14	
G5Day				
G5DayMA				
Gdir				
GNumDaysUp			0.11	
GPerChg				
GStd				
SP2Day			0.02	
SP3Day				
SP4Day				
SP5Day	0.57			
SP5DayMA				
SPDir			0.06	
SPNumDaysUp	0.09		0.05	
SPPerChg	0.01			
SPStd			0.02	

Table 6. Matrix showing Actual vs Predicted counts of series directional movement per period.

Act	Predicted							
	Calm		Bubble		Crisis		After	
BDir Tp1	Dn	Up	Dn	Up	Dn	Up	Dn	Up
Dn	285	334	190	133	145	13	-	562
Up	169	503	98	261	24	115	-	635
GDir Tp1	Dn	Up	Dn	Up	Dn	Up	Dn	Up
Dn	401	232	125	178	116	30	299	268
Up	325	333	28	351	15	136	173	457
SPDir Tp1	Dn	Up	Dn	Up	Dn	Up	Dn	Up
Dn	437	206	-	302	129	21	-	530
Up	284	364	-	380	35	112	-	667

Inspecting each prediction pair across the two rows for Brent oil, we see that during the calm, bubble, and crisis periods, the number of correct predictions outnumbered incorrect predictions in each of these. In the After period, where the C5.0 methodology failed to generate a model, we see that the number of incorrect predictions dominated. For Gold tomorrow, GDirTp1, we have in every period, model predictions where the number of correct forecasts are greater than the incorrect ones. For the S&P target, SPDirTp1, models failed to be generated in two of the periods, bubble and after. For the periods where C5.0 was able to generate a model, calm and crisis, both models were correct more than incorrect in each direction. In the periods for bubble and after, the models predicted only one direction for every day (up). It happened that a majority of the days were up, so these also show more correct than incorrect predictions.

Another way to look at the forecast numbers is by the percent of times each model was correct, whether in predicting and up or a down direction for tomorrow. Table 7 displays these percentages.

Table 7. Percent of correct forecasts in each model.

	Calm	Bubble	Crisis	After
BDirTp1	0.61	0.66	0.88	0.53
GDirTp1	0.57	0.70	0.85	0.63
SPDirTp1	0.62	0.56	0.81	0.56

During the crisis period, every target was predicted correctly more than 80% of the time. In the other periods, correctness ranged from a low of 53% to a high of 70%. During the calm period, predictive correctness was close to 60% for each target. In the bubble period, the predictive value dropped for the S&P, but rose for both oil and gold. In the crisis period, it was easier to predict all series direction tomorrow. Then after the crisis, all numbers dropped significantly.

We used two models, Association Analysis and C5.0 Decision Tree, to construct predictions for tomorrow's direction in Brent oil, Gold, and the S&P 500. Association analysis can only be used when there are rules that get triggered, while the decision tree model can be applied on any day. Comparing Table 4 confidence numbers to Table 7 percent of correct forecast numbers, we see that the C5.0 model was a clear winner in the crisis period. Using association analysis only on days when rules were triggered would have significantly limited your trading opportunities, and on these days, the confidences were much lower than the correctness of the decision tree models. In the After period for the SPTp1 target the association analysis rules only referred to the Up direction, and the decision tree model only predicted Up. Confidence and correctness percents were very similar. The Calm period decision tree model was correct about 60% of the time.

5 Conclusions

When searching for inter-relationships between two or among several assets or economic variables, regression and time series techniques are often employed. Inter-relationships among gold, oil and equities have attracted numerous studies and results are

often sample dependent. The literature in this area proposes the hypothesis that oil prices contribute to inflation and such inflation induces further increases in the price of gold. When inflation is mild, the valuation of equities is not affected; however, increased inflation affects equity returns.

In this paper we follow association analysis to investigate changing inter-relationships among gold, oil and equities during evolving regimes. In particular, looking backwards since the beginning of 2000, we identify four regimes called the calm period from January 4, 2000 to March 31, 2005; the bubble period of April 1, 2005 to December 31, 2007; the crisis period of January 2, 2008 to March 13, 2009 and finally the after the crisis period of March 16, 2009 through January 24, 2014. For each of these periods we perform three searches. First, we investigate the series of gold, oil and equity daily price direction, up or down, using Association Analysis to investigate which, if any, series typically moved in the same broad patterns within a day. Next, we look at today's movements to see if rules appear predicting tomorrow's movement. Finally, we will use both movement and numeric data based on the three series in a decision tree to forecast tomorrow's direction for each series.

There are numerous finding since there are four periods with three searches in each but the key finding of this work is that during the crisis period the inter-relationships among gold, oil and equities were the strongest as judged by the effectiveness of prediction, followed by the bubble period. In contrast both during the calm or after the crisis period, the inter-relationships weakened. Our results are informative because they highlight that within a very long time series sample uniform inter-relationships hardly persists; instead these inter-relationships evolve depending on the specific characteristics of the sequence of regimes. If regimes are characterized by low volatility and thus financial calmness the idiosyncratic markets characteristics of each asset prevail and thus reduce inter-dependencies; however, when financial turmoil emerges, it dominates idiosyncratic characteristics and inter-relationships strengthen because asset prices become highly correlated.

6 References

[1] Barro, Robert and Sanjay Misra (2013), Gold Returns, NBER Paper No. 18759

[2] Baur, Dirk G. and Brian M. Lucey (2010), Is Gold a Hedge or a Safe Haven? An Analysis of Stocks, Bonds and Gold, The Financial Review, 45, pp. 217-229.

[3] Reboredo, Juan (2013), Is Gold a Hedge or Safe Haven Against Oil Price Movements**?,** Resources Policy, 38, pp. 130-137.

[4] Cinera, Cetin, Constantin Gurdgievb, Brian M. Luceyb (2013), Hedges and safe havens: An examination of stocks, bonds, gold, oil and exchange rates, International Review of Financial Analysis, 29, pp. 202–211.

[5] Sari, Ramazan, Shawkat Hammoudeh and Ugur Soytas (2010) Dynamics of Oil Price, Precious Metal Prices, and Exchange Rate, Energy Economics, 32**,** pp 351-362.

[6] Malliaris, A.G. and Mary Malliaris (2013), Are Oil, Gold and the Euro Inter-related? Time Series and Neural Network Analysis, Review of Quantitative Finance and Accounting, 40, pp.1-14.

[7] Narayan, Paresh Kumar (2010), Gold and Oil Futures Markets: Are Markets Efficient, Applied Energy, 10 pp. 3299 – 3303.

[8] Fan, Ying and Jin-Hua Xu, (2011), What Has Driven Oil Prices Since 2000? A Structural Change Perspective, Energy Economics, 33**,** pp 1082-1094.

[9] Evanoff, D., George Kaufman and A.G. Malliaris (2012), Editors, "New Perspectives on Asset Bubbles"**,** Oxford Univerity Press, New York.

Using Bootstrap Aggregated Neural Networks for Peripheral Nerve Injury Treatment

Munish B. Shah[1], **Wei Chang**[1], **Kathleen McGuire**[1], **William Koch**[2], **Yan Meng**[2], **and Xiaojun Yu**[1]
[1]Chemistry, Chemical Biology, and Biomedical Engineering, Stevens Institute of Technology, Hoboken, NJ 07010, USA
[2]Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07010, USA

Abstract - *Accidents and trauma can cause severe peripheral nerve injuries and may require surgical intervention. While autografts are considered the current gold standard for complete regeneration of damaged nerves: their scarcity, potential loss of function at the donor site, and potential mismatch in axon diameter limits their use in practice and begs the need for optimal nerve guidance conduits (NGCs), which are the current viable alternative. The major challenges in current NGC research is the inability to account for variations in gap lengths, materials, and enhancement factors. Also, there is an inability to estimate the performance of NGCs, without in vitro and in vivo studies, so that it may be optimized to achieve maximum recovery for an injury. We propose a prediction model based on bootstrap aggregated neural networks in this paper that addresses these challenges and can alleviate the conventional burdens involved in the development of an NGC.*

Keywords: peripheral nerve injury; neural networks; nerve guidance conduit; data collection, bootstrapping

1 Introduction

The peripheral nervous system is made up of the nerves and ganglia outside of the central nervous system, which is the brain and spinal cord. The function of the peripheral nervous system is to bridge the central nervous system with the rest of body. Peripheral nerve injury (PNI) affects approximately 200,000 patients in the United States, with greater numbers reported globally [1]. PNI can be severe enough to result in a loss of function to certain parts of the body especially if there is a gap in the nerve. The gold standard to repair nerve injuries is through two methods: direct coaptation of the proximal and distal stump when the nerve gap is $\leq$4mm and when the nerve gap is >4mm, application of an autograft, in which a patient's own nerve from a secondary location on the body is applied to the primary site of injury. Autografts are scarce, can lead to a loss of function at the secondary site, and can have a mismatch in axon diameters; all of which encompass the drawbacks associated with this approach [2]. While allografts can be applied, they too have the same limitations as autografts. A viable alternative is nerve guidance conduits (NGCs), which are tubular guidance channels that are coapted to the proximal and distal ends of an injured nerve to guide the regenerating axon from the proximal stump to the distal stump. NGCs can enhance nerve regeneration if coupled with growth factors, cells, and proteins as well as aligned fibers [3]. Conventional methods of assessing the best method to optimizing NGCs is through in vitro and in vivo studies which expend time, capital, and resources. This is a continuous process that is repeated after the critical parameter(s) is altered until the expected results are achieved. Having an ability to bypass this process or at least alleviate the burdens faced by researchers so as to progress in optimizing NGCs and more effectively treat PNI.

A second challenge within the research and development of NGCs is the inability to account for the variety of materials used for fabrication, variety of factors that can be coupled with an NGC to promote nerve regeneration, variations in gap length, and the variety of experimental practices used by researchers to conduct in vivo studies to name a few which limit the comparison of NGCs. While these two challenges are currently hindering the pace of NGC optimization and its potential to match the ability of autografts, we address these challenges in this paper through the presentation of a prediction model based on a normalization standard noted as L/Lc.

2 Data Preparation

This section is focused on the normalization standard, the criteria of which we used to build our dataset. Additionally we detail our process of collecting our data and the format of our dataset.

2.1 Normalization Standard

A normalization standard allows one to compare products or processes that may vary in at least one aspect by adjusting a critical common parameter(s) associated with them. A normalization standard for NGCs related to PNI has been proposed by Dr. Ioannis Yannas's group of the Massachusetts Institute of Technology. This standard is denoted as L/Lc and is a calculated ratio of the gap length divided by the grafts critical axon elongation. It was developed to compare the regenerative activity of NGCs while accounting for aforementioned factors. L/Lc is based on the value of %N which reflects the successful rate of regeneration within an experimental group. For example if within a sample size of 5, 4 samples showed successful regeneration then the %N value is 80%. Having a parameter such as %N is critical in assessing how effective an NGC is and exactly how

repeatable and reliable successful regeneration is. While researchers generally report results and compare the performance of an NGC to the results of a tubular conduit and an autograft, it is also vital to report the %N to give an insight of how repeatable results are.

A plot of %N with respect to gap length was constructed using data collected from studies performed on rat sciatic nerves using silicone based NGCs. This yielded an S shaped curve which is used to approximate the value of Lc, the critical axon elongation, which is the number of samples within a group for which an axon reached at least 50% of the distance from the proximal stump to the distal stump. The Lc is simple a linear shift on the x-axis with respect to the value of %N and the gap length. Once the Lc is approximated for an NGC, it is compared against a standard NGC, defined as a tubular NGC, which is known to yield the poorest performance in nerve regeneration, using equation shown below [4,5].

$$\Delta L = \frac{L}{Lc}^{exp} - \frac{L}{Lc}^{exp} \quad (1)$$

The equation (1) was used to calculate the ΔL value, to provide a quantified measure of regenerative activity of an NGC relative to a standard NGC. This standard is also meant to be used to compare NGC effectiveness in nerve regeneration across different species, but for the purposes of the proof of concept it was validated on only mice and rat data. Since this concept has not yet been extended beyond applicability to rats and mice we focused on using it strictly for dealing with parameters and data from and for rats. Having a normalization standard is critical in the development of NGCs as it can allow researchers to gauge whether their NGC(s) is better than the standard NGCs which in turn will help them assess whether it is a move forward in the field of PNI. Additionally by being able to compare across gap lengths it can allow researchers to assess the capability of their NGC or NGCs developed by others to characterize its limitations as well as understand its advantages for nerve regeneration.

2.2 Data Collection

In the development of any prediction model, a vast and diverse a dataset is key in developing an effective prediction model. A large number of input parameters requires a more diverse and larger dataset so its accuracy is high and can be applied for its intended purpose. In our particular case we had a large number of input parameters and a relatively small dataset.

We focused on collecting data related to NGCs studied on rat sciatic nerves since that is the preferred anatomical site of choice for PNI as well as the fact we used the L/Lc normalization standard to build our dataset. Due to the absence of a central forum dedicated to the exchanging of ideas and research for NGCs related to PNI, we relied on collecting data from scientific publications through an array of databases and journals. We searched for publications on Google Scholar, EBSCOHOST, Science Direct, and the Wiley Online Library. The criteria for including a publication in our dataset was that in it: researchers detailed their protocol for animal studies, use the rat sciatic nerve as their anatomical site, provided values of %N, reported their recovery time(s), detailed their NGC designs, and used a standard NGC (tubular NGCs) to compare their experimental NGC performance. This criteria was designed to conform to the L/Lc normalization standard and to ensure the procedures used and the form of data reporting within the papers was uniform.

Using this criteria we were able to isolate a total of 28 scientific publications and within these publications each type of NGC that served as an experimental group was designated as a 'case' within our dataset. A total of 138 cases were recorded in Microsoft Excel and each parameter involved in the development of the NGC was noted. The output parameters were the value of Lc, ΔL, %N, and L/Lc. The Lc value for each of the negative controls in from each publication were designated as 0, since the goal was to compare the performance of each experimental NGC relative to the negative control. Lc was calculated for each experimental NGC using the S-shaped curve based on the gap length and %N, which were provided in each of the 28 publications. The ΔL for each experimental case was calculated using Eq 1. In our analysis we observed a few cases in which the performance of the experimental NGCs was equal to the negative control, however there were several cases in which the ΔL was as high as 6.8. This provides an assessment that progress is being made within the field to improve NGC design, and quantified measures such as ΔL can allow a better understanding of exactly how well the progress are being made and whether design can be improved or not. Each aspect related to the design of an NGC served as an input to the prediction model and was categorized as shown in Table I. A total of 46 parameters served as inputs. The parameters we noted are not the complete set of parameters used by researchers currently and the list can certainly be expanded in the future. These parameters were selected based on the fact that the publications that met our criteria for inclusion in the building of our dataset used these parameters to achieve successful nerve regeneration with their NGCs. We categorized the parameters as shown in Table I based on what their purpose was in NGC development and how they were applied. Categorizing the parameters clarifies the aspects associated with NGC development and allows a better understanding of which are essential to keep and which are optional. Such categorization can alleviate the challenge with optimization by allowing researchers to focus on a single category.

The output of the prediction model was selected as ΔL, since it reflects the effectiveness of a conduit relative to a standard conduit. The justification for using ΔL as the output for the prediction model stems from the fact it is the only output of the ones noted in the dataset that provides a quantified measure of the effectiveness of an NGC. While Lc, L/Lc, and %N provide a measure of NGC performance, these

parameters are specific to a single NGC case and cannot be used to compare the performance of different NGCs with each other. These parameters would be applicable in the assessment of an individual NGC, however in our approach we aimed to provide researchers with feedback about the status of the design of their NGCs through the ΔL value. ΔL does not have a known maximum limit or minimum limit, but can provide a researcher the opportunity to improve their NGC design and add/remove parameters that may improve/hinder its performance without the need to undertake the burden of doing in vitro and in vivo studies to come to this conclusion.

Table I. Inputs categorized for Prediction Model

Category	Model Input Parameters
Materials Processing	Phase Separation, Hydrogels, Electrospinning
Structure	Fibers, Gel, Permeable, Impermeable, Microsphere, Porous, Lumen
Materials	Collagen, Ethyl Vinyl Acetate, Polycaprolactone, Poly Lactide, Poly Glycolic Acid, Poly Lactide Co Glycotide, Chitosan, Poly Phosphazene, Poly Pyrrole, Poly Sulfone, Silicone
Form	Hydrogel, Liquid, Gel, Matrix, Fiber-Aligned, Fiber-Random, Microsphere, Solid
Growth Factors	NGF, BDNF, CNTF, GDNF, FGF, Denatured FGF, IGF, Laminin, Fibronectin, Schwann Cells, Bone Marrow Stromal Cells, Neural Crest Stem Cells
Growth Factor Arrangements	Gradients or Anisotropic, Isotropic

3 Prediction Model

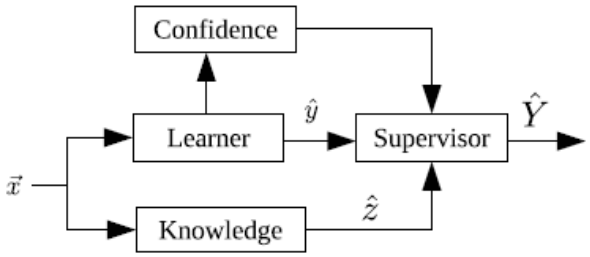
This section details the predictors we considered in the development of our prediction model as well as the design of our final product. We also note the challenges we faced in the application of our model and how we addressed these challenges.

3.1 Design of Prediction Model

In our particular dataset for NGCs we constructed based on the L/Lc normalization standard there was high correlation within the data and a very limited amount of data was available. However in compensation for the lack of data we had some domain knowledge about peripheral nerve injury with which the accuracy of the predictions could be improved if used with the appropriate predictor. The predictors we considered using to incorporate prior domain are virtual examples, hints, support vector machines, and artificial neural networks. Virtual examples are new training examples developed during a learning process using prior domain knowledge [6]. There are two methods to implement virtual examples: transforming the original data or synthesizing new examples [6-8]. The former has the challenge of assessing which method of transformation is appropriate and the potential that the virtual examples may correlate to the original data [9]. Also, it has been noted that virtual examples when tested with real data can result in poor performance [8]. Hints are simply additional knowledge about the target function which is useful during learning [10]. Hints can be applied as virtual examples. While hints have potential in making valid predictions we could not apply them for our purpose due to the drawbacks associated with virtual examples. Support vector machines is a machine learning algorithm that has shown promise in an array of biological areas, however it is computationally expensive and timely which is why it is inapplicable for our purpose [11]. While there is no shortage of predictors, our choice was the artificial neural network (ANN) as it commonly used and some 3rd party software even implement ANNs. While ANN is more complex than some predictors, it allows us the flexibility we require as well as that we have more experience using this predictor than others.

In our use of ANNs we used the technique of bootstrapping, which is a re-sampling technique used in machine learning beneficial when the amount of data available is limited as well as it can improve accuracy [12, 13]. Below is an illustration of the design of the prediction model.

Figure 1. Overview of our Prediction Model



The basic principle is that the PI indicates the level of confidence the learner has on the prediction. The domain knowledge is useful to compensate for the lack of knowledge of the learner. The learner output gets adjusted depending on how confident it is on estimating the output for the current input values. The learner component was defined as the bagged neural networks and was trained using the aforementioned dataset. In our case we used bootstrap aggregated neural networks which are developed when multiple neural networks are trained on bootstrap samples. Aggregated neural networks improve accuracy, improve generalization, and are more robust compared to just a single neural network as we assessed in [14]. Aggregated neural networks have a reduced chance of error compared to a single neural network. Based on the experiments we conducted in [14], we determined the optimal number of neural networks to be 100 when trained with particle swarm optimization, which provides greater accuracy than backpropagation for training neural networks. The confidence component computes the learner's PI for the current inputs. The knowledge component provides a separate estimation of the target function based on acquired domain knowledge. The supervisor component assesses the learner's confidence and determines if the learner's output should be influenced by domain knowledge. The supervisor component was responsible for fusing the prediction made by the knowledge base, the dataset, and the learner component, based on the confidence value, calculated through the prediction intervals (PI). PI determine how far an

estimated value is from the target. The PIs were calculated using the bootstrap aggregated neural networks, which can improve accuracy in predictions for even small datasets [15].

The idea was to keep the architecture simple and straightforward since this prediction model is intended to use as a guideline for future developments for models applied for NGC research. Using PIs allowed us to ensure we were able to provide as accurate of an output as possible given the size of the dataset and the number of inputs we had. The architecture of our prediction model is further detailed and described in [14].

3.2 Challenge of Applying the Prediction Model

A challenge we faced in development was to limit the number of combinations that could be created of the input parameters. This is based on the fact that if the prediction model devises potential combinations on its own then it can have lower accuracy, however if it is given a limited number of combinations of the inputs then the accuracy can be optimal especially within a large number of inputs and a small size dataset. Parameters within categories and parameters across categories can be coupled together to develop an NGC which increases the possible types of NGCs that can be developed. For example, researchers have coupled synthetic with natural materials to develop NGCs. Additionally such materials can be coupled with a growth factor, cells, proteins, or all three factors together. While our prediction model is capable of devising possible combinations that can be made of parameters for potential NGC designs, we limited this capability by providing it the potential combinations of the inputs. We devised combinations through a thorough literature review as well as our own knowledge and experience within the field. This approach allowed us to achieve optimal accuracy in our prediction model as well as account for all possible combinations of input parameters.

4 Experimental Results

As stated above the purpose of the prediction model is to estimate the performance of an NGC developed using a defined set of the inputs. We assessed the accuracy of the prediction model by assessing its training and prediction error, which were calculated using data for which the output was known. This type of data was designated as the control. The control in our case was the dataset we had used to train our prediction model. In order to effectively train the prediction model and assess its accuracy in making predictions, we applied a concept known as cross validation. In cross validation the dataset is split into several subsets of equal size which are used to train and test the accuracy in making predictions of a prediction model [16]. For this particular purpose we chose to apply the seven fold cross validation approach, and as such our dataset was split into seven parts of which 6 parts had 20 cases each and the 7th had 18 cases, since the number of cases could not be evenly divided by 7.

The first six parts were combined to yield a dataset of 120 cases, which were used to train the prediction model. The reason to have a bigger size dataset for training is due to the fact that it is more crucial to train a prediction model better than it is to test its prediction error. A prediction model will be able to make more accurate predictions only if it is trained well. The 120 cases were also used to assess training error of the model. In addition to training the prediction model we quantified the training error and noted it as shown in Table II. We used the remaining 18 cases to assess the prediction error, the results of which are shown in Table III. There were three parts for which the dataset was crucial: training, then assessing training error, and assessing the prediction error.

Table II. Training Accuracy of the Prediction Model

Material	Growth Factors	Permeable	Lumen	Gap Length	Actual ΔL	Predicted ΔL	Difference in ΔL
Collagen	1 M Schwann Cells	Yes	1	18 mm	6.8	5.4	1.4
Collagen	None	Yes	1	22 mm	9.4	7.56	1.84
Ethyl Vinyl Acetate	.004% Fibroblast Growth Factor/BSA	No	1	15 mm	6.8	5.19	1.61
Silicon	30X10^4/mm^2 Calcium Ions	No	1	15 mm	6.8	5.055	1.745
Polylactide	1670 Neural Crest Stem Cells	No	1	10 mm	4.533	3.196	1.337

Table III. Prediction Accuracy of the Prediction Model

Material	Growth Factors	Permeable	Lumen	Gap Length	Actual ΔL	Predicted ΔL	Difference in ΔL
Collagen	None	Yes	1	20 mm	10.8	7.343089	3.456911
Silicon	None	Yes	1	16 mm	5.67	3.5464	2.1236
Collagen	None	No	1	10 mm	5.4	4.910012	0.489988
Polysulfone	100 μg/ml NGF	No	1	10 mm	3.4	2.61726	0.78274
Polylactic-co-glycolactide	None	No	1	10 mm	3.020909	1.730524	1.290385

An aspect to note with Tables II and III is each data recorded is specific to a researcher's expertise and skillset as well as the protocol by which the studies were performed. As with any prediction model our work is also susceptible to error as shown by the results in Table II and Table III, however such an occurrence is expected especially in light of the fact we had a small volume of publications to work with and a high number of input parameters. While there is error present in training the error is precise and the precision in our error ensures that our model is able to make uniform predictions regardless of variation in protocols followed by researchers as well as the fact that basing our prediction model on the L/Lc normalization standard allowed us to normalize our data despite differences in the materials used,

growth factors, cells, and proteins that were coupled, and variations in gap length.

As for the errors in training and prediction, these errors can be reduced with the addition of more cases in our dataset. Our focus in developing the prediction model is emphasize the need for standardization in doing NGC research related to PNI so in future all researchers report their %N values. Having more papers published in this regard will allow us to expand our dataset and conduct future developments on our prediction model so when more cases can be made available through future publications similar to the ones used to develop the dataset for this model the error can ideally be reduced and reliability of our prediction model will increase. Additionally since we categorized the inputs involved in the design and development of an NGC we expect any alteration in any category to impact the value of ΔL to further the understand the relationship the mechanism of nerve regeneration related to an NGC and the parameters that are involved in development. When researchers are able to isolate which category or parameter impact the output the most it will allow them a better understanding of where to make modifications. By having a deeper understanding of the mechanism of nerve regeneration for an NGC researchers can be better prepared to optimize NGCs and have a better grasp on how to deal with critical PNIs.

5 Discussion

To our knowledge our work provides the only prediction model relevant to estimating NGC performance with bootstrap aggregated neural networks to calculate prediction intervals and uses particle swarm optimization for training to assess NGC performance. The work that has been done prior involved the application of a single neural network and ours provides the advantage of random sampling of data, which is a more practical approach to developing a prediction model since all types of data should be viewed in making a prediction rather than focus on a specific set. By focusing our model on being able to analyze all forms of NGC data it is better trained to understand and predict the performance of a novel NGC design. Our aim is to alleviate the conventional burdens associated with NGC development, particularly the necessity to perform in vitro and in vivo studies. By using such a process we ensured our model is able to sample all forms of data that is provided to it in regard to NGCs. This ideally mimics a researchers thought process of reviewing all literature and data available on NGCs to effectively critique their own design using critical thinking and make future developments. We designed our prediction model to mimic the thought process of our prediction model in terms of critical thinking. In our algorithm we design our model to process data similar to how. Since papers tend to report data for an NGC on a specific gap length of study and under conditions specific for experiments, researchers cannot assume the performance of that under different parameters which is why they must account for data provided and apply critical thinking before moving forward in their own designs and experiments.

A valid and applicable prediction model for NGCs has the potential to alleviate the conventional burdens associated with development and assessment of viability as well as the potential to allow researchers to develop the optimal NGCs for specific injuries, such as those specific for a certain gap length of injury. Despite the relatively small dataset size we had to work with we were able to successfully prove our concept of being able to develop a prediction model applicable to NGCs. This prediction model was developed to serve as a guideline for future work.

The L/Lc normalization standard allowed us to effectively develop a prediction model applicable for any gap length as well as standardize for materials and additional factors. We understand that the L/Lc normalization standard is not without error, the critical being the inability to account for recovery time which significantly impacts the outcome of PNI treatment by an NGC. We also realize that L/Lc was proposed solely as a normalization standard however by applying it in terms of building a dataset for use in developing a prediction model allows us the opportunity to contribute a novel approach in assessing the outcome of a potential NGC design rather than rely on the conventional methods.

The vast amount of parameters makes it difficult often for researchers and readers to gauge the complexity of designing an NGC and can further hinder its optimization, as such we aimed to clarify this aspect through the presentation of parameters we applied in our prediction model. Our aim is to allow an organized approach to understanding NGC development as evident by our categorization of the input parameters as well as the potential combinations of these parameters. Finally by being able to modify each parameter or parameters within the categories noted the ΔL can ideally be affected which will allow researchers to understand which category and which parameter(s) has the most impact on nerve regeneration.

By saving cost, time, resources and developing an optimal NGC researchers can expedite the steps from design to commercialization and bring more effective NGCs to market. We designed our model to use random sampling of the data so as to provide an unbiased analysis and output of a novel design. Most importantly researchers will be able to potentially devise effective NGCs that can match the performance of autografts if not outperform them to provide maximum nerve regeneration affected patients. While there are very few, if any, conduits that have come close to providing nerve regeneration comparable to autografts this potential is being attained more each day. Having a valid substitute to autografts will allow clinicians to rely less on autografts, thus eliminating the drawbacks that are associated with them and apply more NGCs to treat even critical PNIs thus effectively dealing with the growing number of PNI cases globally.

6 Acknowledements

This work was generously supported by NIH-R15 NS074404. The author, Munish Bhupendra Shah, was supported by NSF-0740462.

7 References

[1] Lehoe S, Zhang XF, Boyd D. “FDA approved guidance conduits and wraps for peripheral nerve injury: A review of materials and efficacy.” Injury, vol 43, no. 5, pp. 553-572, 2012.
[2] Yu X, Bellamkonda RV. “Tissue-Engineered Scaffolds Are Effective Alternatives to Autografts for Bridging Peripheral Nerve Gaps.” Tissue Engineering, vol 9, no. 3, pp. 421-430, 2003.
[3] Daly W, Yao L, Zeugolis D, Windebank A, Pandit A. “A biomaterials approach to peripheral nerve regeneration: bridging the peripheral nerve gap and enhancing functional recovery.” Journal of The Royal Society, vol 9, no. 67, pp. 202-221, 2012.
[4] Yannas I V, Hill B J. “Selection of Biomaterials for peripheral nerve regeneration using data from the nerve chamber model.” Biomaterials, vol 25, pp. 1593-1600, 2004.
[5] Zhang M, Yannas I, . “Peripheral Nerve Regeneration.” Adv Biochem Engin/Biotechnol, vol. 94, pp. 67-89, 2005.
[6] P. Niyogi, F. Girosi, and T. Poggio, “Incorporating prior information in machine learning by creating virtual examples,” Proceedings of the IEEE, vol. 86, no. 11, 1998.
[7] T. Yu, T. Jan, S. Simoff, and J. Debenham, “Incorporating prior domain knowledge into inductive machine learning,” pp. 1–42, 2007.
[8] D. Pomerleau, “Efficient training of artificial neural networks for autonomous navigation,” Neural Computation, pp. 1–10, 1991.
[9] T. Yu, T. Jan, S. Simoff, and J. Debenham, “Incorporating prior domain knowledge into inductive machine learning,” pp. 1–42, 2007.
[10] Y. S. Abu-Mostafa, “Hints,” Neural Computation, vol. 7, no. 4, pp. 639–671, 1995.
[11] Furey, Terrence S., et al. "Support vector machine classification and validation of cancer tissue samples using microarray expression data." Bioinformatics, vol 16, no. 10 PP. 906-914, 2000.
[12] Efron, Bradley, and Robert J. Tibshirani. An introduction to the bootstrap. CRC press, vol. 57, 1994.
[13] H. J. Ader, G. J. Mellenbergh, and D. J. Hand, Advising on Research Methods: a consultant companion. Johannes van Kessel Publ., 2008.
[14] Koch, William, Yan Meng, Munish Shah, Wei Chang, and Xiaojun Yu. "Predicting nerve guidance conduit performance for peripheral nerve regeneration using bootstrap aggregated neural networks." In Neural Networks (IJCNN), The 2013 IEEE International Joint Conference on, pp. 1-7, 2013.
[15] Breiman,Leo. “Bagging Predictors.” Machine Learning, vol 24, pp. 123-140, 1996.
[16] Kohavi, Ron. "A study of cross-validation and bootstrap for accuracy estimation and model selection." In IJCAI, vol. 14, no. 2, pp. 1137-1145, 1995.

The Significance of the Race Factor in Breast Cancer Prognosis

M. Mehdi Owrang O.
Department of Computer Science, American University, Washington, D.C, USA

Abstract - *In this work, we looked at the significance of the race factor in breast cancer prognosis, using Association rules data mining technique. We utilized XLMiner data mining tool for our experiments. The data used is the National Cancer Institute's SEER Public-Use Data. Several experiments were conducted based on the prognostic factors including those of Age, Behavior code, Stage of cancer, Grade, and Marital status with respect to Race. Our discovered association rules indicate that Japanese patients have better survival rate than White patients and White //patients have better survival rate than Black patients.*

Keywords: Data Mining, Association Rules, Breast Cancer Prognosis,

1 Introduction

Breast cancer is a malignant cancer causing tumor that begins when cells in the breast tissue grow abnormally, without managing cell division and cell death rates. Breast cancer is the most common female cancer in the US, the second most common cause of cancer death in women, and the main cause of death in women ages 40 to 59 [6]. Approximately 232,340 new cases of invasive breast cancer are expected to be diagnosed in the United States in 2013, and almost 40,000 will die from the disease [7, 9]. The lifetime probability of developing breast cancer is one in six overall (one in eight for invasive disease) [8, 9, 25].

Breast cancer treatments can be classified as local or systematic. Surgery and radiation fall under local while chemotherapy and hormone therapy are examples of systematic therapies. Usually for the best results, the two types of treatments are used collectively [9]. Although breast cancer is the second leading cause of cancer death in women, the survival rate is high. With early diagnosis, 97% of women survive for 5 years or more [6, 7, 8].

Although cancer research is generally clinical and/or biological in nature, data mining research is becoming a common match. In medical domains where data and statistics driven research is successfully applied, new and fresh research directions are recognized to promote clinical and biological research.

Forecasting the result of a disease or discovering information previously unknown is one of the most inspiring and challenging tasks in which to develop data mining applications. Survival analysis is a field in medical prognosis that deals with application of various techniques to historical data in order to predict the survival of a particular patient suffering from a disease over a time period [3, 10-15]. With the advancement of technology, automated tools, storage and retrieval of large volumes of medical data are being collected and are being made available to the medical research community who has been interested in developing prediction models for survivability [3, 10-15].

In our prior work [26], we have done experiments on breast cancer data set to discover association rules. In our analysis of the discovered rules, while most of the results were agreeable by domain experts, we did find some inconclusive patterns (i.e., in survivability by age>45, stage of cancer (localized)) that suggested that race factor may have some significance in the survivability prediction of the breast cancer patients. This has motivated us to further examine the significance of the race factor in the breast cancer prognosis.

In this study, we present an analysis of the prediction of survivability rate of breast cancer patients using association rules mining technique and data mining tool of XLMiner [4]. Experimental results are analyzed. The results supports the fact that race has a role in the prediction of the survivability rate of the breast cancer patients.

2 Related research

Several studies have been carried out on the survivability prediction of breast cancer using Naïve Bayes and Classification Trees, Artificial Neural Networks and statistical techniques of regression [3, 10-18]. In their study, Delen et al. [10] preprocessed the SEER data (1973-2000 with 433,272 records contained in a flat file breast.txt) for breast cancer. They removed redundancies and missing information resulting in a data set with 202,932 records, which then pre-classified into two groups of "survived" (93,273) and "not survived"

(109,659) depending on the Survival Time Recode (STR) field. The "survived" class is records that contain value greater than or equal 60 months in the STR field and the "not survived" class represents the remaining records. In this study, authors have used data mining algorithms Artificial Neural Networks, decision trees, and logistic regression to develop the breast cancer prediction models. The results indicated that the decision tree (C5) is the best predictor with 93.6% accuracy on the sample, artificial neural networks came second with 91,2% accuracy and the logistic regression models came to be the worst of the three with 89.2% accuracy.

In the study of Bellaachia et al. [11] (period of 1973-2002 with 482,052 records) the approach takes into consideration, besides the Survival Time Recode (STR), the Vital Status Recode (VSR) and Cause of Death (COD) fields as well. They achieved classification rate of about 87%. Authors have investigated three data mining techniques: the Naïve Bayes, the back-propagated neural network, and the C4.5 decision tree algorithms. In their experiment, the Neural Networks and Decision tree had comparable performance with 86.5% and 86.7% accuracy.

In [18], authors applied seven data mining algorithms, artificial neural networks, Naïve Bayes, Decision Trees (ID3), Decision Trees (J48), DT with Naïve Bayes, Bayes Net, and Logistic Regression on SEER breast cancer data set. In their experiments, logistic regression model had the highest accuracy (85.8%) and Decision Trees (ID3) had the worst accuracy (82.3%).

The use of Association rules for breast cancer prediction has been sparse. The main advantage of this method is that it generates clear and simple rules of the form "IF X THEN Y", which is very transparent and easy to understand by medical people. This motivated us to explore the use of Association Rules for breast cancer survivability prediction [16]. In this study, we used the commercial data mining tool XLMiner [4] for our experiments.

3 Data preparation

We have used the data contained in the Surveillance, Epidemiology, and End Results (SEER) Program Cancer Incidence Public-Use Database for the years 1973-2004 [5]. We queried the SEER database using the SEER Stat software which is a front end tool that connects to the SEER database. Our selection criteria include all records where site recode equals "Breast" the sex is "Female" for the above period. A total of 770,000 records were generated by the SEER database based on our selection criteria. The SEER Breast cancer data consisted of 115 variables [5]. These variables provide socio-demographic and cancer specific information regarding incidence of cancer. Based on SEER personnel advice, 16 variables/attributes have been selected as shown in Table 1. After eliminating redundancy and missing information, and selecting records for the year greater than 1988, we narrowed our selection to 71,077 records. Due to the limitation of XLMiner, the selection was further reduced to 60,000 records. The reader is referred to the SEER documents for detail descriptions and values of all the variables defined in Table 1 [19].

Table1. Variables used for Knowledge Discovery.

Categorical Variable Name	**Distinct Values**
Behavior Code	4
Race	19
Marital Status at Diagnosis	6
Extension of Tumor	23
Radiation	9
Lymph Node Involvement	10
Grade	5
Diagnostic Confirmation	8
Stage of Cancer	5
Cause of Death	2
Primary Site	10
Continuous Variables	**Range**
RX Summ -Surgery Primary Site	0 -99
Number of Primaries	1 -8
Number of Positive Nodes	0-50
Age at Diagnosis	17 – 102
Survival Time Recode Total Months	0 – 83

4 Association rules mining on breast cancer data set

Medical databases are often analyzed with classification trees, clustering, artificial neural networks, and regression techniques [3, 13-15, 17, 18]. Association rules mining technique is used here for predicting the survivability rate of breast cancer patients. In contrast to other data mining techniques, it is adequate to discover combinatorial patterns that exist in subsets of the data set attributes.

4.1 Association rules

Association rules show attributes value conditions that occur frequently together in a given dataset [1, 2, 4].

Association rule mining finds interesting associations and/or correlation relationships among large set of data items. That is, given a collection of items and a set of records, each of which contain some number of items from the given collection, an association function can find rules such as 70% of all the records that contain items A, B, and C also contain items D and E.

There are several numbers that are associated with an association rule. The first number (a) is called the support for the rule. The support is simply the number of transactions that include all items in the antecedent and consequent parts of the rule. The other number (b) is known as the confidence of the rule. Confidence is the ratio of the number of transactions that include all items in the consequent as well as the antecedent (namely, the support) to the number of transactions that include all items in the antecedent. Lift is one more parameter of interest in the association analysis. Lift is nothing but the ratio of Confidence to Expected Confidence [1, 2, 4]. It is defined as Lift=Confidence of the rule / (ratio of the records containing the consequent to the total number of the records in the data set).

Several iterations of model generation were done to achieve an optimized model that had good support for the rules and corresponding confidence of the rules. Unlike the if-then rules of logic, association rules are probabilistic in nature. Using the association rules mining with minimum support of 1000 and minimum confidence of 85%, 33,499 rules has been generated.

4.2 Prognostic factors in breast cancer

Medical prognosis is a field in medicine that encompasses the science of estimating the complication and recurrence of disease and to predict survival of patient [3, 10, 11, 14, 15, 18]. Survival analysis is a field in medical prognosis that deals with the application of various methods to estimate the survival of a particular patient suffering from a disease.

Although scientists do not know the exact cause of most breast cancer, they do know some of the risk factors that increase the likelihood of a woman developing breast cancer. These factors include such attributes as age, genetic risk, and family history among others.

A prognostic factor may be defined as a measurable variable that correlates with the natural history of the disease. The prognostic factors used in the prediction of survival of breast cancer can be separated into two categories: chronological (based on the amount of time present, i.e., Stage of Cancer), or biological (based on the potential behavior of the tumor, i.e., Histological Grade) [6, 20-22]

Lymph node status, tumor size, histological grade are among the prognostic factors in use today [6, 20-22]. Lymph node status is time-dependent factor and is directly related to survival. One of the most significant prognostic factors in breast cancer is the presence or absence of axillary lymph node involvement, which is usually assessed at the time of surgery using sentinel lymph node biopsy or axillary dissection [22]. Macrometastases (>0.2 cm in size) have clearly been shown to have prognostic significance.

Survival is inversely related to the size of the tumor. The probability of long term survival is better with smaller tumor than with larger tumor [6, 20, 21]. Tumor size has long been recognized as an independent prognostic factor and as a predictor of axillary node status, with larger tumors being associated with a worse prognosis and an increased likelihood of nodal metastasis. In [11], authors have used Weka mining tool and ranked the survivability attributes. The result indicates that Extension of Tumor has a higher rank than the tumor size.

Histological grade is being identified as being highly correlated with long term survival. Patients with a grade 1 tumor have a much better chance of surviving than patients with grade 3 tumor [9]. In [10], authors also conducted sensitivity analysis on artificial neural networks model in order to gain insight into the relative contribution of the independent variables to predict survivability. The sensitivity results indicated that the prognosis factor "Grade" is by far the most important predictor, which is consistent with the previous research, followed by "Stage of Cancer", "Radiation", and "Number of Primaries". Why these prognostic factors are more important predictors than the other is a question that can only be answered by medical professional and further clinical studies.

Other factors include the patient's age, general health, estrogen-receptor and progesterone-receptor levels in the tumor tissue.

4.3 Experimental results and analysis

As we noted, it was our intension, through experiments, to study the significance of the established prognostic factors and their combinations on the prediction of the survivability rates of the breast cancer patients. Most of the results from our experiments support the available clinically established prognostic factors.

We were intrigued with the result that survival was high when Race=Japanese. Table 2 shows the survivability distribution by Race, where a value "Alive" for the Cause of death means patient survived and a value "Breast" means patient died. We explored the data for Japanese only but could not determine from the available

fields/data any pattern that could explain it. After talking with domain experts, the result could possibly be highlighted to other factors including those of genetic, food habit, environment and Climate which are not part of the SEER data.

Table 2. Survivability by Race.

Race	Cause of death	Total	%
Japanese	Alive	1085	89%
Japanese	Breast	134	11%
White	Alive	40296	75.1%
White	Breast	13385	24.9%
Black	Alive	3671	58.1%
Black	Breast	2652	41.9%

As noted by Newman [23], breast cancer pathogenesis and epidemiology is complex and influenced by a great number of environmental and lifestyle factors that impact on lifetime hormone exposures. Also, the genetic, socioeconomic, and cultural features associated with ethnic background could further complicate the picture. For African-American women, these different elements tend to generate the unusual kind of patterns of a relatively lower cancer incidence, higher mortality rate, and younger age distribution (age of 45 years have a greater incidence of breast cancer than Caucasian-American women in this young age range) [23, 24]. Our association rule discovery based on race and age indicates that Japanese women, at age of 45 or older, have the highest and black women have the lowest survivability rate. The survivability rate of black women at age of 45 or Less is much lower than the white women, as shown in Table 3. For the age group of "45 or Less", the Asian women (Japanese, Chinese, etc.) did not have enough records to satisfy the threshold set for the support/confidence for the rule.

Table 3. Survivability by Race and Age.

Race	Age	Cause of Death	Confidence %
Japanese	45 Plus	Alive	89.27
Chinese	45 Plus	Alive	85.63
Pilipino	45 Plus	Alive	81.23
White	45 Plus	Alive	74.82
Black	45 Plus	Alive	59.23
White	45 or Less	Alive	76.48
Black	45 or Less	Alive	54.01

Considering the survivability by Behavior Code and Race, we'll see that white women have slightly better chance of survival than the black women in In Situ cancer (97.97%-95.41%), but a much higher chance of survival in Malignant cancer (70.70%-51.92%). We should point out that Japanese patients, surviving with a confidence of 86.29% are doing better than white and black patients in Malignant cancer. Table 4 shows the survivability by Race and Behavior Code.

Table 4. Survivability by Race and Behavior Code.

Race	Behavior Code	Cause of Death	Conf. %	Lift Rate
White	In Situ	Alive	97.97	10.09
Black	In Situ	Alive	95.41	9.83
White	Malignant	Alive	70.70	7.28
Black	Malignant	Alive	51.92	5.34
Japanese	Malignant	Alive	86.29	8.89

Considering the survivability by Stage of Cancer and Race, we'll see that white women are doing better than the black women in In Situ cancer (97.97%-95.41%), Localized cancer (88.86%-79.95%), Distant cancer (90.36%-95.41%. We should point out that again, Japanese patients, surviving with a confidence of 95.17%, are doing better than white and black patients in localized cancer. Table 5 shows the survivability by Race and Stage of Cancer.

Table 5. Survivability by Race and Stage of Cancer.

Race	Stage of Cancer	Cause of Death	Conf. %	Lift Ratio
White	In Situ	Alive	97.97	26.55
Black	In Situ	Alive	95.41	28.03
Japanese	Localized	Alive	95.17	9.80
White	Localized	Alive	88.86	9.15
Black	Localized	Alive	79.95	8.23
White	Regional	Alive	63.58	6.55
Black	Regional	Breast	54.19	15.92
White	Distant	Breast	90.36	26.55
Black	Distant	Breast	95.41	28.03
White	Unstaged	Breast	81.40	23.91

Histological grade is another important factor for the breast cancer prognosis. Based on a biopsy, the grade could take a value of Poorly Differentiated, Grade III which gives a poor prognosis. Patients with a Well differentiated Grade I have a much better chance of survival. Table 6 shows that White women are doing better that the Black women with regard to histological prognostic factor.

Table 6. Survivability by Race and Grade.

Race	Grade	Cause of Death	Conf. %	Lift Ratio
White	Poorly diff., GradeIII	Alive	62.15	6.40
Black	Poorly diff., GradeIII	Breast	53.31	15.66
White	Moderately Diff., Grade II	Alive	83.65	8.62
Black	Moderately Diff., Grade II	Alive	72.27	7.44
White	Unknown	Alive	65.70	6.77
Black	Unknown	Alive	50.56	5.21
White	Well Diff., Grade I	Alive	93.07	9.58
White	Undiff., Grade IV	Alive	78.25	8.06

In our experiments (Table 7), we have found out that Marital Status has some significance (although not conclusive) in the patients breast cancer prognosis. Our analysis of the data showed that survival rates among breast cancer were generally higher among married women. This value varied between racial and age groups, but was on average 10.03 percent among racial groups and 3.62 percent average across age groups.

Table 7. Survivability by Race and Marital Status.

Race	Marital Status	Cause Of Death	Conf. %	Lift Ratio
Japanese	Married	Alive	91.69	9.44
White	Married	Alive	80.47	8.29
Black	Married	Alive	64.78	6.67
White	Divorced	Alive	74.71	7.69
Black	Divorced	Alive	60.30	6.21
White	Single	Alive	72.65	7.48
Black	Single	Alive	54.20	5.58
White	Widowed	Alive	63.31	6.52
Black	Widowed	Alive	50.91	5.24

4.4 Statistical analysis

Our data mining experiments suggest that the race of the patient has some significance in the survivability rate of the patient. The "confidence" levels of the association rules can be verified by statistical assessment of the hypotheses (or rules). The more widely accepted statistical methods are Odds Ratio (OR) [28] and P-value from hypothesis testing [29].

The OR evaluates whether the odds of a certain event or outcome is the same for two groups. Specifically, the OR measures the ratio of the odds that an event or result will occur (i.e., cause of death=Alive) to the odds of the event not happening (cause of death=Breast). Clinically, that often means that the researcher measures the ratio of the odds of a disease occurring or a death from a specific injury or illness happening to the odds of the disease or death not occurring.

Considering the Survivability by race data in Table 2, calculation of the Odds Ratio is shown in Equation 1, where "PG_1" represents the odds of the event of interest for Group 1 (Japanese patients), and "PG_2" represents the odds of the event of interest for Group 2 (White patients).

$$\text{Odds ratio} = \frac{PG_1 / (1 - PG_1)}{PG_2 / (1 - PG_2)} \quad (1)$$

Odds ratio= (odds for Japanese survival) / (odds for White survival)

Odds ratio= ((1085/1219) / (1-1085/1219)) / ((40296/53681) / (1-40296/53681))

Odds ratio= ((.89/.11)) / ((.75/.25))

Odds ratio= 8.10/3.0=2.697

Thus, for a Japanese breast cancer patient, the odds of survival is 2.697 larger than the odds for a White patient. Likewise, for a White patient, the odds of survival is 2.172 larger than the odds for a Black patient. Based on Table 2 (the contingency table), we have done Chi-square tests (using StatCrunch software) to determine whether the results for the ORs are statistically significant. The significance tests resulted in a Chi-square=124.81985 and P-value <0.0001 for the Japanese/White groups and a Chi-square=835.54407 and P-value <0.0001 for the White/Black groups. The high values of the Chi-squares and very low values for P-values indicate that the differences are statistically significant.

Hypothesis testing is a well-established tool for scientific discovery. It enables us to distinguish results that represent systematic effects in the data from those that are due to a random chance. Hypothesis testing involves a comparison of two or more sub-populations.

Using StatCrunch software [27], we have done proportions hypothesis testing with two samples (Japanese and White) summary data of Table 2. The hypothesis was to see if the disparity in the survival rate of the Japanese and White breast cancer patients was statistically significant. The calculated P-value was <o.ooo1, which resulted in the rejection of the null hypothesis that there is no difference in the survival rate between the two populations. Similarly, for the proportions hypothesis test for the White and Black patients survival rates difference, the P-value was again <0.0001, causing the null hypothesis (no differences in the survival rate between the two groups) to be rejected.

The statistical methods validate the data mining result that the race factor is in fact statistically significant in the survivability rate of the breast cancer patients.

4.5 Problems with association rule mining

The problem of mining association rules in a high dimensional data set with numeric and categorical attributes is challenging, due to the large number of patterns rather than to dataset size. Most approaches are exhaustive in the sense that they find all rules above the user-specified thresholds. Such an approach produces a huge number of rules which may contain redundant or irrelevant information, or describes trivial knowledge. In addition, Association rules that involve many items are hard to interpret.

Association rule can be constrained to have a maximum number of attributes in the IF part, if the data mining tool allows you [1, 2, 4]. We can specify minimum % confidence value to reduce the number of rules produced. Likewise, we can increase the value for the support to reduce the number of rules. We should be careful, however, not to set too low values for the confidence and too high values for support as we may inadvertently eliminate the chance of discovering unusual patterns. For example, in our survivability by race experiment, if we set the minimum support value to 2000, we would not find that Japanese patients have higher survivability rate than other races. Finally, use the rules with higher lift value, to reduce the number of rules.

If the number of discovered rules is large, then it is desirable to obtain a few representative rules so that many rules can be derived from them. It is recommended then that a cover of association rules with the same consequent be generated. Given two rules with the same consequent: R_1: X_1=>Y and R_2: X_2=>Y, such that $X_1 \subseteq X_2$, it is said that R_1 covers R_2. Then, R_2 can be included and R_1 omitted. When there are several rules covered by R_2, this will produce a concise summary. This approach is applied to each set of rules with the same consequent but different antecedent item sets. Covers included rules whose antecedent was a superset of the antecedent of simple rules.

5 Conclusion

In this paper, we have done experiments on breast cancer data in order to study the significance of the prognostic factor of race in the overall survivability of the breast cancer patients. Data mining tool of XLMiner and the Association rule mining technique have been utilized in this effort.

The observations can be summarized as characteristics of survived vs. not survived with respect to prognostic factors of Race, Age, Stage of cancer, Grade, and Marital Status. In general, several factors such as Race, Age at Diagnosis, Survival Time Recode can have influences for breast cancer survivability. In terms of survivability rate amongst different races, Asian, especially Japanese have a better rate of survivability. Factors such as food habit, work/occupational environment, and genetics could influence the survivability rate. However, such information was not available on SEER breast cancer data that we used for our experiments.

Currently, we are evaluating additional prognostic factors and/or combination of factors that might be used to predict patient survivability rate. We also need to find breast cancer data sets which include other factors including those of genetic and family history, hormone therapy, information about late or no pregnancy, eating habits in order to better predict the survivability rate of breast cancer patient

6 References

[1] J. Han, M. Kamber, J. Pei. "Data Mining: Concepts and Techniques, Third Edition", *The Morgan Kaufmann Series in Data Management Systems*, 2011.

[2] H. W. Ian, E. Frank. "Data Mining: Practical machine learning tools and techniques, Third Edition". *San Francisco: The Morgan Kaufmann Series in Data Management Systems*, 2011.

[3] T. Jonsdottir., E.T. Hvannberg, H. Sigurdsson, S. Sigurdsson. "The feasibility of constructing a Predictive Outcome Model for breast cancer using the tools of data mining", *Expert Systems with Applications* , Vol. 34, No. 1, PP. 108–118, 2008.

[4] "XLMiner On Line, User Manual" [Online], Available: http://www.solver.com/xlminer-data-mining, [Accessed: 01-September-2013].

[5] "Surveillance, Epidemiology, and End Results (SEER) Program Public-Use Data (1973-2004)", *National Cancer*

Institute, DCCPS, Surveillance Research Program, Cancer Statistics Branch, [Online], Available: www.seer.cancer.gov, [Accessed: 15-January-2013].
[6] M.E. Costanza,, W.Y. Chen. "Epidemiology and risk factors for breast cancer", [Online], Available: ,http://www.uptodate.com/contents/epidemiology-and-risk-factors-for-breast-cancer, [Accessed: 10-Feb.-2013].
[7] R. Siegel. D. Naishadham, A. Jamal., 2012. , *Cancer Statistics, CA Cancer J Clin*, Vol. 62, No. 10., 2012.
[8] "Seer Cancer Statistics Review, 1975-2010", [Online], Available: http://seer.cancer.gov/csr/1975_2010/, [Accessed: 01-September-2013].
[9] "Breast Cancer Q & A/Facts and Statistics, ", [Online] Available: http://www.komen.org/bei/bhealth/QA/q-and-a.asp, [Accessed: 01-March-2013]
[10] D. Delen, G. Walker, A. Kadam. "Predicting breast cancer survivability: a comparison of three data mining methods", *Artificial Intelligence in Medicine*, Vol. 34, No. 2, PP. 113-127, June 2005,
[11] A. Bellaachia, E. Guven. "Predicting Breast Cancer Survivability Using Data Mining Techniques", *Ninth Workshop on Mining Scientific and Engineering DataSets in Conjunction with the Sixth SIAM International Conference on Data Mining (SDM 2006).*
[12] S. Kharya. "Using Data Mining Techniques for Diagnosis and Prognosis of Cancer Disease", International *Journal of Computer Science, Engineering and Information Technology (IJCSEIT)*, Vol. 2, No. 2, 55-56, April 2012.
[13] D. Delen. "Analysis of cancer data: a data mining approach", *The Journal of Knowledge Engineering, Expert Systems*, Vol. 26, No. 1, PP. 100-112, Feb. 2009.
[14] J Thongkam, G. Xu., Y. Zhang, F. Huang. "Breast cancer survivability via AdaBoost algorithms", *HDKM '08 Proceedings of the second Australasian workshop on Health data and knowledge management* , Vol. 80, PP. 55-64, 2008.
[5] S. Gupta, D. Kumar, A. Sharma. "Data Mining Classification Techniques Applied For Breast Cancer Diagnosis And Prognosis", Issn : 0976-5166 , *Indian Journal Of Computer Science And Engineering*, Vol. 2, No. 2, April-May 2011.
[16] T. Mamura, S. Matsumoto, Y. Kanagawa, B. Tajima, S. Matsuya, M. Furue, H. Oyama. "A technique for identifying three diagnostic findings using association analysis", *Med. Biol. Eng. Comput.*, Vol. 45, 51-59, 2007.
[17] J. Gadewadikar, O. Kuljaca1, K. Agyepong, E. Sarigul, Y. Zheng, P. Zhang,. "Exploring Bayesian networks for medical decision support in breast cancer detection", *African Journal of Mathematics and Computer Science Research*, Vol. 3, No. 10, PP. 225-231 (October 2010).
[18] A. Endo, T. Shibata, H. Tanaka. "Comparisons of Seven Algorithms to Predict Breast Cancer Survival", *Biomedical Soft Computing and Human Sciences*, Vol. 13, No. 2, PP. 11-16, 2008.
[19] "SEER Extend of Diseas, 1988: Codes and Coding Instructions, Third Edition", [Online], Available: http://seer.cancer.gov/manials/EOD10Dig.pub.pdf, [Accessed: 20-July-2013], 2014.
[20] I. Soerjomataram, M.W. J. Louwman, J.G. Ribot, J.A. Roukema, J.W.W. Coebergh. "An overview of prognostic factors for long-term survivors of breast cancer", *Breast Cancer Res Treat.*, Vol. 107, No. 3, PP. 309-330, (Feb. 2008).
[21] G. Maskarinec, J. Pagano, G. Lurie, E. Bantum, C.C. Gotay, B.F. Issell, "Factors Affecting Survival Among Women in Hawaii with Breast Cancer", *J Womens Health* (Larchmt). , Vol. 20, No. 2, PP. 231-237, (Feb. 2011).
[22] K.T. Bradley. "Prognostic and Predictive Factors in Breast Cancer", [Online], Available: http://www.cap.org, [Accessed: 10-March-2013], 2007.
[23] L.A. Newman. "Breast Cancer in African-American Women", *Oncologist*, [Online], Available: http://theoncologist.alphamedpress.org/content/10/1/1.full , [Accessed: 01-Sep.-2013], (July 2004),
[24] Claudia R. Baquet, Shiraz I. Mishra, Patricia Commiskey, Gary L. Ellison, Mary DeShields, "Breast Cancer Epiudemiology in Blacks and Whites: Disparities in Incidence, Mortality, Survival Rates and Histology., *Journal of the National Medical Association*, Vol. 100, No. 5, PP. 480-488, May 2008.
[25] "American Cancer Society.Breast Cancer Facts & Figures 2011-2012", [Online], Available: http://www.cancer.org/, [Accessed: 01-Oct-2013]
[26] Owrang O., M.M., Hosseinkhah, F., *2013. Association Rules Mining for Breast Cancer Survivability Prediction, Proceedings of the 28th International Conference on Computers and Their Applications (CATA-2013)*, PP. 159-165, March 4-6, Honolulu, Hawaii, 2013.
[27]StatCrunch,[Online],Available:http://www.statcrunch.com, [Accessed: 10-Oct.-2013].
[28] Mary L. McHugh, "The odds ratio: calculation, usage, and interpretation", Biochemia Medica, The journal of Croatian Society of Medical Biochemistry and Laboratory Medicine, [Online], Available: http://www.biochemia-medica.com/content/odds-ratio-calculation-usage-and-interpretation, [Accessed: 01-Oct-2013].
[29] David S. Moore, George P. McCabe, Bruce A. Craig, "Introduction to the Practice of Statistics, 7th Edition", *W.H Freeman and Company*; 2012

A Proposed Data Mining Model for the Associated Factors of Alzheimer's Disease

Dr. Nevine Makram Labib and Mohamed Sayed Badawy
Department of Computer and Information Systems
Faculty of Management Sciences,
Sadat Academy for Management Sciences
Corniche El Nil, Maadi, Cairo, Egypt
nevmakram@gmail.com

Abstract— *Data mining (DM) may be viewed as the process of detecting and finding knowledge within data warehouses using of a set of analytical and intelligent tools. In this study, we focus on the use of DM techniques for the discovery of the associated factors of Alzheimer's disease (AD), which is a progressive brain disorder that causes a gradual and irreversible loss of some brain functions, including memory and language skills in addition to the loss of the ability to care for oneself. In order to do so, we make use of two techniques namely Naïve Bayes and Decision trees. It was found that the most accurate classification was reached through Decision Tree technique followed by Naive Bayes. Association rules technique was then used to identify the links between different features, and determine the strength of each relationship. Some of the most important associated factors discovered are gender, age group, attention level, education level, and occupation. Future opportunities to be explored by interested researchers may be adding other data mining techniques, such as Genetic Algorithm, to predict the causes of Alzheimer's disease, adding patient data extracted from MRI and/or CT scan in order to get more accurate results and finally using the output of the model in the development of an expert systems for the diagnosis of the disease.*

Keywords: *Data Mining, Naïve Bayes, Decision tree, Association rules, Alzheimer's disease.*

I. INTRODUCTION

Data mining is a process that aims at detecting and finding knowledge within data warehouses through the use of a set of analytical and intelligent tools. It has been used in different areas such as Medicine, with the purpose of improving medical diagnosis, detecting the causes of the diseases, and predicting the patient's health condition in the future. Examples of such applications are the early diagnosis of cancer, and the automated measurement of the weakness of the work and functions of the heart. Other applications address mental illness such as dementia. This study focuses on the use of data mining techniques for the discovery of the associated factors of Alzheimer's disease.

1.1 Background

Alzheimer disease (AD) is a progressive brain disorder that causes a gradual and irreversible loss of higher brain functions such as memory, language skills, and perception of time. This leads eventually to the loss of the ability to care for oneself. It is one of the most common causes of the loss of mental functions in people over the age of 65, as in this age, 5 % to 10 % have Alzheimer's, and this proportion increases to about 10 % to 15 % among those in their 70s and to 30 % to 40 % among people 85 years of age or older [1]. It is a devastating disease because those who suffer from it experience frustration, anger, and fear as the disorder begins to take away their abilities and memories. Hence, it affects not only the patients, but also those who love and care for them as they suffer immeasurable pain and stress watching the disease slowly taking their loved ones from them [2].

1.2 Problem of the Research

The research problem lies in:

1. The difficulty of identifying the real causes of AD.
2. The inability to predict the health status of the patient and calculate the extent to which the patient is suffering from this disease.
3. The difficulty of identifying the relationship between Alzheimer's disease and other diseases.

1.3 Research Objectives

The research aims at:

1. Discovering the associated factors of Alzheimer's.
2. Predicting the rate of Alzheimer disease for a particular patient.
3. Comparing between different data mining techniques in the diagnosis of the disease.

1.4 Importance of the Research

The diagnosis of Alzheimer's reflects the doctor's best judgment about the causes of a patient's symptoms, based on the performed tests. An early diagnosis may help individuals receive treatment for symptoms and gain access to programs and support services. This will enable them to take part in decisions concerning care, living arrangements, money and legal matters. A timely diagnosis often allows the patient to participate in this planning and to decide who will make medical and financial decisions on his or her behalf in later stages of the disease.

II. RECENT STUDIES OF DATA MINING DEALING WITH ALZHEIMER'S DISEASE

This section of the study sheds light on selected recent studies addressing the problem domain.

2.1 Recent Studies

A research paper proposed a novel sparse inverse covariance estimation algorithm that discovers the connectivity among different brain regions for Alzheimer's study [3]. The proposed algorithm can incorporate the user feedback into the estimation process, while the connectivity patterns can be discovered automatically. Experimental results on a collection of FDG-PET images demonstrate the effectiveness of the proposed algorithm for analyzing brain region connectivity for Alzheimer's disease study.

Another research presented a novel technique, based on association rules, that is used to find relations among activated brain areas in single photon emission computed tomography (SPECT) imaging [4]. The aim of this work was to discover associations among attributes which characterize the perfusion patterns of normal subjects and to make use of them for the early diagnosis of Alzheimer's disease. The proposed methods were validated by means of the Leave-one-out cross validation strategy, yielding up to 94.87% classification accuracy, thus outperforming recent developed methods for computer-aided diagnosis of Alzheimer's disease.

A third study proposed various models for the classification of different stages of Alzheimer's disease by considering the different cognitive tests, physical examinations, age, neuropsychiatry assessments, mental status examination and laboratory investigations [5]. These methods included Neural Networks, Multilayer Perceptron, Bagging, Decision Tree, CANFIS and Genetic algorithms. The classification accuracy for CANFIS was found to be 99.55% which was better when compared to other classification methods.

2.2 Results of the Review

Based on the previous review of some studies related to the problem domain, it is concluded that there are several data mining techniques that proved to be successful in the early diagnosis of the disease. The most important of these techniques are Decision Trees, Naïve Bayes, Association Rules, and Neural Network Classifier.

III. DESCRIPTION OF THE PROPOSED DATA MINING MODEL FOR THE ASSOCIATED FACTORS OF ALZHEIMER'S DISEASE

This section provides the description of the proposed model that makes use of data mining techniques in order to discover the associated factors of Alzheimer disease.

First, we will start by developing two models; each of them depends on a specific data mining technique namely Naïve Bayes and Decision Trees, in order to recognize the most influential attributes of the disease. *Second*, attributes extracted from the previous models will be considered as inputs to another model that makes use of Association Rules technique, to determine the relationships between the attributes and their strength regarding the state of the disease.

The proposed data mining model consists of the following stages:

3. 1 Data Collection

Data have been compiled from more than one source as follows:-

3.1.1 Sources:-

A-Textbooks: medical books and references specialized in Alzheimer's disease (AD).

B- Patient Records: extracted from 'Dar Ome we Abe', that provides care for elderly people with Alzheimer's disease, and the educational hospital of Alexandria University.

3.1.2 Methods:-

A- Literature Review of AD to find out the relevant factors and the relationships between them.

B- Structured Interviews with Geriatricians who work in governmental hospitals and private medical centers.

3.2 Data Purifying

In this step, all the missing values were replaced by the arithmetic mean or the mode with respect to that attribute, and all incorrect or non-clear data were excluded.

3.3 Data Selection

Only 45 attributes have been selected from the patients' files based on the recommendation of the geriatricians. Then, the mining techniques were applied to these specific data items in order to reach the ones that are of interest for the domain.

3.4 Data Integration

The data was integrated into one structure as the sample was collected from various sources and formats including text, Excel, and Microsoft database access format.

3.5 Data Mining Tool

The database was built using *SQL Server Management Studio 2008.*This software was selected specifically because of its compatibility with SQL Server Business Intelligence Development Studio. The database was then tested and validated after undergoing 11 stages that resulted in the successful transfer of *868 rows.*

As for the data mining tool, *Microsoft Visual Studio2008* was used since it provides a full set of easy to use, graphical administration tools for creating, configuring and maintaining databases, data warehouses, and data marts.

3.6 Data Mining Techniques

The selected techniques include Decision Trees, Association Rules, and Naïve Bayes.

IV. Exploring the Data Mining Models

4.1 Decision Trees Model

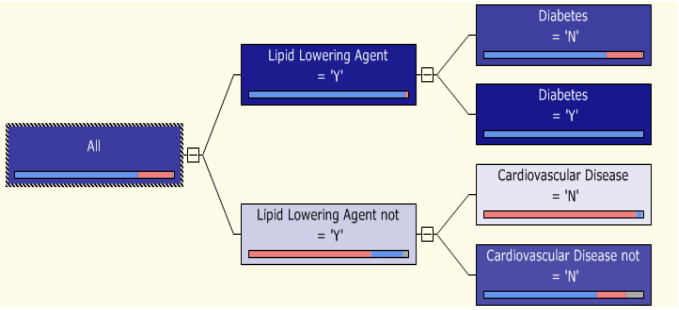


Figure 1. Decision Tree Model

Using the decision tree model, it was found that:
1 - Decision Tree of the diagnosis of Alzheimer's disease consists of three levels, each level is a tipping point to split the data into two parts.
2 - A set of effective attributes in the diagnosis of Alzheimer's disease, are "Lipid Lowering Agent", "Diabetes", and "Cardiovascular Disease".
3 - There is a very strong relation between the incidence of the disease and the presence of "Lipid Lowering Agent = Y" and "Diabetes = Y" together.
4- There is a very strong relation between incidence of the disease and the presence of "Lipid Lowering Agent not = Y" and "Cardiovascular Disease = N" together.

4.2 Naïve Bayes Model

4.2.1 Dependency Network

Following is the dependency network that shows the attributes that have an impact on the diagnosis.

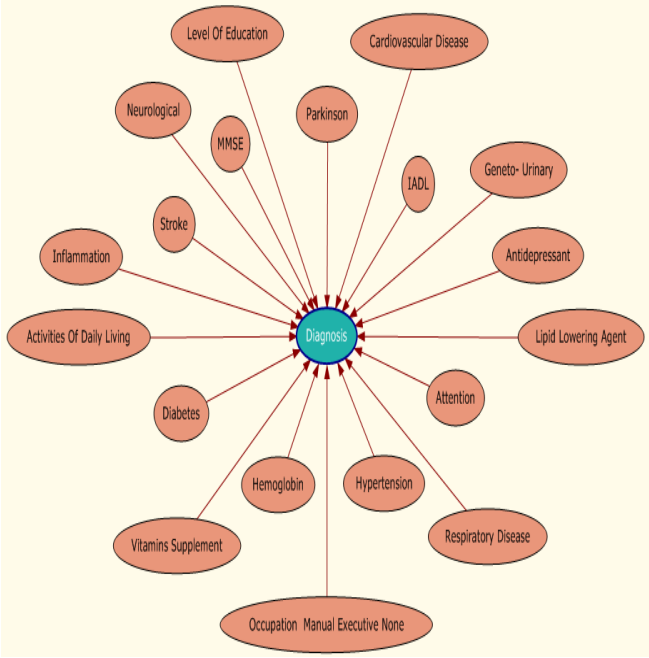


Figure 2. Dependency Network for Naive Bayes Model

4.2.2 Attribute Characteristics

The following figure shows the arrangement of attributes using the percentage of the probability that occur in Diagnosis = Y. These attributes have been arranged in descending order according to the probability of patients suffering from the disease. It has been observed that there is a set of features that occupies the first rank in the probability of the disease. It includes Vitamins Supplement = N, Lipid Lowering Agent = Y, Diabetes = Y, Inflammation = N, Geneto- Urinary = N, Stroke = Y, Respiratory Disease = Y, Cardiovascular Disease = Y, Parkinson = Y, Hemoglobin = Y, Antidepressant = Y, Neurological = Y.

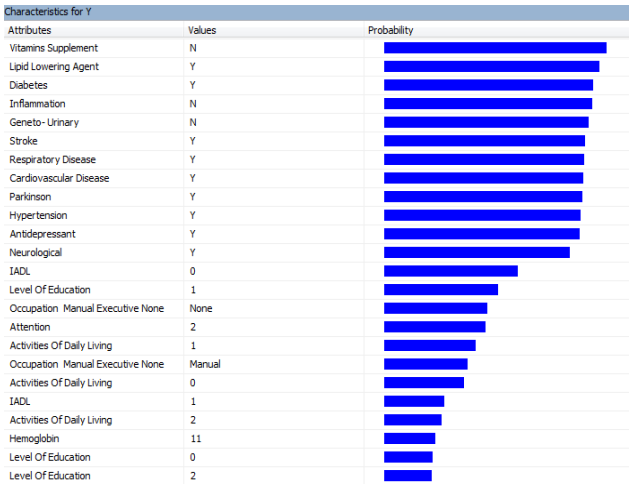


Figure 3. Attribute Characteristics for Diagnosis = 'Y'

V. Validating Model Effectiveness

The effectiveness of both models was tested using two methods: Lift Chart and Classification Matrix. The purpose was to determine which model gave the highest percentage of correct predictions for diagnosing patients with Alzheimer's disease.

5.1 Lift Chart with Predictable Value

To determine if there was sufficient information to learn some patterns in response to the predictable attribute, columns in the trained model were mapped to those in the test dataset. The model, predictable column to chart against, and the state of the column to predict patients with AD were also selected. The following Lift Chart shows the comparison between the different models.

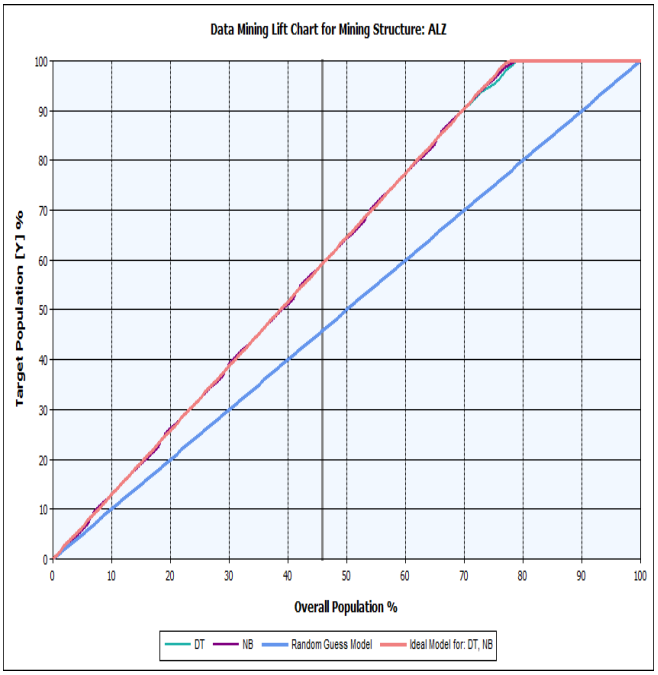


Figure 4. Data Mining Lift Chart for Mining Structure

This chart includes the two models using the same data; the x-axis of the chart represents the percentage of the test dataset that is used to compare the predictions while the y-axis of the chart represents the percentage of predicted values. To determine if there was sufficient information to learn patterns related to the predictable attribute, columns in the trained model were mapped to columns in the test dataset. The top red line shows the ideal model; it captured 100% of the target population for patients with Alzheimer's disease using 60% of the test dataset, the bottom blue line shows the random line which is always a 45 degree line across the chart. It shows that if we randomly guess the result for each case, 50% of the target population would be captured using 46% of the test dataset. Two models line (green represents Decision Trees Model and purple represents Naïve Bayes Model) fall between the random-guess and ideal model lines, showing that both two have sufficient information to learn patterns in response to the predictable state.

5.2 Statistics for the Comparison of Models

Following is a figure that shows a set of statistics for a comparison between the different models.

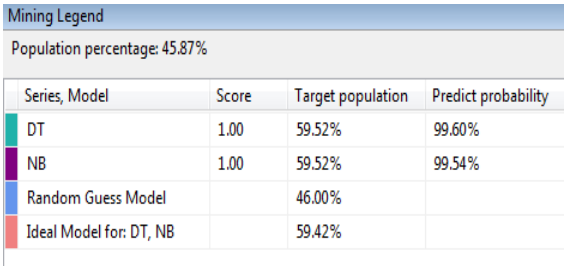


Figure 5. Comparison between the Different Models

The data was interpreted in the form that is suitable for the Decision Trees technique in order to receive a 1.00 in the score column. Moreover, it got a 99.60% in the Predict Probability column, followed by Naïve Bayes technique that has a 1.00 in the column Score and earned 99.54% in column Predict Probability. It also got both of the two models a 59.52% in the column Target Population. Therefore, it is closer to the ideal solution.

Once the testing phase of the models is complete, and their validity is ensured, it is followed by the stage of extracting the factors affect the diagnosis of Alzheimer's disease, as in the following table:-

Table 1. Factors Affecting the Diagnosis of AD

Attributes	
Vitamins Supplement	Lipid Lowering Agent
Diabetes	Geneto- Urinary
Stroke	Cardiovascular Disease
Parkinson	Hypertension
Antidepressant	Neurological
Attributes	
Attention	Activities Of Daily Living
Level Of Education	Occupation Manual Executive None
Respiratory Disease	IADL
Inflammation	Hemoglobin

To finalize the knowledge discovery process, another model is developed. It aims at identifying the extent of correlation of these features with a certain diagnosis. It has as input the features extracted from the previous model.

VI. ASSOCIATION RULES MODEL

6. 1 Rules

The rules represent the qualified association rules.The rule grid displays all qualified rules, their probabilities, and their importance scores. The importance score is designed to measure the usefulness of a rule, the higher the degree of importance of this gives credence to the rule .

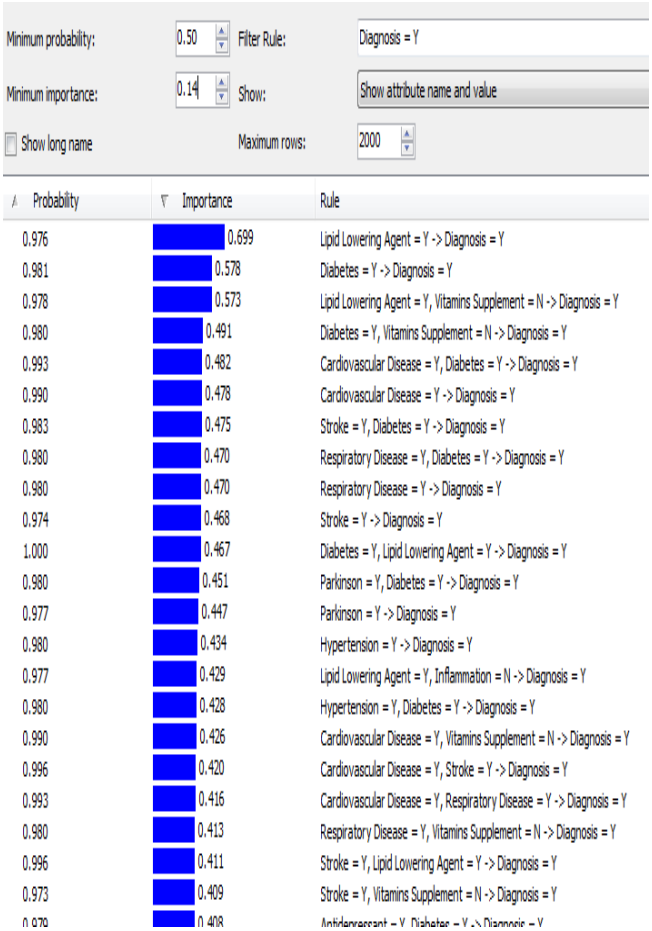


Figure 6. Rules of the Association Rules Model

The following table shows the set of rules produced by the model, which explains the power of relationship between different Attributes with Diagnosis = 'Y'.

Table 2. Relationship between Different Attributes with Diagnosis = Y

Rule	Importance	Probability
Lipid Lowering Agent = Y -> Diagnosis = Y	0.858	**0.979**
Lipid Lowering Agent = Y, Vitamins Supplement = N -> Diagnosis = Y	0.673	**0.982**
Diabetes = Y -> Diagnosis = Y	0.608	**0.991**
Diabetes = Y, Vitamins Supplement = N -> Diagnosis = Y	0.519	**0.990**
Diabetes = Y, Lipid Lowering Agent = Y -> Diagnosis = Y	0.518	**1.000**
Cardiovascular Disease = Y, Diabetes = Y -> Diagnosis = Y	0.518	**1.000**
Cardiovascular Disease = Y -> Diagnosis = Y	0.513	**0.997**
Stroke = Y, Diabetes = Y -> Diagnosis = Y	0.503	**0.990**
Lipid Lowering Agent = Y, Inflammation = N -> Diagnosis = Y	0.496	**0.981**
Antidepressant = Y -> Diagnosis = Y	0.486	**0.975**
Stroke = Y -> Diagnosis = Y	0.484	**0.978**
Hypertension = Y -> Diagnosis = Y	0.470	**0.987**
Hypertension = Y, Diabetes = Y -> Diagnosis = Y	0.468	**0.990**
Parkinson = Y, Diabetes = Y -> Diagnosis = Y	0.468	**0.990**
Respiratory Disease = Y, Diabetes = Y -> Diagnosis = Y	0.468	**0.990**
Respiratory Disease = Y -> Diagnosis = Y	0.468	**0.990**
Parkinson = Y -> Diagnosis = Y	0.463	**0.987**
Antidepressant = Y, Diabetes = Y -> Diagnosis = Y	0.461	**0.990**
Cardiovascular Disease = Y, Vitamins Supplement = N -> Diagnosis = Y	0.452	**0.997**

The following figure shows the link of a group of attributes with diagnosis value = 'Y'.

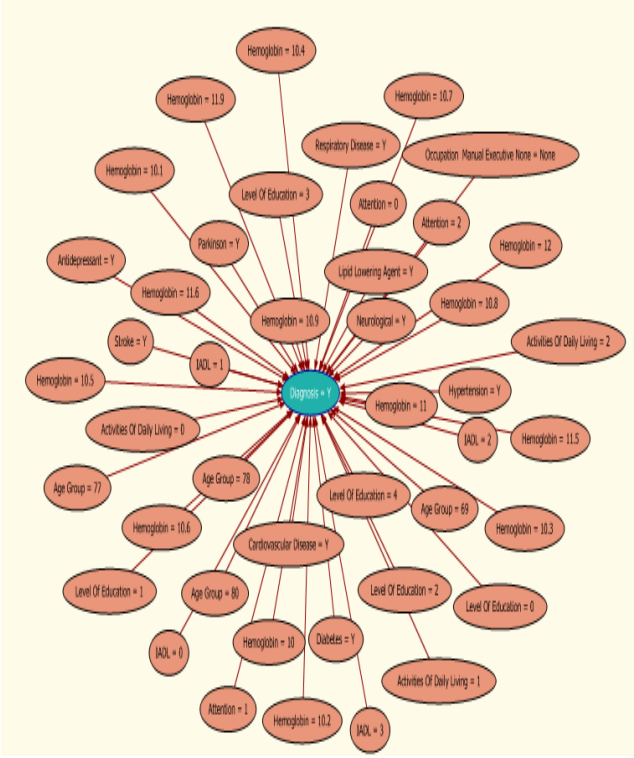


Figure 7. Link to a Set of Attributes to the Disease

6.2 Mining Model for Prediction

6.2.1 Prediction Using the Data Used in the Sample

First, the model is determined based on Decision Tree technique. The following table shows the outcome of prediction.

Table 3. Prediction Results

No	Gender	Age Group	Level of education	Diabetes	Hypertension	Lipid Lowering Agent	Activities of daily living	Diagnosis	Expression
1	Female	66	10	N	N	N	6	None	0.950084602368866
2	Female	65	1	Y	Y	Y	1	Y	0.99596570852441
3	Male	67	2	Y	N	Y	0	Y	0.99596570852441
4	Male	62	9	N	N	Y	5	Y	0.756862745098039
5	Female	75	3	N	N	Y	2	Y	0.756862745098039
6	Female	63	11	N	N	Y	5	Y	0.756862745098039
7	Female	80	1	Y	Y	N	1	Y	0.699745547073791
8	Female	88	2	Y	Y	Y	2	Y	0.99596570852441
9	Female	73	3	Y	Y	Y	1	Y	0.99596570852441
10	Female	65	1	N	Y	N	0	None	0.950084602368866

The previous table shows the probability of occurrence or non occurrence of the disease in 50% of the patients, using Decision Trees.

6.2.2 Prediction Using the Extracted Data

In this step, the diagnosis of the condition is predicted through the use of the three models as shown in the Singleton Query Input dialog box, whose columns are mapped to the columns in the mining model.

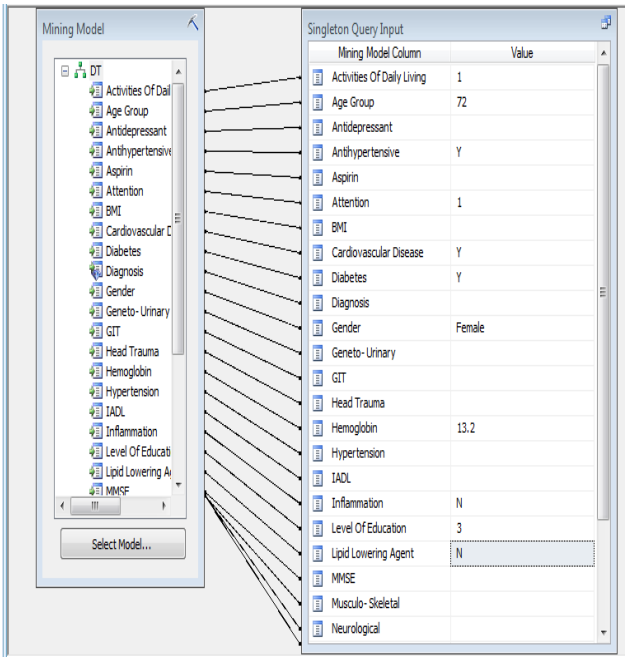


Figure 8. Singleton Query Input Dialog box

The previous figure shows the prediction of having the disease based on the use of some input data about a certain patient and extracting the rest of the data using the three models, based on the trained system.

As for the prediction probability of each of the three models used on the other 50%, they are illustrated in the following tables.

Table 4. Prediction Result of Decision Tree Model

Predict Probability	Diagnosis
0.699745547073791	Y

Table 5. Prediction Result of Naïve Bayes Model

Predict Probability	Diagnosis
0.583410138248848	Y

Table 6. Prediction Result of Association Rules Model

Predict Probability	Diagnosis
0.491304347826087	Y

6.3 EVALUATION OF DATA MINING OBJECTIVES

Three objectives of data mining were defined based on both the exploration of Alzheimer's disease dataset and the objectives of this research. They were evaluated against the trained models. Results showed that all three models had achieved the stated objectives, suggesting that they could be used to provide decision support to medical doctors for diagnosing patients and discovering medical factors associated with Alzheimer's Disease.

The objectives were as follows:

First objective was to discover the significant influences and relationships in the medical inputs associated with the predictable state Alzheimer's disease. The Dependency viewer in Association Rules, Decision Trees and Naïve Bayes models showed the results from the most significant to the least significant medical predictors. Medical Doctors can use this information to further analyze the strengths and weaknesses of the medical attributes associated with Alzheimer disease.

Second objective was to predict those who are likely to be diagnosed with Alzheimer disease, given patients' medical profiles. It was found that all models were able to perform this task using singleton query, which made use of single input cases and multiple input cases respectively, and also to show the rate of the disease.

Third objective: was to compare between the different data mining techniques. It was found that Decision Tree model was the best in the diagnosis process and Naïve Bayes model was the best in identifying the characteristics of patients with Alzheimer's disease and showing the probability of each input attribute for the predictable state.

VII. CONCLUSION AND FUTURE WORK:

7.1 Conclusion

The main purpose of this study was to build a model that has the ability to discover the associated factors of Alzheimer's disease in order to provide a better diagnosis and prognosis. The most important points that have been reached were the following:-

1. Decision Tree technique was able to provide more accurate results than Naive Bayes.
2. Using association rules technique was very useful in identifying the links between different features and determining the strength of each relationship.

7.2 Future Work

Based on the previous conclusions and a number of issues that arose during the study, some topics may be considered as future opportunities to be explored by interested researchers. They are the following :

1. Using additional data mining techniques, such as Genetic Algorithms in the predicting phase of Alzheimer's disease.
2. Adding patient data related to MRI and/or CT scan in order to get more accurate results.
3. Using the output of the model in the development of an expert system for the diagnosis and prognosis of the disease.

ACKNOWLEDGEMENT

The researchers would like to thank the medical doctors and administrative staff who provided them with the required data and knowledge that helped conducting the study.

REFERENCES

[1] E. Floyd, Bloom, B. M. Flint., D. J. Kupfer; the Dana Guide to Brain Health: A Practical Family Reference from Medical Experts, Simon and Schuster publisher, 2006.

[2] C. L. Linda, H..Juergen, and M. D. Bludau; Alzheimer's Disease, ABC-CLIO Publisher, September 2012.

[3] S. Liang, P. Rinkal, L. Jun, C. Kewei, W. Teresa, and L Jing; "Mining Brain Region Connectivity for Alzheimer's disease Study via Sparse Inverse Covariance Estimation," KDD '09 Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1335-1344, 2009.

[4] R.Chaves, J. M. Górriz., J. Ramírez., I. Aillán., D. Salas-Gonzalez, and M.Gómez-Río; "Efficient Mining of Association Rules for the Early Diagnosis of Alzheimer's Disease," Physics in Medicine and Biology, 21, 56(18): 6047-63. doi: 10.1088/0031-9155/56/18/017. Epub Aug 26, 2011.

[5] L. S Joshi, V. Simha , D. Shenoy, K. R. Venugopal, and L. M. Patnaik; "Classification and Treatment of Different Stages of Alzheimer's Disease Uusing Various Machine Learning methods," International Journal of Bioinformatics Research; 2010, Vol. 2, Issue 1, p. 44.

Automated Statistical Data Mining of a Real World Landslide Detection System

A. Divya Pullarkatt[1], B. Geethu Thottungal[2], C. Geethalekshmy V[3] and D. ManeeshaVinodini Ramesh[4]

[1,4]Amrita Center for Wireless Networks and Applications, Amrita Vishwa Vidyapeetham, Clappana P.O., Kerala, India

2,3Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Clappana P.O., Kerala, India

Abstract—*Wireless sensor network for landslide detection deployed in Munnar consists of 150 geophysical sensors which are spatially distributed over 20 Deep Earth Probes (DEP) located at different areas in the deployment site. The data received from each of these heterogeneous sensors are mined to retrieve the correlation between the various parameters contributing to landslide, using appropriate statistical methods. This paper presents an architecture which we have developed for automatic data mining of landslide data which will ultimately help in issuing an early warning for occurrence of landslides. Several algorithms were developed towards achieving this objective of effective data analysis of the continuous real time data collected from the deployment field. The results show that the slope instability in a region is dependent not only the intensity of rainfall but also the antecedent rainfall conditions and soil layer parameters. Each of these different algorithms and its results are explained in detail in this paper.*

Keywords- Landslide, statistical data analysis, pore pressure, rainfall rate, slope instability.

1. Introduction

Landslide is the third most deadly natural disasters on earth and it can be triggered by gradual processes such as weathering, or by external mechanisms like soil erosion, prolonged rainfall, earthquakes, floods, morphological factors like slope angle, physical factors like volcano eruptions etc. Reports show that 17% of disasters occur due to landslides which lead to considerable loss of life and damage to communication routes, human settlements, agricultural fields and forest lands [1]. So it is important for us to early warn the occurrence of landslides for saving lives, property, etc. For detecting landslides, Amrita Center for Wireless Networks and Applications has developed and deployed world's first ever wireless sensor network for landslide detection in Munnar, Kerala, India. Huge amount of data are received in the Data Management Center at the Amrita University from various sensors deployed in the DEPs for monitoring those factors contributing to landslide.

Munnar, the south-east area of Kerala experience several types of landslides, of which debris flows are the most common. Studies conducted in the area indicates that pro-longed and intense rainfall or more particularly a combination of the two and the resultant pore pressure variations are the most important trigger of landslides in that area [8]. Suitable statistical techniques have been employed for analyzing the inter-dependability of these factors and thereby effectively determining the pattern for pore pressure buildup. Based on Richard Iverson Equation [6],[11], permeability, hydraulic conductivity and depth are some important factors which influence the pore pressure build up in soil. Hence our dataset contains information about pore pressure, rainfall rate, depth of each sensor in the DEP, permeability, hydraulic conductivity, antecedent conditions of rain, etc. We have developed algorithms incorporating relevant statistical techniques which will take these parameters as its input, for identifying the correlation between them. These data mining algorithms form an integral part of the automatic data analysis system which we have designed for the Landslide detection system in Munnar.

The remainder of the paper is organized as follows. Section II describes related works in data analysis. Section III delineates the heterogeneous data from landslide detection system. Section IV explains the architecture of automatic data analysis. Section V presents the statistical analysis and the algorithms developed. Section VI deal with the experimentation and evaluation of the proposed algorithms. Finally, we conclude and outline the future work in Section VII.

This work has been partially funded by "Monitoring and Detection of Rainfall Induced Landslide using an Integrated Wireless Network System" project funded by Department of Science and Technology (DST), India and also by "Advanced Integrated Wireless Sensor Networks for Real-time Monitoring and Detection of Disasters" project funded by Ministry of Earth Science, Government of India.

2. Related work

Large numbers of sensors are deployed in Wireless Sensor Network for Landslide Detection at Munnar and the correlation between these huge data plays an important role in landslide detection. Developing algorithms for determining these correlations by analyzing real world data is a challenging job.

Similar work has been identified to be done in cases like thunder storm prediction [2] and slope instability analysis [3]. Paper [2] presents the multiple correlations in the data from thunderstorm and application of these correlations in the prediction of seasonal severe thunderstorms using K-nearest neighbor (K-nn) method, and modified K-nn method. Recently, object-oriented analysis (OOA) is implemented on the data from Light Detection and Ranging (LiDAR) for the landslide identification [4]. Another study is done to examine the long-term parameters in order to define their relations to landslide occurrence [5]. Paper [15] explains statistical analysis of landslide area in the Daunia area, Italy using data from GIS technology. Linear regression method is used for produced a reliable susceptibility map of the investigated area. Topographical/geological data and satellite images were collected and processed using GIS and image processing tools and these data were analyzed and used for landslide hazard mapping in Cameron Highland, Malaysia[16]. Soil properties and rain fall rate are also important factors in their analysis. Basically studies have been done on image data from landslide area which may not be an accurate method for prediction. Complexity of our work lies on the fact that our work deals with the continuous real data from 2009 onwards to till date. Also it is difficult to generalize the factors affecting pore pressure because the soil properties vary with different locations.

Since the deployment area is known for rainfall induced landslides, the proposed data analysis methods concentrate on the analysis of rainfall rate and pore pressure values and find the impact of rainfall on pore pressure buildup.

3. Heterogeneous data from landslide detection system

A wireless sensor network (WSN) for landslide detection is deployed in Munnar. The system is used for continuous monitoring of environmental, geological and hydrological parameters that may trigger a landslide. The real-time data from this system is used for early warning of landslides.

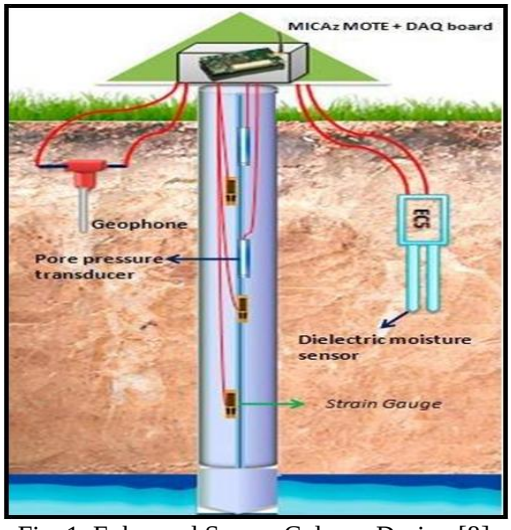


Fig. 1. Enhanced Sensor Column Design [8]

The rainfall induced landslides can be triggered due to heavy or continuous rainfall, increased soil moisture content, rise in pore pressure, excessive weathering, earth quake, toe removal by humans etc. Heterogeneous sensors such as rain gauges, moisture sensors, pore pressure sensors, strain gauges, tilt meters, geophone etc., are deployed to monitor the above parameters. These sensors are attached to a Deep Earth Probe (DEP) as shown in Figure 1.

Currently 20 DEPs are deployed in the landslide prone area as shown in Fig 2. The spatial variability of DEP deployment is dependent on the spatial vulnerability of landslides. Hence the location of DEP deployment is dependent on the site specific geological conditions.

This landslide detection system has a total of 150 geophysical sensors connected to these 20 DEPs. The sensors connected in each DEP is determined by considering various factors such as soil layer structure, soil properties, different parameters to be monitored in each soil layer etc. Hence each DEP consists of multiple types of heterogeneous sensors at different soil layers, at different depths. This contributes to spatial variability of the features and hence each DEP gathers unique information. This system provides the opportunity to collect the real-time data dynamically in any frequency. Currently the sensor data is collected at the rate of one sample per minute and stored in the database.

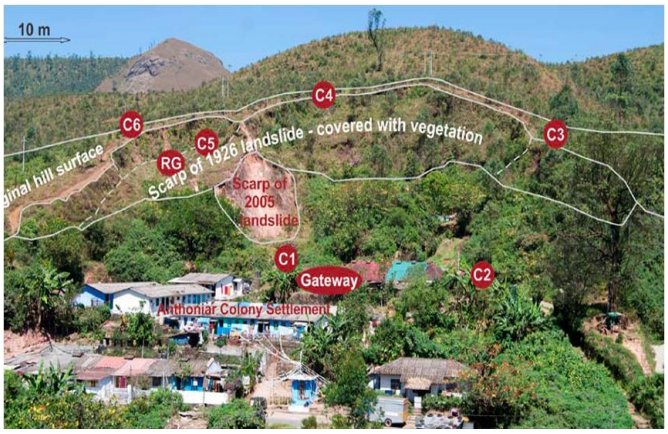


Fig. 2. Locations of the DEPs and Rain Gauge (RG) [9]

4. Architecture of automatic data analysis system

For better forewarning of landslides, identification of inherent relation between landslide parameters are necessary. One of the objectives of this work is to identify the relation between the data from the Wireless Sensor Network System for Landslide Detection deployed at Munnar. Using the knowledge gained from the data mining and data analysis, this system will be equipped to predict the risk levels of landslides and issue early warnings to the community at the deployment area.

For the current deployment, the geophysical sensors are spatially distributed based on many factors such as the number of soil layers, layer structure, soil properties and

variability, hydraulic conductivity of the soil layers, the presence of impermeable layers, the water table height, the bed rock location, depth of the bore hole for deploying the DEP, and the specific deployment method required for each geophysical sensor. Hence the data received from these sensors has to be analyzed considering some of the above factors.

The architecture diagram for automatic data mining and analysis is shown in Figure 3.The heterogeneous data received from multiple sources like WSN, terrain mapping, soil properties, and weather forecast are collected in the database. The pore pressure values and rainfall rate are the main contribution of WSN whereas the terrain mapping gives the spatial data. Soil properties include information such as permeability, hydraulic conductivity, etc. and the data from these multiple sources along with the weather forecast data is used for the learning purpose. The entire data is stored in a database. This data is analyzed using different statistical algorithms. The output of data analysis is given to the knowledge base and also as input to the learning algorithms. The pore pressure buildup is predicted on a spatio-temporal basis after the training and testing process in the learning phase by using a suitable machine learning algorithm. The Factor of Safety (FoS) is then calculated from the predicted pore pressure values. The occurrence of landslide is specified using different risk levels which is based on the calculated FoS value. Based on this risk levels, alert dissemination to the local community is done.

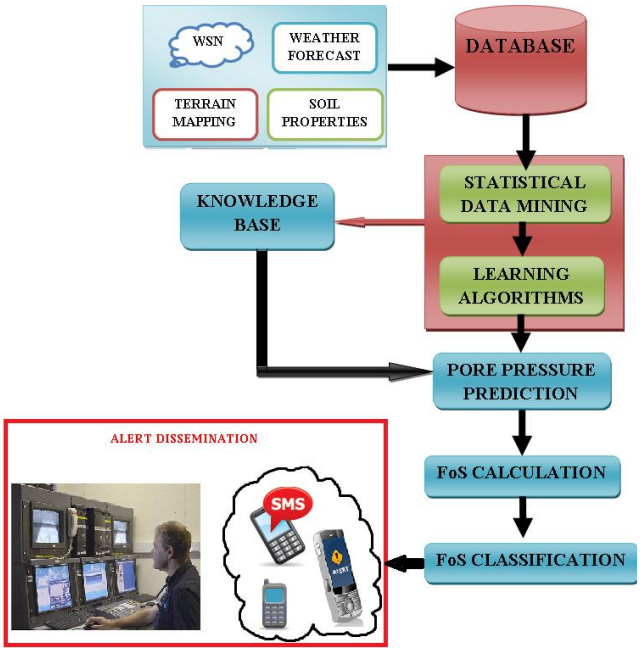


Fig. 3. Proposed System

5. Statistical analysis and its algorithms

Slope instability in rainfall induced landslides is majorly triggered by continuous rainfall or instant heavy rainfall. As the rainfall increases, the pore pressure at each soil layer will build up with respect to the soil structure, its soil properties, vegetation etc. As the pore pressure increases, the cohesion between the soil particles decreases, leading to slope instability. The slope instability [7] can be determined by calculating the FoS of the slope [19].

$$FoS = \frac{C + (\sigma - \mu)\tan\phi}{\gamma z \cos\beta \sin\beta} \quad (1)$$

$$= \frac{C + (\gamma - m\gamma w)\, z \cos\beta \cos\beta \tan\phi}{\gamma z \cos\beta \sin\beta} \quad (2)$$

In equations (1) and (2), C is the cohesion in kN/m^2, γ is the unit weight of slope material in kN/m^3, γ_w is the unit weight of water in kN/m^3, z is the thickness of slope material in m, m is the vertical height of water table above the slide plane, and is dimensionless, β is the slope of the ground surface in degrees, φ is the internal angle of friction in degrees, σ is the normal stress, and μ is the pore water pressure. The increase in rainfall causes increase of pore pressure, leading to the decrease of FoS of the slope. The chances of slope instability increases as the FoS become less than 1.

The conditions leading to the instability of the soil layer has to be continuously monitored, analyzed and the inference has to be derived for an effective early warning system. Hence it is highly necessary to continuously analyze the data received from rain gauge and pore pressure sensors and derive the state of the hill slope. For achieving this objective we have developed several algorithms for data mining the continuous data collected from the deployment field.

Initially data cleaning is done in the input dataset in order to remove the noisy data. In this phase the historic data will be mined to remove any outliers and other noisy data. This cleaned data is then stored in the database.

To determine the effect of rainfall on inducing landslides, it is highly necessary to determine the correlation among the heterogeneous sensor data and identify patterns leading to slope instability. In this work, we are mainly concentrating on the impact of rainfall rate on spatio-temporal pore pressure build up in the landslide prone area. For determining patterns of rain and effect of rain on pore pressure build up, the following algorithms are developed to perform the data mining.

5.1 Classification of rainfall

Instant rainfall conditions along with antecedent rainfall conditions lead to slope instability. The research papers [10], [14] describe the effect of antecedent rain on pore pressure build up. Hence thorough analysis of rainfall data is essential to determine the patterns of rainfall. The algorithm for classification of rainfall is described below. This algorithm consists of three phases as described below.

- Rainfall rate per day: The cumulative value of rainfall data will be determined from the available rainfall data per minute with respect to each day.
- Rainfall rate per multiple days: The cumulative rainfall data for one day, two day, up to 15 days are found and stored in the database with respect to each day.
- Rainfall classification: By using existing classification techniques, the results of the above phases such as rainfall rate per day and rainfall rate per multiple days, will be used to classify the rain data as instant heavy rain, continuous rain, continuous heavy rain, and no rain.

The classification of rain data is used for determining the effect of each class of rainfall on pore pressure build up.

Algorithm1: Analysis of rainfall

Input: Rainfall in millimeter from database
Output: Date of rainfall, cumulative rain data in millimeter, and classes of rain.

1. Retrieve the rainfall_rate from database.
2. *count_of_rainy_day* = 0
3. For each day:
4. cumulative_rainfall=$\sum_{i=1}^{p}$ rainfall_rate
5. *IF* cumulative_rainfall=0
   THEN label="No_rain"
6. *ELSE*
   Total_rain=$\sum_{j=1}^{\text{count\_of\_rainy\_day}}$ cumulative_rainfall$_j$
7. *IF* count_of_rainy_day>1
   THEN label="Continuous_rain"
8. *ELSE*
   Label="Rainy day"
9. *IF* cumulative_rainfall>dynamic_threshold
   THEN Label="Instant_heavy_rain"

5.2 Spatio-temporal analysis of pore pressure data

In each DEP, multiple pore pressure sensors are deployed at different depths. The data from these pressure sensors, implanted at different locations in each DEP located at various geographic locations, are continuously collected. These data are analyzed for determining the inherent spatio-temporal correlations that exist among pore pressure values. These correlations play a vital role in early warning of the risk levels of landslide. The algorithm for spatio-temporal analysis of pore pressure data is described. This algorithm consists of three phases as described below.

- Mean of a Day: In this phase the historic data is used to determine the mean per day for pore pressure sensor values at different depths and different locations.
- Variance of a Day: In this phase the historic data is used to determine the variance per day for pore pressure sensor values at different depths and different locations.
- Maximum of a Day: In this phase the historic data is used to determine the maximum value per day for pore pressure sensor values at different depths and different locations.

Algorithm2: Behavior of pore pressure builds up of Piezometer

Input: Pore pressure values and DEP details from database
Output: Maximum, mean and variance of pore pressure values of each piezometer in a particular DEP in daily basis.

1. Retrieve DEP_Group_No from DEP details and pore_pressure_value from database.
2. For DEP_Group,
3. For i=0 to No_piezometers
4. Piezo_field_name[i]={DEP_Group_No, piezometers_name[i], depth}
5. Let Max_Pore [] =ϕ
6. For each day:
7. For j=0 to No_piezometers
8. For k=0 to No_of_pore_values
9. IF Max_Pore[k] <pore_pressure_value
10. THEN Max_Pore[k] = pore_pressure_value
11. Mean_pore=$\dfrac{\sum_{p=0}^{\text{No\_of\_pore\_values}} \text{pore\_pressure\_valuere}_p}{\sum \text{No\_of\_pore\_valuere}_p}$
12. Variance_pore=$\dfrac{\sum_{p=0}^{\text{No\_of\_pore\_values}} (\text{pore\_pressure\_valuere}_p - Mean\_pore)2}{\sum \text{No\_of\_pore\_valuere}_p}$

The results obtained from these phases are used for further statistical analysis and inference generation. The behavioral pattern of pore pressure values obtained from different depths of each DEP and also from different DEPs could be used to determine the real-time infiltration rate. This could also be used for correlating the effect of amount of rainfall, the duration of rain, soil structure and the depth on infiltration rate.

5.3 Potential vulnerable zone identification

During rain, the water will infiltrate through the permeable soil layer structure and percolate down through each of the layers. This will build up the ground water level. However soil layers differ in their permeability. In some cases intermittent soil layers can be impermeable. The water cannot seep out through the impermeable layer at the same rate as that of the permeable layer, leading to the development of perched water table. This will loosen the soil particles and cause slope instability. Hence monitoring

this phenomenon is necessary for early warning of landslides.

The maximum pore pressure value per day can be calculated using the Algorithm 2. Using this detail along with the DEP details will provide the opportunity to learn the variance of pore pressure values in a DEP. This will help us to understand how the pore pressure build up differ with respect to depth of pore pressure sensor deployment, soil structure, and soil properties. The layer at which pore pressure value is high compared to its lower layer is one of the indications of vulnerable zone. The detailed verification of the presence of vulnerable zone, and its causes are explained in the Algorithm 3.

Algorithm3: Determination of the "vulnerable zone"

Input: Maximum pore pressure data on daily basis from knowledge base
Output: Vulnerable zone

1. Retrieve the depth of piezometers from DEP details from knowledge base
2. Compare the depth of piezometers in each DEP
3. Check whether increasing order
4. Compare the maximum pore pressure value with respect to the depth order
5. IF it follows the same order
   THEN "*Normal zone*"
6. IF irregularity in pore pressure value
   THEN "*Potential Vulnerable zone*"
7. Check the soil properties
8. IF permeability low,
   THEN "*Vulnerable zone*"

5.4 Spatio-Temporal Correlation of Rain and Pore Pressure Data:

The spatio-temporal correlations of rain and pore pressure data play a vital role in changing the risk levels of landslide. The data mining results obtained using the Algorithms 1 and 2 will be used to retrieve the inherent information of slope instability process. The rainfall classification, cumulative rainfall, maximum pore pressure, mean pore pressure, and variance of pore pressure for different time scales and space scales are used to derive the knowledge and patterns necessary for early warning of landslides.

Each DEP contains a maximum of eight pore pressure sensors at different depths. The behavior of pore pressure build up depends on soil properties, position of water table, and current and antecedent conditions of rain [13]. If the pore pressure builds up goes beyond the threshold value, the cohesive force between the soils will be reduced and landslide occurs. Hence the pore pressure is analyzed with respect to rainfall rate to identify and learn the specific behavior pattern of pore pressure build up, patterns changes due to the impact of soil structure, soil properties, hydraulic properties etc.

By analyzing the outputs, we found that the pore pressure build up depends on the amount of rainfall, the duration of rain, soil structure and the depth. In order to determine the time duration which makes pore pressure build up by the impact of each rain, we had done further analysis of the data.

6. EXPERIMENTATION AND RESULTS

The real-time data from this system is continuously collected from 2009 onwards. As of March 2014, the system has continuously collected around 100Million observations, with 60 features including piezometer readings, DEP details, rainfall reading etc from 150 geophysical sensors. The complete data from the dataset is analyzed using different scenarios in order to determine the correlations between them.

As discussed earlier, this study is mainly concentrated on two factors viz. rainfall rate and pore pressure values. Those data are capable to capture the initial triggers and indications of slope instability. Hence in this work, rain gauge and pore pressure sensor's data along with soil properties are used for developing the early warnings. Thus the entire number of feature is reduced to around 30 numbers which forms the initial step of dimensionality reduction of the dataset. These observations from the reduced feature set are then aggregated using suitable statistical methods explained above for further dimensionality reduction and thus reduced to around 1lakh observations.

Using Algorithm 1, the rainfall rate of 2011 and 2012 are analyzed in order to identify the pattern and intensity of rainfall at different times in a year viz. 'no rain' season and 'rainy' season so that it can be used for analyzing the effect of the antecedent rainfall conditions on the pore pressure value. The data analysis was mainly performed by focusing on the data from these rainy periods and the correlations are analyzed based on the time duration of the 'rainy' and 'no rainy' days and also the intensity of rainfall during the 'continuous rainy' days. The cumulative rainfall and maximum pore pressure values of DEP group1 on daily basis for the period of October - November 2012 is shown in fig 5. The highest rain in this season is 24 mm and is on 8th of October.

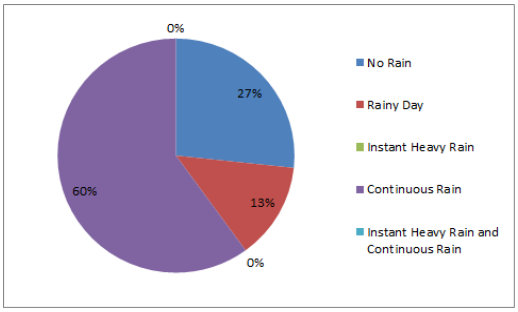


Fig.4. Rain data classification for Oct-Nov 2012

The highest values of pore pressure for *piezo3* at depth 3.3m, *piezo5* at depth 5.5m, *piezo4* at a depth of 8.5m are

22kPa on 20th of October, 140 kPa on 25th of October and 40 kPa on 24th of October. From the graph, it can be inferred that the impact of highest rainfall rate on each pore pressure value occurs after some duration as water takes some time to percolate down from one layer to the other. This is dependent on the depth of the deployment, soil structure, soil properties, and the rate of rainfall and its antecedent conditions. The value of *piezo5* and *piezo3* increase after 12-14 days where as the *piezo4* shows sudden reaction due to the rainfall.

The effect of different rain conditions such as no rain, continuous rain, and instant heavy rain etc., on the infiltration rate is shown in Fig. 6. The graph shows the pore pressure value of the piezometer *piezo1* in DEP group 5 and the rainfall rate during the period of October 2012. During this period, the highest rainfall rate of 24 mm occurred on 8th of October. There was an increase in pore pressure as a result of this rainfall and the value gradually decreased as the rainfall rate dropped. From 9th to 31st October, the rainfall value was almost negligible and it can be seen that the pore pressure value starts decreasing during that period because of the no rain state. As the soil structure vary with the depth, the water infiltration rate in each depth also varies. The results of Algorithm 2 showed that for the same rainfall rate, the effect on each piezometer is seen at different days because of the variation in infiltration rate.

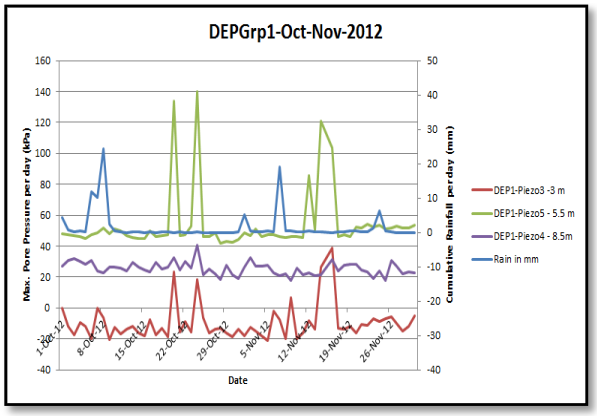


Fig.5. Rainfall rate and Pore Pressure values of DEP1 in Oct-Nov-2012

The deepest piezometer must have the highest pore pressure since it will be below the water table and according to this, the deepest piezometer i.e. *piezo4* at 8.5m must have highest value. But from Fig. 5, it can be seen that *piezo5* at 5.5m shows highest values than *piezo4* and it may be due to the presence of some other water source at 5.5m. Thus the result of Algorithm 3 shows the presence of a vulnerable zone at the position of pore pressure sensor 5 which shows the highest value among all the sensors deployed in that location.

The spatio-temporal analysis of pore pressure value and rainfall rate are shown in figure 7. It contains 5 DEP groups located at different locations and piezometers are deployed in those DEPs at different depths. 5 piezometers are deployed in DEP group1, DEP group 2 contains 3 piezometers, DEP group 4 contains 1 piezometer and DEP group 5 contains 3 piezometers. The graph shows the cumulative value of rainfall and its corresponding maximum pore pressure values for each piezometer deployed in each of the DEPs at different locations during the period of October-November 2012. As mentioned earlier, the highest rainfall rate in this period is 24mm and is on 9th October. The impact of this rain on each piezometer is different. A sudden rise can be seen in *piezo1* and *piezo3* of DEP group5 while *piezo5* of DEP group1 and *piezo1* of DEP group4 shows a rise in its value after 15-18days due to the impact of the same rain. Similarly, *piezo1* and *piezo2* of DEP group2 and *piezo2* of DEP1 show a small rise in 2-3 days whereas *piezo5* and *piezo3* of DEP group1 shows its peak values after 15-18days and *piezo1* of DEP group4 shows a small rise after the same duration. This shows that not just the antecedent conditions of rain alone causes a rise in pore pressure, but it depends also on soil structure & properties as well, because of which some soil layers may have the capability to retain moisture which is why they show an increased rate of pore pressure buildup. Thus by analyzing this graph, we can conclude that the pore pressure build up is also dependent on the depth at which the sensor is placed, the soil structure, duration of the rain and the antecedent condition of rain.

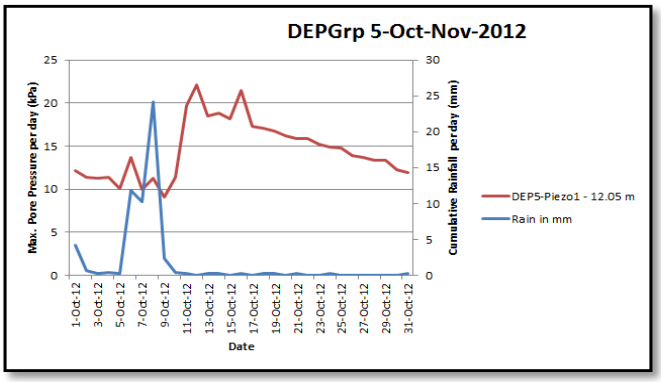


Fig. 6. Rainfall rate and Pore Pressure values of DEP5 in Oct-Nov-2012

7. Conclusion

Determining the relationship between pore pressure and rainfall rate is a challenging and complex process as it is not linear in nature. The proposed system for automatic data analysis of landslide detection system helps to establish the correlations between rainfall and the resulting pore pressure build up in a reasonably accurate manner by taking into account the different spatio-temporal parameters which impacts the occurrence of landslides.

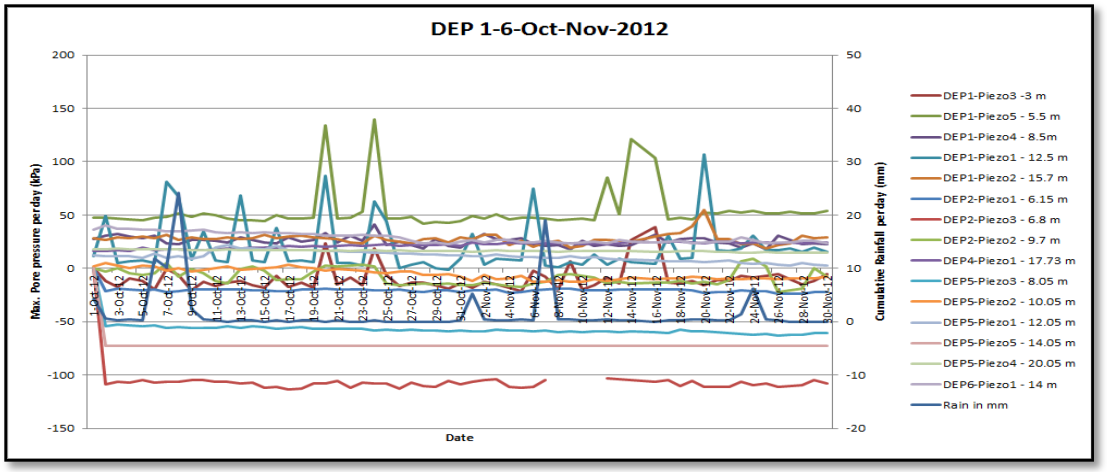


Fig. 7. Rainfall rate and Pore Pressure values of DEPs in Oct-Nov-2012

Through this work we aim to develop an early warning system for landslides by implementing several data analysis and learning algorithms which involves thorough analysis and training of the dataset to detect slope instability based on the pore pressure buildup and presence of vulnerable zones, as well as forecast the risk levels of landslide occurrence. As a future work, we are planning to include the readings from other sensors also like moisture sensor, tilt meter, strain gauge, etc to find the interdependence between each of these parameters thereby developing a more reliable and efficient system which is capable of fore warning the occurrence of landslides.

Acknowledgment

The authors would like to express gratitude for the immense amount of motivation and guidance provided by Sri. Mata Amritanandamayi Devi, the Chancellor, Amrita University. We thank Dr. M R Kaimal, Dr. Sriram Devanathan, Prof. Balaji Hariharan, Mr. Rayudu Y. V, and Ms. Geetha M, Amrita University for their valuable suggestions.

8. References

[1] KyriazisPitilakis, "Vulnerability (physical) assessment of building and infrastructures".

[2] HimadriChakrabarty, C. A. Murthy, and Ashish Das Gupta, *"Application of Pattern Recognition Techniques to Predict Severe Thunderstorms"*, International Journal of Computer Theory and Engineering, Vol. 5, No. 6, December 2013.

[3] P. Samuia, D.P. Kothari, "Utilization of a least square support vector machine (LSSVM) for slope stability analysis", ScientiaIranica, Transactions A: Civil Engineering 18, 5358, 2011.

[4] M. Van Den Eeckhaut , N. Kerle , J. Poesen , J. Hervs"Identification of vegetated landslides using only a lidar-based Terrain model and derivatives in an object-orientedEnvironment",Proceedings of the 4th GEOBIA, - Rio de Janeiro - Brazil p.211, May 7-9, 2012.

[5] Mohammad Onagh, V.K. Kumra and Praveen Kumar Rai, *Application of multiple linear regression model in landslide susceptibility zonation mapping*, International Journal of Geology, Earth and Environmental Sciences, Vol. 2 (2), pp.87-101 May -August 2012.

[6] Tony L. T. Zhan1 and Charles W. W. Ng, "Analytical Analysis of Rainfall Infiltration Mechanism in Unsaturated Soils", International Journal Of Geomechanics © Asce , December 2004.

[7] http://www.tulane.edu/~sanelson/Natural_Disasters/slopestability.html

[8] Ramesh. M.V, Ushakumari. P, "Threshold Based Data Aggregation Algorithm To Detect Rainfall Induced Landslides", in Proceedings of the 2008 International Conference on Wireless Networks (ICWN'08), Vol. 1, Pages 255-261, CSREA Press, July, 2008.

[9] Maneesha V. Ramesh, NirmalaVasudevan "The deployment of deep-earth sensor probes for landslide detection" Landslides, Pages 457–474, December 2011.

[10] LAN Hengxing, ZHOU Chenghu, C. F. Lee, WANG Sijing& WU Faquan "Rainfall-induced landslide stability analysis in response to transient pore pressure" Science in China Ser. E Technological Sciences Vol.46 Supp. 52_68, 2003.

[11] Richard M. Iverson,"Landslide triggering by rain infiltration",Water ResourcesResearch, Vol. 36, No. 7, Pages 1897-1910, July 2000.

[12] Bora Gundogdu , "Relations Between Pore Water Pressure, Stability And Movements In Reactivated Landslides".

[13] ManeeshaVinodini Ramesh, "Design, development, and deployment of a wireless sensor networkfor detection of landslides", Elsevier 2012.

[14] RobinChowdhury, Phil Flentje "Uncertainties in rainfall-induced landslide hazard", Quarterly Journal of Engineering Geology and Hydrogeology, 61–70, October 2001.

[15] F. Mancini, C. Ceppi, and G. Ritrovato, "GIS and statistical analysis for landslide susceptibility mappingin the Daunia area, Italy"Nat. Hazards Earth Syst. Sci., 10, 1851–1864, 2010.

[16] BiswajeetPradhan, ShattriMansor, Saro Lee, Manfred F. Buchroithner, "Application of a Data Mining Model for Landslide Hazard Mapping" The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. XXXVII, Part B8, Beijing 2008.

SESSION

CLASSIFICATION, CLUSTERING, ASSOCIATION

Chair(s)

Drs. Robert Stahlbock
Gary M. Weiss

Feature Selection by Tree Search of Correlation-Adjusted Class Distances

Hassan Rabeti
Research and Development
vAuto Inc., Austin, Texas

Martin Burtscher
Department of Computer Science
Texas State University

***Abstract*—The rapidly growing dimensionality of datasets has made feature selection indispensable. We introduce the TS-CACD feature-selection algorithm, which uses a generalization of the Stern-Brocot tree to traverse the search space. This family of trees supports different divergence ratios, *i.e.*, enables the search to focus on and reach certain areas of interest more quickly. TS-CACD uses a continuous filter method, which combines an inter/intra-class distance measure with a pair-wise ranked feature correlation measure. It requires almost no parameters, explicitly selects the most important features, and performs well.**

I. Introduction

Datasets are rapidly increasing in size and complexity. This growth, especially in dimensionality, makes it progressively harder to discover important information and relationships. Hence, feature-selection algorithms that reduce this complexity are essential for the successful mining of non-trivial data.

In multidimensional datasets, each item has several attributes (called features). For example, a dataset of cars might include the make, model, color, *etc.* of a large number of cars, and we may want to classify which of these cars are likely to develop engine problems. Feature selection aims to minimize the number of features that need to be considered while minimally degrading the classification accuracy or even improving it. Thus, feature selection can be described as determining a combination of features (*i.e.*, a subset) that optimizes an evaluation function. This evaluation function takes into account the number of selected features as well as the classification accuracy, that is, the ability to predict the class of a given item from the dataset. In the car example, feature selection should eliminate all attributes that do not correlate with engine breakdowns, such as the cars' color.

Feature selection is widely used, including in hand-writing analysis [1], social media [2], diagnostic medicine and gene selection in micro-array data [3]. Each domain has its own set of preferences for the feature-selection algorithm. Some prefer to avoid manually optimizing parameters to fine tune the accuracy and instead want an algorithm that explicitly chooses a subset. Others favor algorithms with a complexity that is linear in the number of items in the dataset. The TS-CACD algorithm does not require any parameters, targets medium-sized data sets, and does not run in linear time but therefore considers the interaction between features. A number of excellent publications exist that summarize and compare many different feature-selection algorithms [4], [5], [6].

To successfully perform feature selection, two components are needed: (1) a method to determine subsets and (2) a measure to assess the quality of a subset. Our algorithm uses a tree-search (TS) approach to find subsets and the correlation-adjusted class distance (CACD) to evaluate their quality.

At its core, our feature selection process attempts to determine a scalar coefficient (*i.e.*, a weight) between zero and one for each feature to minimize the dimensionality while maximizing the classification accuracy. To achieve this goal, the weights are chosen to deemphasize redundancy among features by considering the correlation between them. Furthermore, the weights are selected such that items of the same class are placed close to each other whereas items from different classes are placed far apart, which promotes class locality and thus improves the accuracy of many classification algorithms.

For instance, in the car example, the mileage and the year typically correlate and are therefore indicative of the same chance of engine failure. Hence, the weights are selected to ensure that the generated feature list includes only one or the other, thus lowering the dimensionality while maintaining the same predictive power. In contrast, the license plate number generally is not indicative of engine failure and is hence not useful in the classification process or may even degrade it. Thus, this feature will be deemphasized (its weight will be small) whereas other features that do discriminate engine failures will be assigned correspondingly greater weights. To generate the final subset, our approach eliminates all features whose weights are insignificant, *i.e.*, close to zero.

To efficiently explore the search space for determining good sets of scalar coefficients, we introduce a novel tree-search approach that is a generalization of the Stern-Brocot tree. This tree represents a way to construct all rational numbers by starting with two fractions $(\frac{0}{1}, \frac{1}{0})$ and iteratively inserting the mediant between each two adjacent fractions. Thus, every iteration yields a refinement of the previous set of fractions. This construction generates a binary tree that contains every rational fraction exactly once and in reduced form. Figure 1 illustrates the first few levels of the Stern-Brocot tree [7].

By following the edges in this binary tree from the root, we can reach every fraction, *i.e.*, every possible weight assignment for two features. Similarly, in our k-dimensional generalization, we follow edges to explore the search space of the weights for k features. Note that descending down the tree corresponds to a refinement of the search space. In other

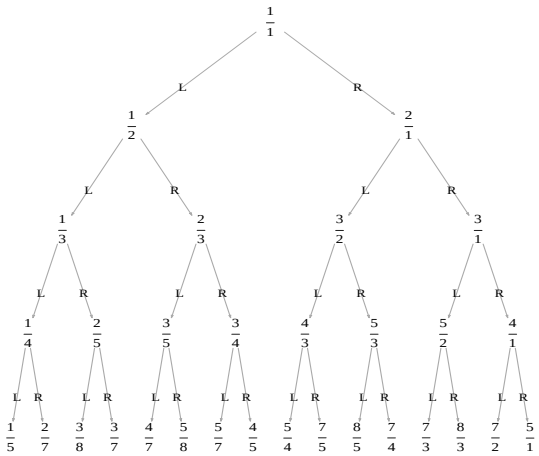


Fig. 1. Top five levels of the Stern-Brocot tree

words, every step narrows the range of possible weights.

The key operation of taking a left or right step in the Stern-Brocot tree can be expressed using the following matrices:

$$L = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \text{ and } R = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Since multiplying by one of these two matrices corresponds to following an edge, a path in the tree is tantamount to the product of a sequence of these matrices. For example, starting at the root, going left, then left again, and then right, we end up at $\frac{2}{5}$. Using the matrix notation, we multiply L by L and then by R. The resulting fraction is the sum of the elements in the bottom row over the sum of the elements in the top row.

$$L \times L \times R = \begin{pmatrix} 3 & 2 \\ 1 & 1 \end{pmatrix} \longrightarrow \frac{1+1}{3+2} = \frac{2}{5}$$

The rest of this paper is organized as follows. Section II compares our approach to related work. Section III describes our correlation-adjusted class distance in detail. Section IV explains our tree-based search strategy. Section V discusses how we use the tree search in the context of our evaluation function. Section VI presents the results. Section VII concludes.

II. Related Work

A well-known filter method is Relief [8]. Relief and its variants rank features based on the number of near hits and misses in the original space. This is done by repeatedly selecting a random sample and measuring its distance to the nearest sample of the same class (hit) and the nearest sample of a different class (miss). The final result is a ranking of how relevant each feature is in predicting the class. The user then has to select the proper subset by deciding which features to keep or discard based on the ranked relevance.

The core principle, and a similarity to our approach, is that Relief assigns meaning to the distances between neighboring samples based on the distance of their classes. However, unlike our method, Relief measures these distances in the original space. This can be a problem because removing a feature potentially changes the distances, which might increase the relevance of another feature and rank it higher in the new space. Unfortunately, it is computationally infeasible to check every combination of features to find the optimal subset. Later we discuss how we address this issue using the tree search.

The feature ranks that Relief and similar algorithms provide are a guide to help the user select an appropriate subset. Whereas this allows the consideration of multiple subsets, it does not explicitly select features. Moreover, these choices are constrained to be 'near' each other in terms of possible subsets. In particular, by changing the cut-off point, the user can include more or fewer features, but he or she cannot try new combinations such as removing a higher ranked feature and adding two lower ranked features instead that, together, supersede the higher ranked feature. In contrast, our tree traversal produces explicit subsets from the entire range of possible subsets. This is an important yet often overlooked concept. A consequence of this restriction of Relief is that it has a lower asymptotic time complexity than our method.

Another filter approach that is related to ours is EUBAFES (EUclidean BAsed FEature Selection) [9]. It is similar in that it also uses a class distance measure and scalar feature weights to enable a continuous search. One key difference is that our method uses a correlation measure to balance the class distance measure and to promote a reduction in dimensionality. EUBAFES relies on parameters to stabilize its class distance measure. When switching from one dataset to another without adjusting these parameters, we found a similar design to often converge to one extreme of the subset size or to be dominated by the parameters rather than the data. In contrast, our approach only has two internal parameters that are designed to counterbalance each other, meaning that they work well across a range of datasets. Hence, the users of our approach do not have to tune parameters. Another major difference is that EUBAFES employs a Euclidean metric in the distance measure. We found an inverted sum, *i.e.*, the sum of the inverse distances, to result in better subset quality.

There are also feature selection algorithms that perform regularization on the feature weights (generally on binary weights) by penalizing larger feature subsets, which, in turn, can force a reduction in dimensionality. Whereas we also employ such a strategy, we do so by placing a constraint on our feature weights that can only force a reduction in dimensionality in the presence of the correlation adjustment.

III. Correlation-Adjusted Class Distance

In this section, we present the CACD criterion function and discuss the reasons for choosing this function.

A. Criterion

With Q denoting the number of features and N denoting the number of instances in a dataset, let our domain be $\Re = \{X_1, \ldots, X_N\} \in \mathbb{R}^{N \times Q}$, where $X_i = \{x_{i,1}, \ldots, x_{i,Q}\}$ for $1 \leq i \leq N$. Furthermore, let $C = \{c_1, \ldots, c_N\}$ be the set of class labels so that each c_i is associated with instance X_i for $1 \leq i \leq N$. Based on these inputs, we want to compute

a set $W = \{w_1, \ldots, w_Q\}$ of positive feature weights (scalar coefficients) that satisfy the L_2 constraint

$$\sum_{q=1}^{Q} w_q^2 = 1, w_q \geq 0 \tag{1}$$

We then define the distance $d_{i,j}$ between two instances X_i and X_j as a function of these feature weights

$$d_{i,j}(W) = \sum_{q=1}^{Q} w_q |x_{i,q} - x_{j,q}| \tag{2}$$

To also capture non-linear correlations between features, we use the Kendall Tau-b [10] distance, which is defined as

$$\tau_{i,j} = \frac{n_c - n_d}{\sqrt{n_1 n_2}} \tag{3}$$

where n_c is the number of concordant (correlated) pairs, n_d is the number of discordant (anti-correlated) pairs, n_1 is the number of pairs not tied (neither correlated nor anti-correlated) in X_i, and n_2 is the number of pairs not tied in X_j.

The above constraint on the weights and the two equations represent the fundamental components of our criterion. Constraint (1) acts as a regularization strategy by limiting the number of features selected. If some features have large weights, others must necessarily have small weights. Since we are only performing pair-wise correlation comparisons, we may end up with an over-approximation, especially for subsets that include more features. As a counterbalance, we inflate the score of larger subsets by using the L_2 (sum of squares) instead of the L_1 norm while reducing the score based on the correlation. Eq. (2) simply applies a weighted transformation and measures the L_1 distance between two samples. Finally, Eq. (3) measures the ranked correlation between pair-wise features without dependence on the linearity of the correlation. It does this by only considering the signs (but not the magnitude) of the values, thus making it unsusceptible to outliers.

The following defines the interaction between the above measures. Given a set of weights, we define an approximation of the general correlation between our features by

$$K(W) = \frac{Q(Q-1) - \sum_{i=1}^{Q-1} \sum_{j=i+1}^{Q} w_i w_j |\tau_{i,j}|}{Q(Q-1)} \tag{4}$$

To handle class imbalances, *i.e.*, inputs where most instances are of the same class, we define the frequency of the intra-class (equal or 'eq') and inter-class (not equal or 'ne') occurrences

$$f_{eq} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} [c_i == c_j] \tag{5}$$

$$f_{ne} = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} [c_i \neq c_j] \tag{6}$$

With IS denoting an inverted sum, we define the intra-class distance $IS_{eq}(W)$ and the inter-class distance $IS_{ne}(W)$ as

$$IS_{eq}(W) = f_{ne} \times \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{[c_i == c_j]}{\mu + d_{i,j}(W)} \tag{7}$$

$$IS_{ne}(W) = f_{eq} \times \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \frac{[c_i \neq c_j]}{\mu + d_{i,j}(W)} \tag{8}$$

with constant parameter $\mu = 1$. Any $\mu > 0$ is suitable to avoid divisions by zero, but choosing too large a μ results in the distances not mattering. Naturally, we want the distances to be small when the classes match and large when they do not. The value of μ is empirically chosen as part of our design. Figure 2 illustrates that this choice does not greatly impact the score function. The total class distance is

$$IS_{total}(W) = IS_{eq}(W) + IS_{ne}(W) \tag{9}$$

Now we can define our criterion $J(W)$, which is

$$J(W) = K(W) \frac{IS_{eq}(W)}{IS_{total}(W)} \tag{10}$$

where $K(W)$ serves as a correlation adjustment.

The effect of using the sum of inverse distances is similar to the least squares method. However, in the least squares method, we discourage the non-equal terms whereas in this 'most inverses' method we encourage the equal terms.

Our method fundamentally promotes locality, *i.e.*, it does not force unrelated clusters together but instead allows for disjoint clusters, which has a number of implications. It makes no assumptions about the general nature of the sample set (it does not dictate one rule globally), it allows for the existence of multiple centers within one system, it is minimally affected by outliers, and it measures simultaneous and possibly disjoint clusters while performing feature weighing.

B. Discussion

The first significant concept to note is that we deliberately inflate the distances when more features are present. This is achieved by the interaction between the distance metric (2) and the constraint on the feature weights (1). The use of squares rather than some other exponent in the constraint is one of the internal parameters mentioned earlier. We use an appropriate power to make the constraint act much like the correlation. Without the effect of μ on the correlation measure, if we used an L_1-projection constraint ($\sum |w| = 1$) instead of constraint (1), the number of features present (the subset size) would not affect the final score function because we would have a homogeneous space (*e.g.*, the weight vectors $\{1, 2, 5\}$ and $\{2, 4, 10\}$ would be the same after normalization). However, given our L_2 projection, the weight vectors are larger when considering a subset with more features. So, depending on the similarity of the feature weights ($w_1 \approx w_2 \approx \ldots \approx w_Q$), the

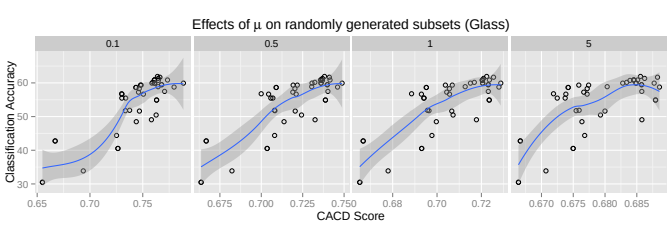


Fig. 2. Illustration of the effect of different values for μ

size of the weight vector will vary akin to how the respective points on the surface of a unit hypersphere are greater than or equal to the points on the surface of a unit hypercube. As mentioned above, this 'buffer' is needed to compensate for the nonlinear overestimation that is introduced by only considering pair-wise correlations. Considering all correlations would be computationally intractable for high-dimensional datasets and would typically only yield a small improvement in accuracy.

Since μ determines the degree of this increase for larger subsets, varying μ allows to adjust the relative influence of the correlation measure (4) on our criterion and, thus, we have a mechanism to 'encourage' pair-wise uncorrelated subsets with little extra computation. In other words, μ represents a knob to bias the criterion more towards correlation or more towards class distance. For example, in case of redundant dimensions with well-organized classes, a larger μ would devalue subsets that include redundant dimensions despite the subset's excellent ability to discriminate the classes. Note that all of this can be achieved with just one simple parameter.

Another important feature of our approach is the choice of an inverted sum to measure the quality of the class separation. It provides several key benefits. First, it (along with using Kendall's Tau-b) makes the criterion robust in the presence of outliers. Since we are dividing by the distance, outliers, which have a large distance, contribute essentially nothing to the inverted sum. In contrast, a least squares approach is easily thrown off by a single large outlier. Second, our method naturally encourages locality, *i.e.*, many small distances. Hence, it encourages class separation on a local context as the large-distance terms are insignificant. Third, it does all this without making assumptions about independence or a general inherent structure among the features. We consider these aspects fundamental advantages of our design.

The inverted sum also provides an important performance-optimization opportunity. Since only the short-distance terms matter, using a nearest-neighbor list in the search method incurs a minimal approximation error. When restricting the algorithm to only considering a fixed number of nearest neighbors, the time complexity decreases by a factor of $\mathcal{O}(n)$.

Many of the components of our criterion are embedded within our design (*e.g.*, μ) or pre-computed and therefore do not add significantly to the computation cost. However, they do add to the complexity of the score function. Since we are performing feature selection in a continuous manner, this makes it more difficult to find optimal subsets and may cause some search strategies to require extra iterations to achieve equivalent quality. This is one of the reasons why, in the following section, we introduce a search strategy that is not dependent on analytic methods or linear programming models.

IV. Tree Search

This section introduces our tree-based search and discusses its matrix and tree representation as well as some of its computational qualities. Since each feature weight can have any value between 0.0 and 1.0, it is important to establish a systematic and efficient way of subdividing and traversing the search space. We use a generalization of the Stern-Brocot tree.

A. Overview

A number of pre-existing search strategies that produce reliable results only work for criterion functions that meet certain strict requirements, such as having to be monotonic. Other search strategies tend to get stuck in local optima. Our approach does not suffer from either of these limitations.

One of the fundamental qualities of our tree search is the effect that taking a direction, *i.e.*, choosing a child to follow, has on the range of possible values for the weights. Consider the Stern-Brocot tree in Figure 1. Given an initial choice of 'left', all subsequent paths are limited to values below 1. Hence, the tree represents an effective way of partitioning the search space into independent regions with little computational overhead or dependency among workers. Another important quality is that the tree makes it possible to quantify the distance between different subsets in a meaningful manner without the need to compute the actual weights (*cf.* Section V).

Additional benefits of using a tree-based search include the following. In the presence of a dominant path, *i.e.*, a monotonic score function, only one child needs to be followed after each step and the search converges exponentially. Even when optimizing a function that is not analytic, our approach is guaranteed to terminate. First, unlike in gradient descent, the tree search cannot jump back and forth between the same two values because the step size diminishes with each level in the tree. Second, every downward step moves us closer to the (local) optimum as the range of possible weights is further confined. Hence, any desired degree of accuracy can be attained. Finally, the generated weights are guaranteed to be unique and therefore fit well with and can also be used in the parameterized coefficient search paradigm that is common in many combinatorial optimization problems that occur in fields like statistics, data mining, and linear programming.

B. Matrix representation

We consider the matrix representation very useful for reasoning about the tree-based search method. After all, deconstructing the matrices enabled us to greatly reduce the computational complexity of our approach (*cf.* Section V).

Given the real numbers $r \in (0, 1)$ (a factor that determines how quickly the tree 'spreads out') and $d \in \mathbb{N}$ (the number of features), let J_d denote a d-element vector of all ones, $S_d(i)$ a $d \times d$ zero matrix with r in every row of the i^{th} column,

$$S_d(i) = rJ_d e_i = \begin{bmatrix} 0 & \cdots & r & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & & r & & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & r & \cdots & 0 \end{bmatrix}_{d \times d}$$

and $I_d^-(i)$ the $d \times d$ identity matrix with the i^{th} 1 removed

$$I_d^-(i) = I_d - e_i^\top e_i = \begin{bmatrix} 1 & 0 & & \cdots & & 0 \\ 0 & \ddots & & \ddots & & \vdots \\ & & 1 & & & \\ \vdots & \ddots & & 0 & \ddots & \vdots \\ & & & & 1 & \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & \cdots & & \cdots & 0 & 1 \end{bmatrix}_{d \times d}$$

The transformation matrix $T_d(i)$ $1 \leq i \leq d$ is their sum.

$$T_d(i) = S_d(i) + I_d^-(i)$$

Finally, we define a sequence $P = \{p_1, p_2, \ldots, p_k \mid 1 \leq p_i \leq d \forall i\}$ as a *path*, with each p_i denoting a step or direction in the path (*i.e.*, choosing the i^{th} child of the current node in the tree), and the vectorization operation V of a path P as

$$V(P) = r\Big(T_d(p_1)T_d(p_2)\ldots T_d(p_k)J_d\Big)^\top = \vec{v} \in \mathbb{R}^d$$

To simplify the notation, we assume that d and r are given constants unless otherwise stated. For example, $r = 1$ and $d = 3$ yield the following on the path $\{2, 3\}$

$$\begin{aligned} V(\{2,3\}) &= \Big(T(2)T(3)J_d\Big)^\top \\ &= \left(\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}\right)^\top \\ &= \left(\begin{bmatrix} 1 & 1 & 2 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix} \times \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}\right)^\top \\ &= (4, 2, 3) \end{aligned}$$

This is analogous to summing across each row, multiplying by r, and placing the result in the i^{th} column. Furthermore, right-multiplication by $I^-(i)$ results in the same matrix with the i^{th} row removed as illustrated in the following

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1d} \\ a_{21} & a_{22} & \cdots & a_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ a_{d1} & \cdots & \cdots & a_{dd} \end{bmatrix}$$

$$\begin{aligned} A \times T(i) &= A \times S(i) + A \times I^-(i) \\ &= \begin{bmatrix} 0 & \cdots & r\sum a_{1i} & \cdots & 0 \\ \vdots & \ddots & r\sum a_{2i} & \ddots & \vdots \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & r\sum a_{di} & \cdots & 0 \end{bmatrix} + \begin{bmatrix} a_{11} & \cdots & 0 & \cdots & a_{1d} \\ a_{21} & \ddots & 0 & \ddots & \vdots \\ \vdots & \ddots & 0 & \ddots & \vdots \\ a_{d1} & \cdots & 0 & \cdots & a_{dd} \end{bmatrix} \\ &= \begin{bmatrix} a_{11} & \cdots & r\sum a_{1i} & \cdots & a_{1d} \\ a_{21} & \ddots & r\sum a_{2i} & \ddots & \vdots \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{d1} & \cdots & r\sum a_{di} & \cdots & a_{dd} \end{bmatrix} \end{aligned}$$

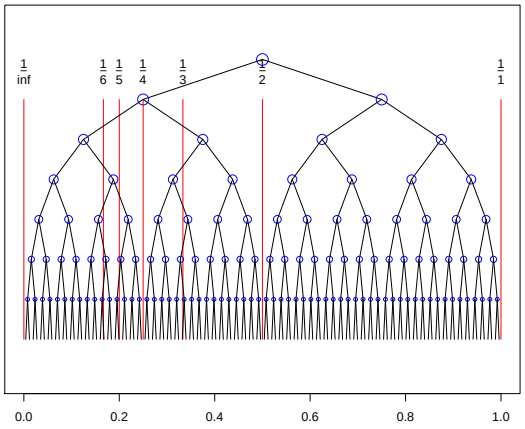


Fig. 3. $r = \frac{1}{d}$ divergence ($r = \frac{1}{2}$, $d = 2$)

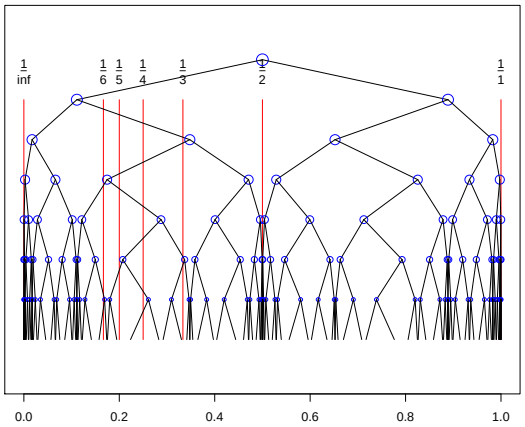


Fig. 4. $r < \frac{1}{d}$ divergence ($r = \frac{1}{7}$, $d = 2$)

This construction ensures that, for any node, there exists no other node in the tree that is a scalar multiple of it (this is equivalent to the reduced-form quality of the rational numbers in the original Stern-Brocot tree). This is important since multiples of the same set of weights would be mapped to the same final weights and thus result in redundant evaluations.

C. Tree representation

In the following illustrations, we use the projection (normalization) of $\vec{v}$ onto $\sum_i v_i = 1$ to show the effect of the divergence ratio r on the tree. Informally, the divergence determines how rapidly the tree 'spreads out'. Note that our Stern-Brocot variant is not identical to the original as we evaluate $\frac{v_1}{v_1+v_2}$ while the original Stern-Brocot approach evaluates $\frac{v_1}{v_2}$. We made this change because our projection generalizes to higher dimensions. This difference is inconsequential for demonstrating the effect of the divergence. Whereas our variant deviates from the original tree (which is no longer a subset of our generalization), we can still obtain the rest of the original tree, which spans $(1, \infty)$, by inverting all nodes $\in (0, 1)$.

Figure 3 presents a perfectly balanced version of our tree. With $r = \frac{1}{d}$, the projection space evenly partitions the region $(0, 1)$ for any d, that is, any number of features. From the point of view of a pre-projection vector, this divergence ratio

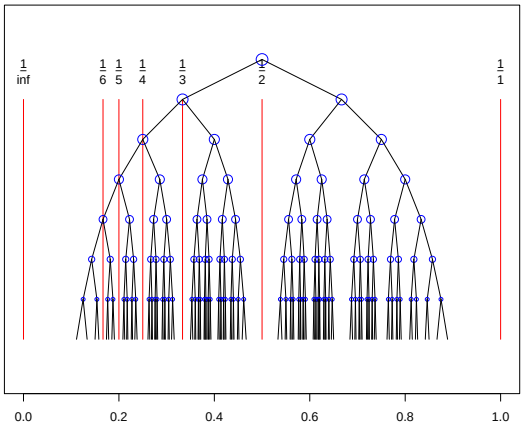


Fig. 5. Stern-Brocot divergence ($r = 1$, $d = 2$)

means that we adjust the resulting vector length after adding two vectors in each step down the tree to form a new vector of the same length. It is important to note that not all weights are reachable using a finite number of steps given $r \neq 1$, but this is not an issue in terms of our search as we can still get arbitrarily close and then terminate the search.

Figure 4 illustrates a divergence above the balance point. We found such $r < \frac{1}{d}$ ratios to be the most useful because, for dimensionality reduction, we prefer faster divergence to the boundaries. If the good solutions lie near the boundaries, the original Stern-Brocot divergence ($r = 1$) performs poorly since the rate of expansion towards the borders decreases as we go down the tree, as Figure 5 illustrates.

Figure 6 presents these divergence ratios in $d = 3$ projected to $\vec{u} = \left(\frac{v_1}{v_1+v_2+v_3}, \frac{v_2}{v_1+v_2+v_3}\right) \in \mathbb{R}^2$. It shows the limitations of using an $r = 1$ divergence in higher-dimensional domains, which only covers a small part of the search space (it takes many steps to get close to the borders) whereas $r = \frac{1}{d}$ covers the space evenly and $r < \frac{1}{d}$ emphasizes the border regions.

V. Computational Qualities

Figure 7 illustrates, for a 5D dataset, how every path from the root (at the center) of our search tree leads to a unique distribution of the five weights (each slice of a pie represents a weight). In particular, every step along a path subdivides the search space into ever smaller partitions, that is, it restricts the range of possible values for each weight. We repeat this process until there is no more improvement in $J()$ or we approach a local optimum. Given a specific path, we can define our score function as

$$J(W) = J\left(\frac{V(P)}{||V(P)||_2}\right)$$

where V is the vectorization function and P is the path.

Our search method starts from identity (where all weights are equal) and uses a divergence parameter r that is either equal to or less than the inverse of the dimensionality. In the latter case, the tree emphasizes the boundary regions where we assume the optimal results to be located, *i.e.*, where the lower-dimensional subsets reside since some of

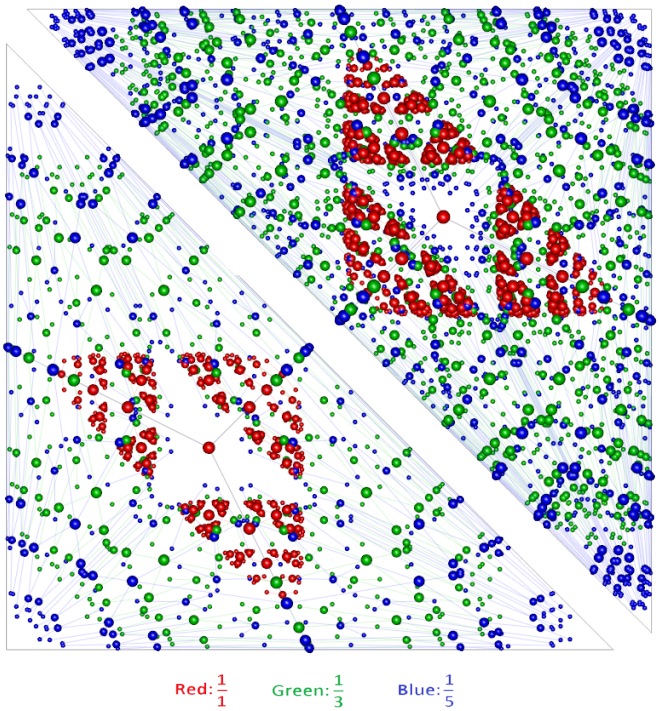


Fig. 6. Comparison of the 5^{th} layer (bottom triangle) and 6^{th} layer (top triangle) ($r_{red} = \frac{1}{1}$, $r_{green} = \frac{1}{3}$, $r_{blue} = \frac{1}{5}$, $d = 3$)

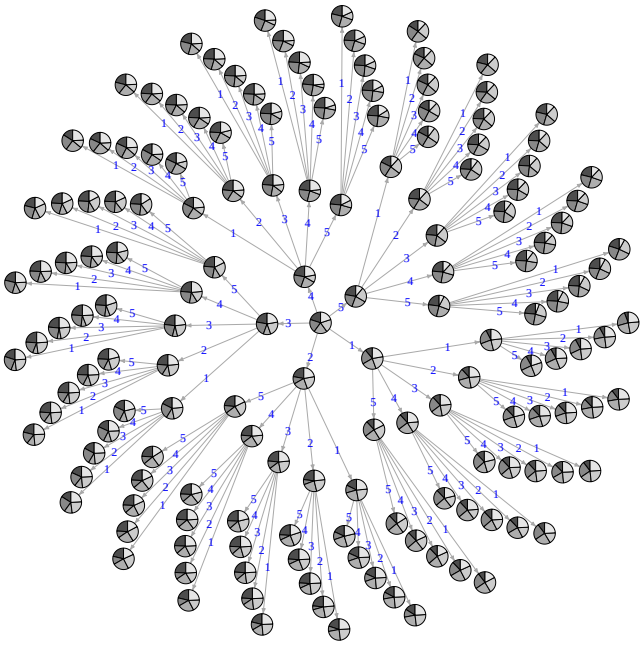


Fig. 7. First four levels of our five-dimensional search tree where the size of the pie slices represents the magnitude of each of the five weights

the weights are zero on the boundaries. This is similar to a gradient descent except the step size is given by the tree depth. Moreover, instead of moving locally (additively), we are moving globally (multiplicatively) within the region of the search space determined by the parent node. Every step down the tree further constrains this region. Note that this is all done

as part of the tree traversal without additional computations.

Algorithm 1: Vectorize(P)

Input: r is a scalar constant, k is the number of steps in the path, and d is the number of features

Input: $P = \{p_1, p_2, \ldots, p_k \mid 1 \leq p_k \leq d\,, k \in \mathbb{N}\}$

Output: $\vec{v} = \{v_1, v_2, \ldots, v_d\}$

begin

/* populate matrix */

for $i = 1 \rightarrow d$ **do**

col$[i] \longleftarrow e_i$ $\quad \{\{1, 0, \ldots, 0\}, \{0, 1, \ldots, 0\}, \ldots\}$

end

sums $\longleftarrow \{1, 1, ..., 1\}$ $\quad$ {given identity I_d}

/* apply transformations */

for $i = 1 \rightarrow k$ **do**

old $\longleftarrow$ col$[p_i]$

col$[p_i] \longleftarrow r \times$ sums

sums $\longleftarrow$ col$[p_i]$ + sums − old

end

return *sums*

end

Note that each assignment requires d operations.

In high-dimensional datasets, concurrently following multiple tree branches using the matrix representation is expensive as multiple large matrices have to be stored. Fortunately, this problem can be remedied by identifying a tree node using the path leading to it, which allows us to exploit the vectorization process of a path. Without this optimization, performing a $V(P)$ calculation for some path $P = \{p_1, p_2, \ldots, p_k\}$ requires k matrix multiplications, which take $\mathcal{O}(kd^3)$ operations. Reusing the old row sums in Algorithm 1 reduces the complexity to $\mathcal{O}(d^2 + kd)$, where the $\mathcal{O}(d^2)$ component is for generating the initial sums and the $\mathcal{O}(kd)$ component is for applying the transformations. We can further lower this complexity to $\mathcal{O}(kd)$ by recording the sum for each step, *i.e.*, by storing $d^2 + d$ values instead of only d^2 values. This small increase in storage greatly reduces the amount of computation because it eliminates the need to reinitialize the matrix; instead, we can simply continue the traversal from a tree node using the recorded sum. This benefit makes the path approach a computationally interesting alternative to the matrix approach. Moreover, both types of search are amenable to parallelization on multicore processors if desired.

Another useful aspect of the path notation is that it allows the direct measurement of the distances between subsets without having to compute any weights. As mentioned, this benefit is often overlooked. For example, given $r = 1$, $d = 3$ and the paths $a = (3, 1, 1, 1), b = (2, 1, 1, 1)$ and $c = (1, 3, 3, 3)$, path a may at first glance appear to be closer to b than to c, but this is not the case. Path c is quite close to a much like paths $(LRRR\ldots)$ and $(RLLL\ldots)$ converge towards each other in the original 2D Stern-Brocot tree in Figure 1. To formally show that path a is closer to c than to b, let us deconstruct the three vectors into binary form with respect to each possible direction. As shown below, we decompose the k elements of each path into d binary vectors with k elements where each element indicates the presence or absence of a particular direction in that position. For example, following child 3 in the first step of path a results in 0s in the first position of all d vectors except in the vector representing 3, which has a 1 in the first position. The remaining bits are determined similarly.

$$a = (3, 1, 1, 1) = \{[0, 1, 1, 1]_1, [0, 0, 0, 0]_2, [1, 0, 0, 0]_3\}$$
$$b = (2, 1, 1, 1) = \{[0, 1, 1, 1]_1, [1, 0, 0, 0]_2, [0, 0, 0, 0]_3\}$$
$$c = (1, 3, 3, 3) = \{[1, 0, 0, 0]_1, [0, 0, 0, 0]_2, [0, 1, 1, 1]_3\}$$

Given this format, we can easily compare the paths by subtracting one from the other. Since 1 and -1 are equally far away from 0, it suffices to only consider the absolute value of each term. For improved readability, we convert the binary differences into decimal vectors as a final step.

$$a - b = \{[0, 0, 0, 0]_1, [1, 0, 0, 0]_2, [1, 0, 0, 0]_3\} = \{0, 8, 8\}$$
$$a - c = \{[0, 0, 0, 1]_1, [0, 0, 0, 0]_2, [0, 0, 0, 1]_3\} = \{1, 0, 1\}$$

Since $d = 3$, the distance between two paths is a vector of three elements. Clearly, the vector $\{0, 8, 8\}$ is longer than the vector $\{1, 0, 1\}$, confirming that path a is closer to c than b.

We are explaining this computation in such detail to highlight that one can quickly compute the distance between two paths using simple binary operations, *i.e.*, without the matrices, vectorization, or explicit computation of the weights. This is useful for exploring multiple tree branches based on diversity because the distance of each candidate path can be rapidly evaluated so that sufficiently different paths can be chosen.

Finally, let us take a look at the coefficients (vectorization) generated by these paths: $V(a) = (5, 8, 4)$, $V(b) = (5, 4, 8)$ and $V(c) = (4, 8, 5)$. Again, $V(a)$ is more similar to $V(c)$ than to $V(b)$. More importantly, this technique can also be reversed to generate a new path from existing paths to promote diversity in the search space, which is another benefit of our approach.

Although the presented type of search is ideal for continuous $J()$ functions, it is also capable of producing a binary backwards or forwards search. For $r = 0$, the algorithm performs a binary backward selection with the restriction that it can only take a specific step of a path once to ensure termination. If we define a new vector $\vec{u} = 1 - \frac{\vec{v}}{\sum v_i}$ with $r = 0$ and a similar path restriction on repeating a step, we perform a binary forward selection. We are not suggesting that anyone use our method in this way. Rather, we want to point out that the popular binary backwards and forwards searches emerge as two special cases of our more general approach.

A. Traversal procedure and termination

Our search method starts at the root, with identity weights, and calculates a coefficient of variation based on all children's $J()$ score, which determines the number of branches to consider. The higher the variation is, the more children are visited. Generally, only few children need to be followed, thus avoiding and drastically outperforming an exhaustive search.

This process continues recursively, with some decay based on the depth as a soft termination criterion to limit the total computation. The coefficient of variation also serves as a hard termination criterion because, with each increase in depth, the $J()$s of the children get closer to each other. Eventually, the coefficient becomes arbitrarily small (indicating that the score is not improving) because each step considers a finer granularity than the previous step. An added benefit is that, given a change in our weights (let us call this distance δ), we can make certain assumptions about the corresponding change in our $J()$ function. This allows us to finish the search outside the tree using an iterative uphill climb with δ as its step size because we can guarantee (with arbitrarily high probability depending on the threshold) that we will not miss a local optimum, *i.e.*, we will be in the safe zone. This hybrid approach between the tree search and the following uphill climb greatly decreases the running time as the convergence of the tree search slows down with increased depth.

VI. Evaluation

A. Datasets

We use nine popular datasets from the UCI machine learning repository for our evaluation [11], [12], [13]. Aside from ensuring proper formatting, we did not modify these datasets. Table I summarizes pertinent information about each dataset.

TABLE I
Datasets

Dataset	Number of classes	Number of features	Number of instances
Madelon	2	500	2000
Breast cancer	2	10	699
Indian liver patient	2	10	583
Musk	2	166	476
Ionosphere	2	34	351
Glass identification	7	9	214
Sonar	2	60	208
Parkinsons disease	2	23	197
Lung cancer	3	56	32

B. Methodology

We compare our TS-CACD approach to the L0, mRMR, InfoGain, ReliefF, and CfsSubset feature-selection methods. The L0 subsets were generated using Matlab with the mc_svm (multi-class support vector machine) function. For mRMR, we used the mRMR website to perform the feature selection [14]. We employed Weka for the remaining three methods [15].

To evaluate each method, including ours, we averaged the classification accuracy of the NaiveBayes (a Bayesian model with a probabilistic metric), J48 (a decision tree with a region-based metric), IBk (a k-nearest neighbor algorithm with a distance metric), and SMO (a support vector machine with a kernel metric) classifiers with ten-fold cross-validation on the resulting subset recommendation using the R package RWeka [16]. We performed our testing with these four archetypal classifiers to encourage diversity, *i.e.*, so as not to bias the results toward a single classifier and its associated metric. Moreover, we made sure that none of the four algorithms consistently performed poorly for a given classifier, which would otherwise have resulted in an unfair comparison due to a poor match of classifier to feature-selection method.

The TS-CACD results were obtained on an implementation that works as described in this paper. In particular, it uses a fixed $\mu = 1$ so as not to rely on a parameter (note that, for some subsets, a slightly smaller μ would give better results). For the methods that produce a ranked list of features, we attempted to match either the score or the dimensionality to remain unbiased unless there was a strong preference for a specific subset, in which case we show results for that subset.

The evaluated TS-CACD implementation does not take advantage of some of the discussed performance optimizations (such as the nearest neighbor search) as it is primarily intended to demonstrate the quality of our proposed feature-selection method. Nevertheless, preliminary tests of a nearest-neighbor implementation on randomly generated subsets from the Sonar dataset resulted in less than a 1% change in $J()$ score, indicating that only considering the k nearest neighbors does not hurt the classification accuracy significantly while, at the same time, making the implementation much faster.

C. Results

Table II compares the classification accuracy and the number of selected features of the six methods on the nine datasets. TS-CACD strictly outperforms the other algorithms on Parkinsons and ILPD. On these two datasets, it ties with one other method each for the smallest subset but outperforms all methods in classification accuracy. On Sonar, Ionosphere, and Glass, TS-CACD also achieves the highest accuracy of all tested methods but not the smallest subset size. Nevertheless, in two cases it produces the second smallest subset. On Madelon and Musk, our method yields the smallest subset. On Madelon, no method is strictly better than any other evaluated method, *i.e.*, for each pair of methods, one has a higher accuracy whereas the other results in a lower subset size. On the remaining two datasets, Lung Cancer and BCW, our method is a close second in terms of accuracy but yields much smaller subsets than the most accurate methods. Importantly, there is no case where one of the other methods provides better accuracy and a smaller number of features. In summary, TS-CACD results in the best or close to best accuracy on seven of the nine datasets and yields the smallest subsets on the remaining two datasets. This outcome highlights the merit of our approach, that is, the importance of considering multiple dimensions simultaneously when selecting features.

Since TS-CACD's performance is a combination of the CACD metric and the tree search, we also evaluated the CACD metric in isolation, *i.e.*, independent of any search. In particular, we score random subsets using the CACD metric. The subsets were generated by first choosing a random size between one and the number of features and then randomly selecting features until the chosen subset size has been reached. Figure 8 presents the results where the lines in each panel are linear trend lines and the shaded regions represent the confidence interval for the associated plot.

TABLE II
CLASSIFICATION ACCURACY AND NUMBER OF SELECTED FEATURES BY METHOD

Dataset	w/o feat. sel.		L0		mRMR		InfoGain		ReliefF		CfsSubset		TS-CACD	
Madelon	59.462	500	63.650	3	70.275	11	70.637	16	**72.682**	20	68.175	7	62.050	**2**
Musk	81.985	166	78.571	23	72.899	20	77.836	**14**	73.897	21	**82.983**	36	77.416	18
Sonar	75.360	60	78.606	**6**	76.201	30	78.125	21	77.884	21	76.923	19	**79.206**	24
Lung Cancer	50.781	56	69.531	7	**75.120**	20	71.093	8	72.949	**4**	68.750	7	72.656	7
Ionosphere	87.250	34	86.609	13	89.316	10	88.176	18	89.102	**5**	89.814	14	**90.384**	9
Parkinsons	83.333	23	84.102	7	83.903	5	85.897	**3**	86.410	5	85.128	10	**86.538**	**3**
ILPD	65.112	10	71.012	**2**	64.193	5	65.780	3	64.837	3	66.166	5	**71.549**	**2**
BCW	95.815	9	93.919	3	94.921	5	**96.030**	8	94.563	**2**	95.815	9	95.565	3
Glass	60.864	9	60.981	5	60.981	5	62.500	7	60.514	**3**	60.864	7	**62.850**	5

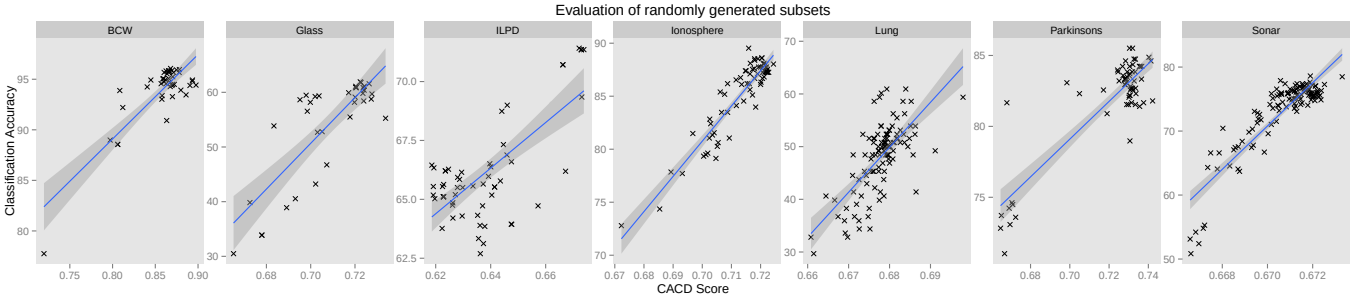


Fig. 8. Analysis of the CACD metric without any search

The results from Figure 8 are consistent with the results from Table II. Moreover, they show a strong correlation between the CACD score and the final classification accuracy, indicating that the CACD metric works well. Considering that some of these subsets are unlikely to appear in combination with a search, this demonstrates that the CACD metric accurately assesses subsets independent of our tree search. Hence, the CACD metric is likely to also be useful in combination with other (non-tree-based) search techniques.

VII. CONCLUDING REMARKS

This paper presents and evaluates our correlation-adjusted class distance criterion. In combination with our novel tree-search approach, it is very successful in finding competitive subsets in 9 datasets. The paper further presents an effective way of combining a distance and a dependency measure.

In future work, we plan to present some of the more mathematically oriented qualities of this tree and to generalize its notation. We would also like to find better alternatives to our coefficient of variation branching method so that it can consider multiple consecutive steps in a path in each iteration.

VIII. ACKNOWLEDGMENTS

Martin Burtscher and his research team are supported by NSF grants 1141022 and 1217231 as well as grants and gifts from NVIDIA Corporation and Texas State University.

REFERENCES

[1] T. Caesar, J. Gloger, and E. Mandler, "Preprocessing and feature extraction for a handwriting recognition system," pp. 408–411, 1993.

[2] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," pp. 183–194, 2008.

[3] Y. Saeys, I. Inza, and P. Larraaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/23/19/2507.abstract

[4] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Trans. on Knowl. and Data Eng.*, vol. 17, no. 4, pp. 491–502, Apr. 2005. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2005.66

[5] P. Pudil, J. Novovicova, and J. Kittler, "Floating search methods in feature selection," *Pattern Recognition Letters*, vol. 15, no. 11, pp. 1119 – 1125, 1994. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0167865594901279

[6] C. Yun and J. Yang, "Experimental comparison of feature subset selection methods," 2008. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.107.6075

[7] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete mathematics: a foundation for computer science*, 2nd ed. Addison-Wesley, 1994.

[8] K. Kira and L. A. Rendell, "A practical approach to feature selection," in *Proceedings of the ninth international workshop on Machine learning*, ser. ML92. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1992, pp. 249–256.

[9] M. Scherf and W. Brauer, "Feature selection by means of a feature weighting approach," Forschungsberichte Kunstliche Intelligenz, Institut fur Informatik, Technische Universitat Munchen, Tech. Rep., 1997.

[10] M. Kendall and J. D. Gibbons, *Rank Correlation Methods*. Edward Arnold, 1990.

[11] M. F. Akay, "Support vector machines combined with feature selection for breast cancer diagnosis," *Expert Systems with Applications*, vol. 36, no. 2, Part 2, pp. 3240 – 3247, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417408000912

[12] A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml

[13] M. A. Little, P. E. McSharry, E. J. Hunter, and L. O. Ramig, "Suitability of dysphonia measurements for telemonitoring of parkinson's disease," *IEEE Transactions on Biomedical Engineering*, 2008.

[14] H. Peng, F. Long, , and C. Ding, "Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy," vol. 27, no. 8.

[15] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H., "The weka data mining software: An update," *SIGKDD Explorations*, vol. 11, no. 1, 2009.

[16] K. Hornik, C. Buchta, and A. Zeileis, "Open-source machine learning: R meets Weka," *Computational Statistics*, vol. 24:2, pp. 225–232, 2009.

Multi-Level Synthesis of Frequent Rules from Different Databases Using A Clustering Approach

Khiat Salim, Belbachir Hafida and Rahal Sid Ahmed

Abstract—*For an effective decision-making process, knowledge workers need data, which may be geographically spread in different locations. In such circumstances, multi-database mining using local pattern analysis plays a major role in the process of extracting knowledge from different databases. During this process, a standard data mining algorithm is applied in the individual data sources in order to discover the local patterns which are synthesized in global ones. This process is called the synthesizing frequents patterns process in two levels. In fact, a few papers define a synthesis frequents patterns in multi level. Therefore, to discover the global, exceptional, majority, sub-global and local patterns is NP-complete problem. Thus, in this paper we are proposing a synthesizing model for multi-level synthesis of local patterns on the basis of the clustering approach. We also present some experiments studies to demonstrate the effectiveness of the proposed model in synthetic and real world datasets which is the petroleum field.*

Keywords *- Multi-Databases mining, Multi-level rule synthesis, Local pattern analysis, Global rules, Exceptional rules.*

I. INTRODUCTION

Nowadays, the increasing use of multi-database technology, such as advances in information and communication technologies has led to the development of many multi-database systems for real world applications. For decision-making, large organizations need to mine their multiple databases distributed through their branches. To do that, the traditional method called the mono-databases mining, can be applied, which consists of mining a huge database. However, there are various problems with this approach like performance, data privacy, data distribution loss and destroy useful patterns. To address these problems, new multi-database mining has been recently recognized as an important research area. A pattern can be a frequent itemset or an association rules for example Sucre coffee can be considered as an itemset and Sucre→coffee is an association rule that mean if we sale the Sucre we also sale the coffee with some probability. Multi-database mining (MDBM) can be defined as a process of mining data from multiple databases, which may be heterogeneous and finding novel and useful patterns of significance. While collecting all data together from different branches might produce a huge database and lose some important patterns, forwarding the local patterns rather than the original raw data to centralized company headquarter provides a feasible means of dealing with MDBM problems. Conversely the mono-database mining process, the multi-database mining process provides significant advantages like (1) Capture the data source individuality (2) Less data movement when the volume of data was important at different sites (3) data privacy (4) less expensive when huge data are distributed at various sites. The local patterns analysis approach is probably the most used in the MDBM process. To reap and forward potential patterns from individual data sources, a synthesizing model is applied. For more detail on the multi-database mining process you can consult the following works [17][24].

However in the real world, the structure of an interstate company is usually more complex where each branch can also have sub-branches and so on. For discovering the interesting patterns in such organization we propose a new process named *multi-databases mining multi-level*. It can be defined as the process of synthesizing frequent rules from different data sources at multiple levels of abstraction to form global, majority, exceptional and local rules. Majority rules [22] can grasp the distribution of rules in local ones and reflect the "commonness" of branches in their voting. High-vote rules are useful for global applications of interstate companies. Exceptional rules [19] can grasp the individuality of branches. It often present as more glamorous than high-vote rules in such areas as marketing, science discovery and information safety. Global rules can grasp the globality of rules and reflect the distribution of the rules supports. It detects the global rules instead the mono-database mining (put all databases in a huge database and apply a classic mining algorithm). In other words, it reflects the global rules which are tailed with the mono-database mining. To discover such patterns is NP-Complete problem. In the literature, we found only one paper [21] that reduces the multi-level synthesis algorithm complexity. The authors proposed a reduction of the search space based on two parameters: the Y effective and Y nominal. They proposed a synthesizing model for multi-level synthesis of local patterns on the basis of two, rule-selection measures -namely effective and nominal vote rates. Using these rule selection measures,

S.Khiat is with the National Polytechnic School Oran (ENPO) and University of Science and Technology-mohamed Boudiaf (USTO-MB), Computer Science and Mathematics Faculty, ORAN, CO 31000 ALGERIA (corresponding author to provide phone: +213-560-102-192; fax: 213 41 29 00 95; e-mail: salim.khiat@univ-usto.dz).

H.Belbachir, was with the University of Science and Technology-mohamed Boudiaf (USTO-MB) ,Computer Science and Mathematics Faculty, ORAN, CO 31000 ALGERIA. (e-mail: h_belbach@yahoo.fr).

S.Rahal, was with the University of Science and Technology-mohamed Boudiaf (USTO-MB) ,Computer Science and Mathematics Faculty, ORAN, CO 31000 ALGERIA. (e-mail: rahalsa2001@yahoo.fr).

synthesized patterns are classified into groupings of global, sub-global and local rules. With this, not only high-frequent rules but also frequent rules are taken care of and synthesized into appropriate set of sub-global rules. But this algorithm extracts only the global, sub-global and local patterns, and need a lot of input parameters like minimal support and confidence, the Y effective, Y nominal and the complexity still high. In this paper, in order to reduce the complexity of the multi-level synthesis algorithm, we propose to use the clustering method for reducing the complexity in time computing.

The remaining of the paper is organized as follows: Section II introduces the problem statement. Related work is discussed in section III. Section IV gives the detail of the proposed approach for multi-Level rule synthesis. Experimental results are described in the section V and in last section we terminate with conclusion and future work.

II. PROBLEM STATEMENT

Let the following diagram shows in Fig 1 a multi-level hierarchy of six sites:

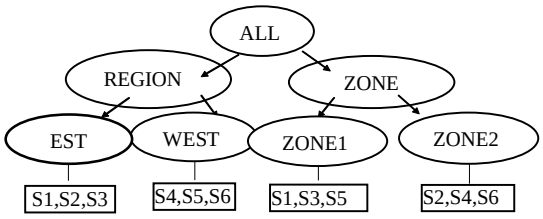


Fig 1: Multi-level Hierarchy

The interstate business organization has a large transaction in each of its six operating sites at different region or zone. The process of multi-databases mining using local patterns analysis consists of mining the individual databases at each branch and the discovered patterns are forwarded to the East or West sub-branch which are forwarded to the centralized company headquarter. As same the local patterns are forwarded to the Zone1 and Zone2 sub-branch which are forwarded to the centralized company headquarter. The challenge is to find the global, majority and exceptional patterns in each level.

More formal, lets N Site S_1, S_2 ... S_n belong to a given company and each site has M items. For example, table I describes several possibilities for discovering global and sub-global patterns:

For *i =1* ***To*** *M* ***do***

For *j =1* ***To*** *N* ***do*** $\mathbf{Sup_g(m_i)=\sum (w_j * sup_j(m_i))/j}$

$\boldsymbol{Sup_g(m_i)}$ *: The global pattern support of* m_i

$\boldsymbol{W_j}$ **:** *The weight of the site j based on transaction population*

$\boldsymbol{sup_j(m_i)}$ *: The support of pattern* $\boldsymbol{m_i}$ *in the site* ***j***

So for M patterns and N sites we have O ($M(2^N - (N+1))$) operations. Thus, the complexity of the classical algorithm is in the order of $O(M2^N)$, which is an exponential complexity. The problem may correspond to the classical problem of partitioning according to [15] which is NP-complete problem. In this situation, we formulate the problem as a clustering problem in order to reduce the complexity of multi-level synthesis algorithm [21] in time processing. Let n branches and the set of frequent itemsets FIS (Di,minsupp) corresponding to database Di and a given value of minimum support *minsupp*, for i=1,2,…,n. Thus, our problem could be stated as follows:

Find the best non-trivial partition (if it exists) of {D1, D2, …,Dn} using FIS(Di,minsupp), for i=1,2,…,n. In other hand, find the best partition that contains a lot of global, majority and exceptional pattern.

We can after date extract global, majority and exceptional patterns for each cluster. Fig.2 shows an example of two clusters formed after applying the clustering algorithm. Cluster 1 contains four sites S_1,S_2,S_3,S_4, cluster 2 contains two sites S_5,S_6.

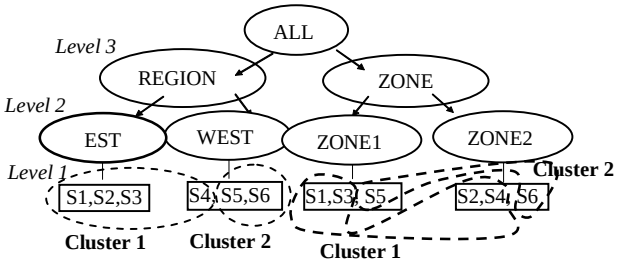


Fig 2: The two clusters for each decomposition

Suppose that after applying a standard algorithm like Apriori algorithm [18] we discover three distinct local patterns (A,B,C) two (A,B) of them in the sites S1 and S2 and the last (C) in the site S6. Those patterns can constitute an interesting knowledge for the manager in these sites (level 1).

After computing the global (or exceptional or majority) patterns for the first two patterns we calculate the global support only in all the sites and in the cluster 1 and for the third pattern we compute the global support in all the sites and in the cluster 2. So we have only four operations rather than $6x2^3$=48 operations. We suppose that the patterns (A,B) are global in cluster 1 and the pattern C are global in the cluster 2.

-In the Region decomposition we can say that (A,B) can help the manager in level 2 (EAST) and in level 3 (Region) for the top manager.

- Also in the Zone decomposition the patterns (A,B) can help the manager at level 2 (ZONE) for the top manager.

We proceed similarly for the exceptional and majority patterns in order to discover the patterns in each level.

In the next section, we will give an overview of the clustering approach on the basis of local patterns and choose the appropriate method.

III. RELATED WORK

Clustering is the process of grouping similar objects into different groups, or more precisely, the partitioning of a data set into subsets, so that the data in each subset according to some defined distance measure

Clustering has taken place in wide range of applications starting from text and media categorization, intrusion detection to health care [16][9][12][13]. In recent years, a

number of clustering approaches have been proposed with varied application perspectives, each of them aiming to improve the overall clustering. So in the literature, we found only one work [1] that clusterize the database on the basis of the frequent itemsets. These authors provided a clustering algorithm on a set of databases based on local frequent itemsets (FIS). The goal was to find the best non-trivial partition using FIS (Di, *minsupp*) through several steps. For this, the authors have defined two similarity measures, Sim1 and Sim2: Sim1 measure the similarity of the database according their FIS. Two databases are similar if they have a lot of similar itemsets without using the support. By cons, Sim2 measure the similarity of the database according there FIS and the weight of databases. Two databases are similar if they have a lot of similar itemsets with using the support. We can see that in the definition of the similarity matrix between databases DSM (Database Similarity Matrix), a problem may arise when the number of databases is important, the size of the array will be large and the computational time of similarities will be very high.

Finally, we can observe that this approach can't identify the patterns in multi-level abstraction and clustering is based on the transactions databases rather the frequent local patterns. Also the agglomerative algorithm used consume a high computation time. By cons, the spectral method can yield good results with a reduced set data like the patterns discovered at each databases, using an appropriate measure of similarity. It is based on a similarity matrix between object and might even improve the quality of clustering in the best database partition algorithm, instead of using the agglomerative method. In the next section we define our novel synthesizing local patterns process based on the clustering method especially the spectral method.

IV. PROPOSAL MULTI-LEVEL RULES SYNTHESIZING METHOD

In this section, we will propose our approach in order to synthesize the global, exceptional and majority patterns from local patterns in the multi-level process. The interstate company often consists of multi-level branches each of them has a database. Multi-databases mining process can be achieved by two phases: Intra-site and Inter-site processing.

The intra-site process: It consists of applying an algorithm for extracting local patterns in all databases. In our study, we use APRIORI [18] and FP-GROWTH [7] algorithm to extract at each database the set of local frequent itemsets in sparse and dense databases respectively.

The Inter-site process: In this step we apply a spectral algorithm on the frequent itemsets in order to identify the databases clusters that are similar for every kind of patterns.

The fig.3 illustrates this process. We can decompose the inter-site process on three steps:

a. Define de similarity function and the matrix similarity corresponding to the two measures Sim_G, Sim_{ME}.
b. Apply the spectral algorithm.
c. Identify global, majority and exceptional patterns in each cluster.

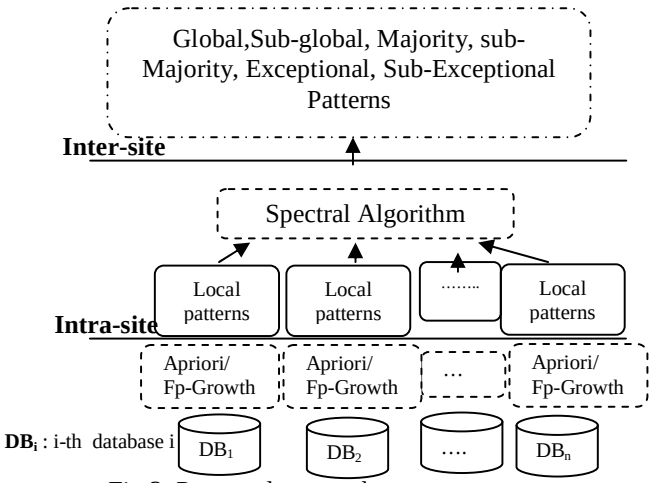


Fig 3: Proposed approach

A. *Define the similarity function*

Definition1. The measure of similarity sim_{ME} between databases D_i and D_j is defined as follows.

$$sim_{ME}(Di, Dj) = \frac{|LFIS(Di) \cap LFIS(Dj)|}{|LFIS(Di) \cup LFIS(Dj)|}$$

Where, the symbol $\cap$ and $\cup$ denote the intersection and the union operations of the set theory respectively.

The similarity measure Sim_{ME} is the ratio of the number of local frequent itemsets (LFIS) common to D_i and D_j, and the total number of distinct frequent itemsets in D_i and D_j. The partition generated is a set of cluster database that contains majority or exceptional itemsets. Two databases D_i and D_j are similar according the sim_{ME} if they contain an important number of majority or exceptional itemsets between two databases.

Definition2. The measure of similarity sim_G between databases D_i and D_j is defined as follows:

$$sim_G(Di, Dj) = \frac{\sum_{m \in \{LFIS(Di,\alpha i) \cap LFIS(Dj,\alpha j)\}} supp_G(m,\ DiDj)}{\sum_{m \in \{LFIS(Di,\alpha i) \cup LFIS(Dj,\alpha j)\}} maximum\{supp(m, Di), supp(m, Dj)\}}$$

Where, the symbol $\cap$ and $\cup$ denote for the intersection and the union operations of the set theory respectively.

m : Local frequent itemset,

$supp(m, Di)$: The support of itemset m in Di.

$supp_G(m, DiDj) = W_j \times supp_j(m) + W_i \times supp_j(m)$

$W_i and W_j$: Are the normalized weights of databases i and j based on population transaction.

$$W_j = \frac{w_j^|}{\sum_{j=1}^{n} w_j^|}$$

With $W_j^|$ is the number of transaction in database j.

n: the number of databases.

Two databases D_i and D_j are similar according the sim_G if they contain an important number of global itemsets between the two databases.

Theorem1. The similarity measure sim_k satisfies the following properties, (k=ME,G)

$$(i)\ 0 \leq sim_k(Di, Dj) \leq 1,$$

$$(ii)\ sim_k(Di, Dj) = sim_k(Dj, Di)$$
$$(iii) sim_k(Di, Di) = 1, for\ i, j = 1,2 \ldots, n$$

Proof. The properties follow from the definition of sim_k, (k= ME,G).

We express the distance between two databases in term of their similarity.

Definition3. The distance measure $dist_k$, between two databases D_1 and D_2 based on the similarity measure sim_k is defined by $dist_k(D_1,D_2)= 1 - sim_k(D_1,D_2)$, (k= ME,G)

A good distance measure satisfies the metric proprieties. Higher the distance between two databases, lower is the similarity between them.

Theorem2. $Dist_{ME}$ is a metric over [0,1].

Theorem3. $Dist_G$ is a metric over [0,1].

Proof. We show that $dist_G$ satisfies the triangular inequality.

$$dist\mathrm{G}(Di, \mathrm{Dj}) = 1 - \frac{\sum_{m \in \{LFIS(Di,\alpha i) \cap LFIS(Dj,\alpha j)\}} supp_G(m,\ DiDj)}{\sum_{m \in \{LFIS(Di,\alpha i) \cup LFIS(Dj,\alpha j)\}} maximum\ \{supp(m, Di), supp(m, Dj)\}}$$

Definition4. Let $D=\{D_1,D_2,\ldots,D_n\}$ be the set of all databases. The database similarity matrix DSM_k of D using the similarity measure sim_k, is a symmetric square matrix of size n by n, whose (i,j)-th element $DSM_{ij}k(D)= sim_k(D_i,D_j)$; for $D_i, D_j \in D$, and i,j=1,2,…,n, (k= ME,G).

For n databases, there are C_n^2 pairs of databases. For each pair of databases, we compute similarity between them. If the similarity is high then the databases may be put in the same class.

B. Apply the spectral algorithm

The main tools for spectral clustering are graph Laplacian matrices. For clustering we have based on two spectral clustering, the normalized and unnormalized methods which are based on notions of graph theory and algebra linear. Our algorithm involves the following steps:

- Calculate the laplacian matrices (Normalized/ Unnormalized)
- Calculate the Eigenvalues with using QR algorithm;
- Clusterize databases by clustering the eigenvectors with Kmeans algorithm.
- Test the quality of clustering with RationCut criteria for unnormalized method and Ncut for normalized method.

B.1. Calculate the Laplacian matrix:

In graph theory, Laplacian matrix is a matrix representing a graph. It is used in the context of graph partitioning by spectral methods. For the calculation of the Laplacian matrix from the adjacency matrix the similarity graph, we distinguish two cases:

- The unnormalized graph Laplacian matrix is defined as:

$$L = D - W$$

where D is the diagonal matrix and W is the degree of adjacency matrix of weighted graph. An overview over many of its properties can be found in [2] [3].

- There are two matrices which are called normalized graph Laplacians in the literature. Both matrices are closely related to each other and are defined as:

$$L_{sym} = I - D^{-1/2} W D^{-1/2}$$
$$L_{rw} = D^{-1} L = I - D^{-1} W$$

We denote the first matrix by L_{sym} as symmetric matrix, and the second one by L_{rw} as closely related to a random walk.

For more detail proprieties L_{sym} and L_{rw} readers can consult the first page of the standard reference [6] for normalized graph Laplacians.

B.2. Calculate the eigenvalues:

Jacobi's method and QR iteration are two of the most common algorithms for solving the problem of eigenvalues and singular value problems. According to [8] the Jacobi method is simple and gives more accurate than the QR algorithm, but it is less effective, and slower with large constant factor than the QR algorithm. However, the QR algorithm is the appropriate method for the calculation of all eigenvalues/vectors in small matrix (n<2000) [5]. It is one of the main algorithms used in the numerical linear algebra for its simplicity, speed, use of orthogonal matrix and its exceptional stability [14].

- **Principal of the QR algorithm**

This method is applied for any matrix. It consists for decomposing any matrix A on scalar of simples matrix.

$$\boldsymbol{A = QR}$$

Where:

Q: ***Orthogonal*** matrix which the columns are ***orthonormalizing*** such as: ${}^tQQ = I$ *and* $det(Q) = \pm 1$

R: ***triangular upper*** matrix which is a square matrix where the lower triangular values delimited by the principal diagonal that is null.

The algorithm is divided into two steps:

-The original matrix which is transformed in a finites number of steps into matrix *Hessenberg* form where all elements that is under the first sub-diagonal (i.e diagonal under the principal diagonal) are nulls, or the matrix with *tridiagonale* form. This step allows accelerating the convergence method.

-The iterations *QR* applied in *Hessenberg* matrix or in the *tridiagonale* matrix.

QR Algorithm

Let $A \in \mathbb{C}^{n \times n}$. This algorithm calculate the triangular top matrix T and the unitary matrix U such as: $A = UTU^*$ is the decomposition of A.

$Set\ A_0 := A\ and\ U_0 := I.$

For *k*=1, 2,… do $A_{k-1} := Q_k R_k$; /* *QR factorization**/

$A_k := R_k Q_k$;

$U_k := U_{k-1} Q_k$; /* *Update transformation matrix**/

End for

$Set\ T := A_\infty\ and\ U := U_\infty$ [25].

B.3. Clustering the eigenvector:

After calculating the *k* eigenvalues/vectors, the majority of algorithms rank the Eigenvectors as a column matrix U. Lines of this matrix must be grouped with ***k-means*** algorithm.

- **K-Means Algorithm:**

k-means algorithm is a method for clustering data into a k homogenous class. In particular, the simple k-means clustering algorithm has no difficulties to detect the clusters in this new representation. Readers not familiar with k-means can read up on this algorithm in numerous text books, for example in [20].

B.4. Graph cut point of view

The intuition of clustering is to separate points in different groups according to their similarities. For data given in form of a similarity graph, this problem can be restated as follows: we want to find a partition of the graph such that the edges between different groups have a very low weight (which means that points in different clusters are dissimilar from each other) and the edges within a group have high weight (which means that points within the same cluster are similar to each other). In this section we will see how spectral clustering can be derived as an approximation to such graph partitioning problems.

For that, We will see that relaxing Ncut leads to normalized spectral clustering, while relaxing RatioCut leads to unnormalized spectral clustering[23].

- **Approximating RatioCut for arbitrary k**

The relaxation of the RatioCut minimization problem in the case of a general value k follows a followed principle: Given a partition of V into k set$A_1, \dots, A_k$, we define k indicators vectors $h_i = (h_{1,i}, \dots, h_{n,i})'$ by :

$$h_{i,j} = \begin{cases} \frac{1}{\sqrt{|A_i|}} & if\ Vi \in A_j \\ 0 & otherwise \end{cases}$$

Where (i=1…n ; j=1…k)

Then we set the matrix $H \in \mathbb{R}^{n\times k}$ as the matrix containing those k indicator vectors as columns. Observe that the columns in H are orthogonal to each others, that is $H'H = I$. We can see that:

$$h_i' L h_i = \frac{cut\ (|A_i|, |\bar{A}_i|)}{|A_i|}$$

Moreover, we can check that:

$$h_i' L h_i = (H'LH)_{ii}$$

Plugging those formulas together we get:

$RatioCut(A_1, \dots, A_k) = \sum_{i=1}^{k} h_i' L h_i = \sum_{i=1}^{k} (H'LH)_{ii} = Tr(H'LH)$,

Where Tr denotes the trace of a matrix. So the minimization problem of $RatioCut(A_1, \dots, A_k)$ can be rewritten as follow:

$$\min_{A_1,\dots,A_k} Tr(H'LH) \text{ subject to } H'H = I$$

We can define the relax problem by allowing the entries of the matrix H to take arbitrary real values. Then the relaxed problem becomes:

$$\min_{H \in \mathbb{R}^{n\times k}} Tr(H'LH) \text{ subject to } H'H = I.$$

This is the standard form of a trace minimization problem and again some version of the *Rayleigh-Ritz* theorem [10], tells us that the solution is given by choosing H as the matrix which contain the first k eigenvectors of L as columns. We can say that the matrix H is in fact the matrix V used in the *unnormalized spectral algorithm* [23].

- **Approximating Ncut**

Technique very similar to the ones used for *RatioCut* can be used to derive normalized spectral clustering as relaxation of minimizing *Ncut*. When $k > 2$, we define the cluster indicator vector f by:

$h_i = (h_{1,i}, \dots, h_{n,i})'$ by :

$$h_{i,j} = \begin{cases} \frac{1}{\sqrt{vol(A_i)}} & IF\ i \in A_j \\ 0 & OTHERWISE \end{cases}$$

After we set the matrix $H \in \mathbb{R}^{n\times k}$ as a matrix containing those k indicators vectors as columns. Observe that $H'H = I, h_i' D h_i = 1\ and\ h_i' L h_i = cut(A_i, \bar{A}_i)/vol(A_i)$. So we can write the problem of minimization *Ncut* as :

$$\min_{A_1,\dots,A_k} Tr(H'LH) \text{ with } H'DH = I.$$

Relaxing the discreteness condition and substituting $U = D^{1/2}H$ we obtain the relaxed problem:

$$\min_{U \in \mathbb{R}^{n\times k}} Tr\left(U'D^{1/2}LD^{1/2}U\right) \text{ subject to } U'U = I.$$

This is the standard trace minimization problem which is solved by the matrix U which contains the first k eigenvectors of L_{sys} as columns. Re-substituting $H = D^{-1/2}U$, we see that the solution H consists of the first k generalized eigenvectors of $Lv = \lambda Dv$. This yields the normalized spectral clustering algorithm according to[11].

C. Identify global, majority and exceptional patterns in each cluster

In multi-level organization, generating global, exceptional and majority patterns in our approach is done from each cluster. We use the majority method cited in [22], and the exceptional method cited in [4][19]. For example, we check if a pattern is global in all the sites, if it is not we check in the clusters. If the pattern is not global in all sites and in the clusters we affirm that the pattern is local. In conclusion, the knowledge generated is specific to each level of the hierarchy and each decision maker has the knowledge according to its level of abstraction.

V. EXPERIMENTS

To show the performance of our approach Multi-level synthesis of frequent rules, we follow three studies:

Study 1: Compare the computational time of our approach with the clustering approach based on algorithm [1].
Study 2: Compare the results of our approach with the multi-level synthesis algorithm [21].
Study 3: Interpretation our results using the production dataset of petroleum installation.

Study 1: We experimented with the same sparse dataset T10I4D100[1] and T40I10D100K[1] used in [1] and divided these datasets into ten databases, each one contains 10.000 records, Fig.4 and Fig.5 show the difference between our results and the results of the approach cited in[1]. We observe that their execution time is in the order of minutes while our approach is in the order of seconds. We can conclude that our approach is faster than the approach cited in [1].

[1] http://fimi.ua.ac.be/data/

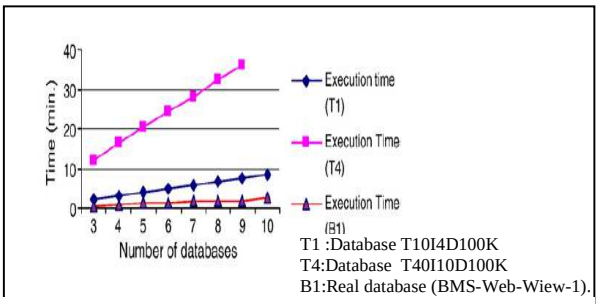


Fig 4: Execution time versus the database number [1]

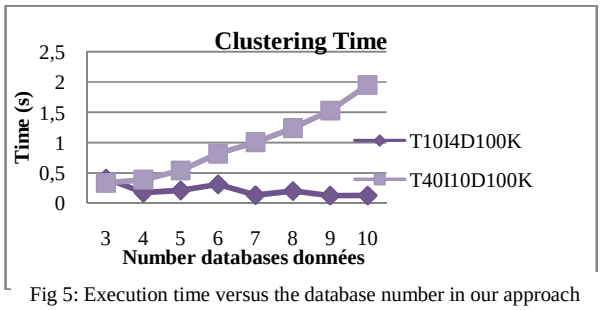


Fig 5: Execution time versus the database number in our approach

Study 2: In order to verify the effectiveness of our approach, we conducted some experiments on a dataset used in [21]. Results obtained by this one are compared with our approach. Before applying our approach, it is necessary to review the hierarchical structure used, which was presented as follows in Fig. 6. We have ten sites partitioned in North, South, East and West.

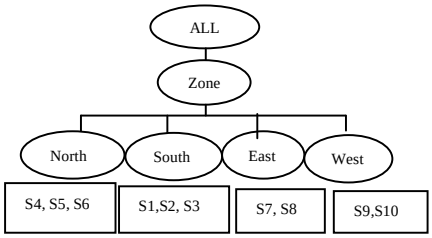


Fig 6: Hierarchy used in [21]

Patterns discovered in [21] are global, sub-global and local rules. By cons, the results obtained by our approach are presented as global, sub-global, exceptional, sub-exceptional, majority, sub-majority and local rules as shown in table II. We can say that our approach gives more knowledge than [21]. For example, rules AB → G and CF→ J in [21] are listed as sub-global rules in a standard classifier while in our approach as shown in table II they are listed as exceptional rules at south and East level respective.

Study 3:

We will try in this study to apply our algorithm on the production data set in petroleum field. It is important to exploit the huge amount of production data in order to extract useful and knowledge to each data source that can be used for decision-making in order to optimize the petroleum installations or the production process. The database production is fueled by the daily data entered and validated by the relevant services in the plant unit (daily report). We are interested to the lost produce data (LOP).

The goal of this experiment is to discover the causes of the lost production (LOP) quantity. For that we arrived to extract more interesting rules in different levels that will help decision makers to making the right decisions. Fig.7 shows the hierarchy of the organization of petroleum installation. We have four plants (level 1) distributed over GL (Gas liquefied) and GP (Gas propane) plants (level 2). For each plant has a manager which insured the good operation. And in the GL and GP plants there are managers that insured the right operation for the GL plants and GP plants. Finally in the centralized company manager there are managers that insure the good operation of the whole of the plants. Each manager needs the appropriate pattern for his level

TABLE II : RESULTS

	Rule	Global support	Site	Remarks
Global rule	A→B	0,363	S1, S2, S3, S4, S5, S6, S7, S8, S9	
	A→C	0.286	S1,S2,S3S4 ,S5,S6,S7,S 8	Eight sites
Majority rules	A→B	0.363	S1,S2,S3S4 ,S5,S6S7,S 8,S9	All sites
	A → C	0.286	S1,S2,S3S4 ,S5,S6S7, S8	Eight sites
Exceptio nal rules	AC→F	0,1875	S1,S5,S7	
	CD→E	0.3125	S7,S8,S9	At east, west and over All
	AB→D	0.375	S1, S2	At south level
	AB→G	0.3125	S1, S2	At South level
	CF → J	0.26	S9	At east level

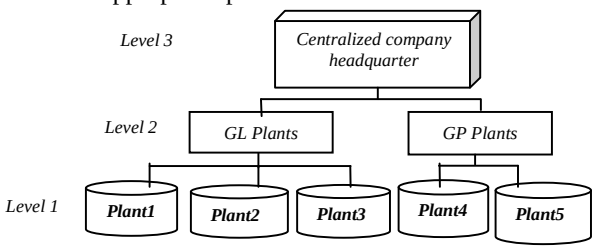


Fig 7: Structure of Petroleum Company

The five databases contain data between 10.000 and 30.000 records for 10 years. The structure of the database for each plant is: *LOP (Unit, Day_hh_mm, train_Code, LOP_problem_code, class_code, LOP_Qty).*

Where: *Unit:* The plant; *Day_hh_mm :* The date in hour and minutes; *Train_code:* The code of the train; *LOP_problem_Code:* The code of LOP problem; *Class_code:* the classes code of the problem; *LOP_Qty:* The quantity of LOP.

For building the transactional database, the domain expert must select the TID (Transactional identifier) field which can be the *Train_code or Day_hh_mm* and the field ITEMS which can be the LOP*_Problem_code* or C*lass_code*. For this study we set the TID by the *Train_code* and the ITEMS by *LOP_Problem_code*.

The goal of this experiment is to discover the causes of the quantity of lost production (LOP) which can be done by the rules discovered at each level. The interpretation of association rules generated was done by the domain expert production. Managers have found realistic and meaningful rules that describe what happened in the plant at different times. In follows we will illustrate some rules interpretation.

Rule1 [SF]→ [PR]: This rule means that expertise (SF) causes and inappropriate operator training or inexperience can generate a process cause (PR or P) in the production process in all the plants. This knowledge is important for the top manager.

Rule2 [IN] →[PR]: Instrumentation (IN or I) consists of counting and the operating mastery parameters (pressure, temperature, flow, etc. ...) which the specific instruments measures. This rule shows that the instrumentation lack can cause a process problem (PR), for example outbreak a boiler. This rule is specific to GL plants. This knowledge is important for the GL Plants managers.

VI. CONCLUSION

This paper proposed process for multi-level synthesizing global patterns using a clustering approach. It draws on the advantage of clustering method to reduce the time processing. In actually, it greatly reduces the time computing and has improved the quality and efficiency of frequent itemsets mining in each level. This efficiency is performed by the global, exceptional and majority itemsets in several level of abstraction. Experiments show that our model can outperform the clustering model in [1] in time processing. In addition, our model gives more patterns than [21] at several organization levels. Moreover, the proposal synthesizing model can be used for mining association rules on multi-level branch company effectively and can constitute a tools for helping the managers to make the right decisions.

Future work will be directed toward the use of other algorithms for the last step of spectral clustering in order to obtain better results. For that, we plan to use of the Fuzzy C-Mean which is a clustering method widely used for the classification of data. This algorithm can capture some missing patterns that the classical K-means can't do it.

VII. ACKNOWLEDGEMENT

The authors would like to thank the anonymous reviewers for their detailed comments on the first version of this paper. Their feedback has helped improve the paper vastly. We also express our thanks to students, Chetouane Farah and Boukoussa Sofiane, of the University of Science and Technology –Mohamed Boudiaf Oran (USTOMB) in Algeria for their contribution to the implementation of the algorithm.

REFERENCES

[1] Animesh Adhikari, P.R. Rao, Goa University, Goa 403 206, India, "Efficient clustering of databases induced by local patterns", ScienceDirect Elsevier Journal pp.925–943, 2008.

[2] B.Mohar. «The Laplacian spectrum of graphs». In Graph theory, combinatorics, and applications. Vol. 2 (Kalamazoo, MI, 1988) (pp. 871 – 898). New York: Wiley (1991)

[3] Mohar, B. «Some applications of Laplace eigenvalues of graphs». In G. Hahn and G. Sabidussi (Eds.), Graph Symmetry: Algebraic Methods and Applications (Vol. NATO ASI Ser. C 497, pp. 225 – 275). Kluwer (1997).

[4] Chengqi Zhang, Meiling Liu, Wenlong Nie, and Shichao Zhang, «Identifying Global Exceptional Patterns in Multi-database Mining », IEEE Computational Intelligence Bulletin Feburuary 2004 Vol.3 No.1.

[5] David S. Watkins, *"Understanding the QR algorithm, Part IX"*, Department of Mathematics, Washington State University, 2008.

[6] Chung, F. « Spectral graph theory» (Vol. 92 of the CBMS Regional Conference Series in Mathematics). Conference Board of the Mathematical Sciences, Washington.

[7] Jiawi Han, Jina Pei, Yiwen Yin, Runying Mao, « Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach » Data Mining and Knowledge Discovery, 8:53-87, 2000

[8] James Demmel and Kresimir Veselic, *"Jacobi's method is more accurate than QR"* CiteSeerX 1992

[9] Becker, H., Namman M., and Gravano, L. 2010. «Learning similarity metrics for event identification in social media» In Proceedings of the Third ACM International Conference on Web Search and Data Mining, USA, p 291-300

[10] Lütkepohl, H. "Handbook of Matrices". Chichester: Wiley(1997).

[11] Shi, J. and Malik, J. «Normalized cuts and image segmentation». IEEE Transactions on Pattern Analysis and Machine Intelligence,22(8),888–905(2000)

[12] Bharti, K., Jain, S., and Shukla, S. 2010. «Fuzzy k-means clustering viaJ48 for intrusion detection system» International Journal of Computer Science and Information Technologies, Vol. 1, p 315-318.

[13] Lui, Y., Cai, J., Yin, J., and Fu, A. 2008. «Clustering text data streams» Journal of Computer Science and Technology (Jan 2008) , p 112-128.

[14] MARK ASCH, Cours :*"Méthodes numériques pour le calcul des valeurs propres"*,2008*http://academia.edu/3002048/METHODES_NUMERIQUES_POUR_LE_CALCUL_DES_VALEURS_PROPRES*

[15] Michael R. Garey, David S. Johnson, «Computers and Intractability: A Guide to the Theory of NP-Completeness» ACM Digital library 1979.

[16] Kulkarni, P., Kulkarni, M. 2002. «Advance forecasting methods for resource management and medical decision making» In Proceedings of National Conference on Logistics Management: Future Trends.

[17] Pralhad Ramachandrarao; Animesh Adhikari, Witold Pedrycz, 2010 « Developing Multi-Database Mining applications », Advanced Information and Knowledge Processing; Springer-Verlag London.

[18] Rakesh Agrawal, Ramakrishnan Srikan « Fast Algorithms for Mining Association Rules » In VLDB'94 pp 487-499.

[19] Shichao Zhang, Chengqi Zhang, Jeffrey Xu Yu, «An efficient strategy for mining exceptions in multi-databases ». Article in press;An international journal information Science. Elsevier.2003

[20] Hastie, T., Tibshirani, R., and Friedman, J. «The elements of statistical learning». New York: Springer (2001).

[21] Thirunavukkarasu Ramkumar, Srinivasan Rengaramanujam; «Multi-Level Synthesis of Frequent Rules from Different Data-Sources», International Journal of Computer Theory and Engineering, Vol. 2, No. 2 April, 2010

[22] Thirunavukkarasu Ramkumar, Rengaramanujam Srinivasan, « Modified algorithms for synthesizing high-frequency rules from different data sources »,Springer-Verlag London Limited 2008.

[23] Ulrike von Luxburg , «A Tutorial on Spectral Clustering, Max Planck Institute for Biological Cybernetics», Technical Report No. TR-149 August 2006

[24] Xindong Wu, Shichao Zhang, Chengqi Zhang; « Multi-Database Mining » IEEE Computational Intelligence Bulletin Vol.2 No.1 2003.

[25] ETH Zurich, Cours chapter 3: « The QR Algorithm», 2012. *http://people.inf.ethz.ch/arbenz/ewp/Lnotes/chapter3.pdf*

Mammogram Classification Using Association Rule Mining

Deepa S. Deshpande[1], Archana M. Rajurkar[2], and Ramchandra M. Manthalkar[3]
[1]Department of Computer Science & Engineering, MGM's Jawaharlal Nehru Engineering College, Aurangabad, Maharashtra, India
[2]Department of Computer Science & Engineering, MGM's College of Engineering, Nanded, Maharashtra, India
[3]Department of Electronics & Telecommunication Engineering, SGGS Institute of Engineering & Technology, Nanded, Maharashtra, India

Abstract - *Breast cancer is the primary and the most common disease found among women. It is responsible for rapid growth in mortality rate among all types of cancers in women. Today, mammography the most powerful screening technique is used for early detection of cancer which increases the chance of successful treatment. Screening with mammography can show changes in the breast up to 2-3years before a physician can feel them. But it is not a perfect solution due to abnormalities that are not observable, abnormalities that are misinterpreted and technical problems in the imaging process. Therefore there is a significant need of computer aided detection system which can produce intended results and assist medical staff for accurate diagnosis at an early stage of breast cancer. Much research has been done in the field of mammogram classification during last two decades. With all this effort, there is still no widely used method for mammogram classification because this domain requires high accuracy. With this objective, an attempt is made to build mammogram classification system using association rule mining. In this paper an efficient association rule mining method is proposed to mine hidden relationships and trends in the digital mammograms for accurate classification. Experiments are carried out using MIAS Image Database. The performance of the proposed method is compared with the standard Apriori algorithm. It is found that the proposed method performs better due to reduction in multiple times scanning of database which results in less computation time. Also the mammogram classification system using this method can provide better accuracy up to 90% as compared with the other associative classification techniques. Thus this paper illustrates the use and effectiveness of association rule mining for mammogram classification.*

Keywords: Mammograms; Classification; Association Rule Mining

1 Introduction

The occurrence of breast cancer is increasing globally and disease remains a major public health problem. Statistics from breast cancer India website [34] tells that the breast cancer accounts for 25% to 31% of all cancers in women and the average age of developing the breast cancer has shifted from 50-70 years to 30-50 years. Globocan (2008) data [35] tells that number of new cases of breast cancer are 1,15,000 and the number of deaths are 53,000 in India, which shows rapid growth in death of women suffering from this disease i.e. one death for every two cases detected. The earlier the cancer detected the better treatment can be provided. Mammography is considered the most reliable radiological screening technique in early detection of breast cancer. But 10 – 30 % of malignant cases are not detected for various reasons like abnormalities that are not observable, abnormalities that are misinterpreted and technical problems in the imaging process. It has been proven that double reading of mammograms (consecutive reading by two radiologists) increased the accuracy, but at high cost. Therefore there is a significant need of computer aided diagnosis (CAD) system [44][49] to assist medical staff. Much research has been done during last few years in the field of mammogram classification[25][28][36-43][48][50][52][54][56][57], still there is no widely used method to classify mammograms due to the fact that medical domain requires high accuracy and adequate classification to handle the large amount of data made available by advancements in imaging. Different methods have been used for mammogram classification such as wavelets[26][58], fractal theory[30], fuzzy set theory [27][51][55], markov model[31] and neural networks [32].Though most of these methods seem to be powerful, developing more and more accurate classification systems remains the subject of active research in the medical domain. In today's emerging world data mining has been proved a very important tool in decision making. It can be one of the promising alternatives for achieving higher classification accuracy. This motivates us to develop classification system using association rule mining. With the retrospective study of different association rule mining techniques, we proposed an efficient method of association rule mining which generates frequent feature set by judging not only the occurrence of the individual feature but also the importance of that occurrence to derive associations among extracted features of mammograms. The merits of the proposed method are that it reduces repetitive passes over transaction database; it finds

less number of frequent features set level by level without backtracking in 50% less time as compared with standard Apriori algorithm [17] and also mine potential associations among the extracted features from the mammograms. These resulted associations are modeled for classification. The classification process involves training phase and testing phase. In training phase, training class is created based on scores of associations among extracted features and these scores are used to classify the pervious unseen mammograms in the testing phase. The classification system adopts proposed method of association rule mining to discover all rules to provide complete picture of the domain. The paper is organized as follows. Work related to association rule mining is discussed in section 2. System description is given in section 3. The traditional Apriori algorithm and proposed association rule mining algorithm are described in section 4 and section 5 respectively. Section 6 is dedicated to experimental work done and conclusion is given in section 7.

2 Background review

The problem of mining association rules over market basket data was introduced by R. Agrawal, T. Imielinski and A. Swami [19] and referred as AIS algorithm. It requires many passes over the database. Candidate item sets are generated and counted on-the-fly as the database is scanned. Another algorithm SETM [10] was presented by M. Houtsma and A. Swami for the same task. Like AIS it also generates candidate item sets on-the-fly based on transaction scans from the database. However it makes use of SQL to compute large item sets. Both AIS and SETM generate too many candidates causing it to waste too much effort. The most popular Apriori algorithm [17] is proposed by R. Agrawal and R. Srikant for mining association rules. It adopts BFS and counting occurrences strategy. The traditional Apriori algorithm generates too many candidate item sets but wastage like AIS decreases from the third pass onwards. Repetitive scanning of transactional database and generation of large candidate item sets are the major drawbacks of Apriori Algorithm. So improvement to Apriori is essential concern. Many efficient and scalable techniques have been examined by different researchers. In order to decrease scanning the database, Savasere et al.[21] introduced Apriori like PARTITION algorithm which does split–horizon on database for parallel processing of datasets and uses set intersections to determine support values. J.S. Park et al. [7] introduced DHP using hash technique for partitioning to reduce repetitive scans of database. New algorithm Eclat [12] was discovered by M. J. Zaki for fast discovery of association rules. It combines DFS with TID list intersections. DIC [20] is further variation of Apriori algorithm proposed by S. Brin et al. Prefix tree is employed for strict separation between counting and generating candidates. Interlocking support determination and candidate generation decreases the database scans. In 1998 C. C. Aggarwal and P.S. Yu [2] introduced a method, which uses adjacency among item sets and does not use vertical database format. In this method special threshold called primary threshold is defined. It avoids generating redundant association rules however it does not perform better for support thresholds lower than the primary threshold. CARMA [3] continuous association rule mining algorithm developed by C. Hidber allows user to change the support threshold and determines precise support of item sets to extract frequent item set. CARMA is faster than Apriori at low support threshold only. New way of mining association rules using trie structure to preprocess the database was presented by A. Amir et al. [1]. It is based on Rymon's set enumeration tree search. All transactions are mapped into trie structure with all support counts of items. Frequent item sets are generated by traversing trie structure using depth-first search. M.J. Zaki [11] studied the use of lattice theory. He used vertical database format and Boolean powerset lattice to introduce an algorithm that performs significantly better than Apriori. But this approach requires supplementary step for mapping the database to vertical format. Boolean powerset lattice requires much space to store labels and tid lists. F. Coenen et al.[4] considered trie structure with partial totals of the support counts of item sets. However it requiers another step to get actual count of an item set by summing up partial counts. Yew-Kwong Woon et al. [22] introduced an algorithm with structure called SOTrieIT (Support-ordered Trie Item set) for the fast discovery of frequent item set. The fast association rule mining algorithm for dynamic updated databases is proposed by Ni Tain-quan et al. [15] to overcome the difficulty of updating frequent item sets in the dynamic database. Many attempts have been made with bitmap techniques in the association rule algorithm [8][9][12-14][16] to improve the performance. Intersection and count operations of bitmap offer fast computation with efficient storage. The well known Frequent Pattern growth (FP-growth) algorithm [6] also gives good results. It adopts DFS and counting occurrences strategy and maintains Frequent Pattern-tree (FP-tree) of database. This generates frequent item set without generation of candidate item set and reduces multiple times scanning of database. The implementation of FP-tree is very much complex. Therefore FP-growth only gives better performance at low support thresholds.

In short, many researchers developed several methods to improve the performance of Apriori algorithm. All these methods considered only the occurrence of an item but not the importance of that occurrence for generation of frequent item set. This motivates us to develop variation of association rule mining to generate frequent item set with less computational complexity by judging the importance of occurrence of an individual item from the database.

3 System description

Major tasks involved in the mammogram classification system are data pre-processing, feature extraction and subset selection, data normalization, classification using association rule mining.

3.1 Data Pre-processing

The mammogram images used for experimentation purpose are taken from the mini mammography database made freely available by Mammography Image Analysis Society (MIAS)[45]. Mammogram images present in the image databases need to preprocess in order to improve the quality of images. In image processing [33][47] histogram equalization is a method of contrast adjustment using the image's histogram. Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to get better contrast. Therefore histogram equalization technique from the spatial domain in the image processing is applied to make contrast adjustment so that the abnormalities of the mammogram images will be better visible. This helps to improve the efficiency of mining task.

3.2 Feature extraction and subset selection

The feature extraction [46] is necessary in order to create the transactional database to be mined. The extracted features are then organized in a database, which is the input for the mining task of the classification system. Therefore fixed number of images (50) are used from each category of mammogram i.e. normal, benign and malignant and texture features [23][24][37] are extracted from these mammograms using Gray Level Co-occurrence Matrix (GLCM) method. Texture features are preferred as they are able to distinguish between normal and abnormal patterns. GLCM statistical method is used because it considers spatial relationship between neighbor pixels. Total 18 descriptors listed in "*Table I*" are extracted from GLCM texture measurement using Matlab. Most of these features are irrelevant and redundant. Therefore feature subset selection is required to limit the number of input features to achieve better accuracy. It reduces the feature space and also computational complexity. Basic heuristic method of forward selection is applied with dissimilarity measure to select most relevant five features. Forward selection procedure starts with reduced set which initializes to null. The best of features is derived from dissimilarity measure and it is added to the reduced set. Stepwise best of the remaining features is added to the reduced set to determine most relevant five features. The selected features are Sum of squares variance (F_1), Auto correlation (F_2), Cluster shade (F_3), Sum variance (F_4) and Cluster prominence (F_5). These features are then mapped into conventional database format in which columns are texture features representing an image and rows have values corresponding to those texture features. The transactions are of the form [Image ID, F_1; F_2; :::; F_5] where F_1:: F_5 are relevant 5 features for a given image. Thus transactional database is prepared for three basic categories of mammograms and used as a training dataset for classification.

3.3 Data normalization

Once the features have been selected, there is need to normalize these real value features into binary form so that it can be used for association rule mining. Hence min-max normalization [5] is applied on the selected features which normalize them based on their minimum and maximum values. Thus selected features are normalized to binary form and organized in a database in the form of transactions, which in turn constitute the input for deriving association rules.

3.4 Classification using association rule mining

Association rule mining technique is applied to the mammogram database. Extracted texture features of mammogram images are considered as items, and each image feature representation is a record**.** Generation of frequent feature set and discovery of association rules is carried out by the proposed method of association rule mining given in *"Fig.1"*. The association rule mining based classification composed of two phases. First phase represents the training phase of classification which discovers associations among texture features from the transactional database of all three categories of mammograms, while the second one deals with classification of new mammogram. Scores are assigned to all such association rules based on their accuracy measures [29] like support, confidence, lift, completeness, certainty factor. Association rules along with their scores are modeled for classification for three basic categories of mammograms i.e. normal, benign, and malignant. For appropriate classification of query mammogram, association rules are derived from query mammogram. Each rule is then matched with modeled classification rules for three basic categories or classes. If match found, scores are added on class by class basis and finally classification of the query mammogram is done to the class having highest cumulative sum. Since scope of this paper is limited to present a novel approach of association rule mining, details of classification are not provided here.

4 Traditional Apriori algorithm

The traditional Apriori algorithm [5] uses downward closure property which states that each subset of frequent item set must also be frequent. This algorithm was suggested by R. Aggrawal and R. Srikant in 1994 [17] and is one of the most important data mining algorithms. It uses a breadth first search approach, first finding all frequent 1-item set and then discovering frequent 2-itemset and continues by finding increasingly larger frequent item sets. It is named so because it uses prior knowledge of frequent item set property. It takes database D of t transactions and min-supp threshold represented as fraction of t as input. Apriori generates all possible frequent item set L_1, L_2 L_3…L_k as output. The algorithm proceeds iteratively. Item sets with single item are considered for generating frequent item set in the first pass. In the subsequent passes frequent item sets identified in the previous pass are extended with another item to generate

frequent item sets. The algorithm terminates after k passes if no frequent k- item set is found. The problem of mining association rules is decomposed into two steps:
1. Discover the frequent item sets i.e. sets of item sets that have transaction support above user pre-defined min-supp.
2. Use frequent item sets to generate association rules for the database.

Overall performance of mining association rules is determined by the first step because after finding frequent item set the corresponding association rules can be derived in straightforward manner. Therefore this paper focuses on finding frequent feature set using traditional and proposed algorithm of association rule mining.

5 Proposed method

Algorithm: Proposed method of association rule mining
Input: D – Transaction database, p – Set of items, min-supp – Minimum support threshold
Output: L- Large item set, R- Association rules
Begin
Step 1: k =1; C_k = Candidate item sets with length k.
Step 2: For i= 1 to p
Scan Database to calculate Supp-Yes and Supp-No of i
Add item i to C_k
End for
Step 3: For each item set in C_k
if Supp-Yes (item set) > min-supp then
Add that item set to L_k
Step 4: for k > 1
If L_{k-1} is not null then
Generate C_k from L_{k-1}
For each item set in C_k
If Supp-Yes (each item from item set) > Max (Supp-No of each items in item set) then
Add that item set to L_k
Step 5: L = $\cup L_k$
Step 6: k = k + 1
Step 7: Repeat steps 4 to 6 until no larger item set is found
End

Fig. 1. Proposed method of association rule mining

Like Apriori algorithm, initially transactional database D is represented by bit matrix in which each row corresponds to an item set for transaction and the columns correspond to the items. Presence of an item in the respective transaction is represented as 1 and absence as 0. Then candidate 1-itemset C_1 is formed by scanning database to get two support threshold values for an individual item i.e. count of presence of an item (Supp-Yes) and count of absence of an item (Supp-No). The large 1-itemsets L_1 for the given transactions is obtained from C_1. After that, candidate 2-itemsets C_2 can be derived from L_1. The proposed algorithm then finds all the large 2-itemsets L_2 for the given transactions by comparing the support value of count of presence of each item of candidate 2-itemset C_2 with the maximum of the count of absence support values of the items contained in it. This feature provides a good pruning effect. The same procedure is repeated until all large item sets have been found. The algorithm is given in *"Fig.1"*

6 Experimental work done

We tested our classifier using image database publically made available by Mammography Image Analysis Society (MIAS). Total 322 images are present in this database. Among these images, 208 belong to the normal class, 63 belong to the benign class, and 51 belong to the malignant class. We selected this dataset because it is freely available and commonly used by most of the researchers for mammogram classification. From this dataset we used 150 images (50 images per class) as training dataset and 25 images from each basic category for testing. Initially digital mammograms from training dataset are pre-processed using histogram equalization to make the abnormalities better visible. Then texture feature set listed in "*Table I*" is extracted using GLCM statistical method. These features are then mapped into conventional database format where columns are texture features representing an image and rows have values corresponding to those texture features. These features are passed through feature subset selection process. The best five features that present high dissimilarity with the other features are selected using forward selection method. The selected features are *Sum of squares variance (F_1), Auto correlation (F_2), Cluster shade (F_3), Sum variance (F_4) and Cluster prominence (F_5).*Transactional database is prepared by normalizing the selected texture features to binary form using min-max data normalization technique. Then the traditional Apriori algorithm and proposed association rule mining algorithm are applied on the training dataset with user defined minimum support threshold as 30%. Thus the dataset is mined using both the algorithms to obtain frequent feature set for all three predefined classes of mammograms. Performance evaluation for malignant class is discussed here.

TABLE I. SAMPLE DATA : GLCM FEATURES FOR NORMAL & ABNORMAL MAMMOGRAM

Feature no	GLCM Features	Normal	Abnormal
1	Autocorrelation	8.4775	12.7618
2	Contrast	0.0317	0.0384
3	Correlation	0.9957	0.9962
4	Cluster Prominence	619.6648	657.5643
5	Cluster Shade	70.0332	59.4065
6	Dissimilarity	0.0238	0.0263
7	Energy	0.4879	0.3887
8	Entropy	1.2330	1.3717
9	Homogeneity	0.9890	0.9881
10	Maximum probability	0.6845	0.5738
11	Sum of squares: Variance	8.4253	12.6947
12	Sum average	4.3703	5.5394
13	Sum variance	24.8066	37.9370
14	Sum entropy	1.2137	1.3518
15	Difference entropy	0.1097	0.1160
16	Information measure of correlation	-0.9140	-0.9152
17	Information measure of correlation2	0.9351	0.9493
18	Inverse difference moment normalized	0.9995	0.9995

Frequent feature set for malignant class using Apriori Algorithm:--

$L_1 = \{\{F_1\}, \{F_2\}, \{F_4\}, \{F_5\}$

$L_2 = \{\{F_1, F_2\}, \{F_1, F_4\}, \{F_1, F_5\}, \{F_2, F_4\}, \{F_2, F_5\}, \{F_4, F_5\}\}$

$L_3 = \{\{F_1, F_2, F_4\}, \{F_1, F_2, F_5\}, \{F_2, F_4, F_5\}, \{F_1, F_4, F_5\}\}$

$L_4 = \{\{F_1, F_2, F_4, F_5\}\}$

Frequent feature set = L_1 U L_2 U L_3 U L_4

Total number of frequent items = 15

Time Required: 0.1247Sec

Frequent feature set for malignant class using proposed method of association rule mining :--

$L_1 = \{\{F_1\}, \{F_2\}, \{F_4\}, \{F_5\}$

$L_2 = \{\{F_1, F_2\}, \{F_1, F_5\}, \{F_2, F_5\}, \{F_4, F_5\}\}$

$L_3 = \{\{F_1, F_2, F_5\}\}$

Frequent feature set = L_1 U L_2 U L_3

Total number of items in frequent feature set = 09

Time Required: 0.0550Sec

TABLE II. PERFORMANCE EVALUATION OF APRIORI VS PROPOSED ALGORITHM

Association Rule Mining	No of Frequent Items	Time Required to Generate Frequent Item Set
Apriori Algorithm	15	0.1247sec
Proposed Algorithm	09	0.0550 sec

Experimental result shows that the proposed method of association rule mining finds less frequent feature set level by level without backtracking in 50% less time as compared with Apriori algorithm. It is thus more time-efficient. Unlike Apriori algorithm it scans the transactional database only once, which results in less computation. Therefore frequent feature set obtained by this proposed method is used to derive association rules for all three predefined classes of mammograms. Sample of association rules derived for malignant class is shown in "*Table III*".

TABLE III. SAMPLE ASSOCIATION RULES FOR MALIGNANT CLASS

Rule No	Association Rules
1	$\{F_1, F_2\} \Rightarrow \{F_5\}$
2	$\{F_1, F_5\} \Rightarrow \{F_2\}$
3	$\{F_2, F_5\} \Rightarrow \{F_1\}$
4	$\{F_1\} \Rightarrow \{F_2, F_5\}$
5	$\{F_2\} \Rightarrow \{F_1, F_5\}$
6	$\{F_5\} \Rightarrow \{F_1, F_2\}$

Scores are assigned to all such association rules based on their accuracy measures like support, confidence, lift, completeness, certainty factor. These rules along with their scores are modeled for classification. For appropriate classification of query mammogram, association rules from query mammogram are derived. Each rule of query mammogram is matched with modeled classification rules for three categories. If match found, scores are added on class by class basis and classification of the query mammogram is done to the class having highest cumulative sum. Result of classification obtained by this classification system is given in "*Table IV*". The confusion matrix has been obtained from the testing part. 25 images from each basic category are used for testing. In this case out of 25 actual malignant images 4 images were classified as normal. In case of benign and normal all images are correctly classified. The confusion matrix is given in "*Table V*". Given *m* classes, $CM_{i,j}$, an entry in a confusion matrix, indicates # of tuples in class *i* that are labeled by the classifier as class *j*.

TABLE IV. RESULT OF CLASSIFICATION

Normal	100%
Malignant	84%
Benign	100%

TABLE V. CONFUSION MATRIX

Actual Class	Predicated Class		
	Benign	Malignant	Normal
Benign	25	0	0
Malignant	0	21	4
Normal	0	0	25

TABLE VI. CONTINGENCY TABLE

Category		Predicted Class	
		Normal	Abnormal
Actual Class	Normal	True Positive (TP)	False Negative (FN)
	Abnormal	False Positive (FP)	True Negative (TN)

In most of the research work for mammogram classification, accuracy measures like Sensitivity, Specificity, Precision and Recall are used for cost benefit analysis. Hence we preferred these measures for computation of accuracy for the mammogram classification system. The terms used to express accuracy measures are given in the contingency table "*Table VI*" where TP stands for True Positive i.e. images which are normal and labeled as normal by classifier. FP stands for False Positive i.e. images are abnormal but labeled as normal by classifier. FN stands for False Negative i.e. images which are normal but labeled as abnormal by classifier and TN stands for True Negatives i.e. images which are abnormal and labeled as abnormal by classifier.

Sensitivity = TP/Pos /* true positive recognition rate */
Specificity = TN/Neg /* true negative recognition rate */
Precision = TP/ (TP + FP)
Recall = TP/ (TP + FN)
Accuracy =
Sensitivity * Pos/(Pos + Neg) + Specificity * Neg/(Pos + Neg)

TABLE VII. CLASSIFICATION ACCRACY BY ALTERNATIVE MEASURES

Contingency Table Category	TP	FP	FN	TN	Sensitivity	Specificity	Accuracy(%)	Precision(%)	Recall(%)
Normal-Benign	25	00	00	25	1	1	100	100	100
Normal-Malignant	25	04	00	21	1	0.84	92	86	100
Normal-Abnormal	25	04	00	46	1	0.92	90	86	100

Sensitivity is the conditional probability of detecting cancer while there is really cancer in the image and Specificity is the conditional probability of detecting normal breast while the true state of the breast is normal. Precision and Recall measures are used to quantify the efficiency of the proposed classifier. Precision score of 100% for class C interprets that every tuple labeled as belonging to class C does indeed belong to class C. Recall score of 100% means that every tuple from class C was labeled as belonging to class C. Classifier accuracy with different accuracy measures is presented in "*Table VII*". We can notice from the formula of precision and recall that values of FP and FN tend to zero when precision and recall tend to 100%. Results show that False Positives and in particular False Negatives are almost zero with our classification which is very much desirable for mammogram classification. We compared proposed classification technique with the other classification techniques in terms of accuracy. "*Table VIII*" presents result of using BPNN, ARC-AC, ARC-BC, JAC, WAR-BC and our classification technique for MIAS dataset. It shows that the classification technique using proposed method of association rule mining performs well reaching over 90% in accuracy as compared with the other classification techniques.

TABLE VIII. COMPARISON WITH DIFFERENT CLASSIFICATION TECHNIQUES

Sr. No	Classification Technique	Accuracy	Reference
1	Back Propagation Neural Network (BPNN)	81%	[41]
2	Association Rule-based Classification with All Categories (ARC-AC)	69%	[42][43]
3	Association Rule-based Classification By Categories (ARC-BC)	80%	[42][43]
4	Joining Associative Classifier (JAC)	77%	[59]
5	Weighted Association Rule Based Classifier (WAR-BC)	89%	[53]
6	Classification technique using proposed method of association rule mining	90%	-----

7 Conclusion

Research contributes to build mammogram classification system using efficient method of association rule mining. Experimental result shows that the proposed method finds less frequent feature set level by level without backtracking in 50% less time as compared with Apriori algorithm. Also it reduces repetitive passes over the transactional database, which results in less computation. Association rules are derived from the frequent feature set obtained by proposed algorithm. The resultant association rules are then modeled for classification. This classification system performs well incurring accuracy as high as 90% for normal/abnormal mammogram classification as compared with other existing classification techniques. Thus, we investigated the use of association rule mining in the field of medical image analysis for the problem of mammogram classification. It can be used as a second reader to improve the detection performance of radiologists at early stage of breast cancer. Also it can reduce the computation cost of mammogram image analysis and can be applied to other image analysis applications.

8 References

[1] A. Amir, R. Feldman, and R. Kashi, "A new and versatile method for association generation", Information Systems, Vol. 22 ,no. 6,pp.333-347,1999

[2] C. C. Aggarwal and P.S.Yu," Online generation of association rules", Proc 14th Int'l Conf. Data Engg. pp 402-411, 1998.

[3] C. Hidber ,"Online Association Rule Mining", proc ACM SIGMOD Conf. pp 145-154,1999.

[4] F. Coenen, G. Goulbourne and P.H. Leng, "Computing Association Rules Using Partial Totals", proc. 5th European Conf. Principles and Practice of Knowledge Discovery in Databases, pp.54-66, 2001

[5] J. Han, M. Kamber (2001), *Data Mining*, Morgan Kaufmann Publishers, San Francisco, CA.SIGMOD Conf.,pp-1-12,2000

[6] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation", Proc. ACM SIGMOD International Conference on Management of Data, May 16-18, Dallas, Texas, pp 1-12

[7] J. S. Park, M.S. Chen P.S. Yu," Using Hash-based Method with Transaction Trimming for Mining Association Rules", IEEE Trans. Knowledge and Data Engg., vol 9,no. 5, pp. 813-824, Sept/Oct 1997.

[8] Lin T.Y.," Data Mining and Machine Oriented Modeling: A Granular Computing Approach", Journal of Applied intelligence, Oct 2000.

[9] Lin T.Y., Lounie E,"Finding Association Rules using Fast bit Computation: Machine-Oriented Modeling",IS-MIS-2000.

[10] M. Houtsma and A. Swami," Set oriented mining of association rules", Research report RJ 9567, IBM Almaden Research Center, San Jose, California, Oct 1993.

[11] M. J. Zaki, "Scalable algorithms for association mining", IEEE Trans. Knowledge Data Eng. ,vol. 12,no. 3,pp 372-390 May/June 2000.

[12] M. J. Zaki et al., "New Algorithms for fast discovery of Association Rules", in KDD-97

[13] M. J. Zaki ," Generating non-redundant association rules", in KDD-2002

[14] Morzy T., Zakrzewiez M.," Group Bitmap Index: A Structure for Association Rule Retrieval", Prod. of the 4th International Conf. on Knowledge Discovery and Data Mining (KDD-98).

[15] Ni Tian-quan, Wang Jain-dong, Peng Xiao-bing and Liu Yian, "A fast association rules mining algorithm for dynamic updated databases", Inform. Technol. J. 8(8); 1235-1241, 2009, ISSN 1812-5638.

[16] P. Shenoy, J.R. Haritsa, S. Sudarshan, G.Bhalotia, M. Bawa and D. Shah ,"Turbo-Charging Vertical Mining of Large Databases", Proc. ACM SIGMOD Conf.,pp.22-33,2000

[17] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules", Proc. 20th Int'l. Conf. Very Large Databases,pp.487-499,1994

[18] R. Agrawal, R. Srikant, Q, Vu, "Mining association rules with item constraints", The Third International Conference on Knowledge Discovery in Databases and Data Mining, 1997, pp.67-73

[19] R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", In Proc of the ACM SIGMOD

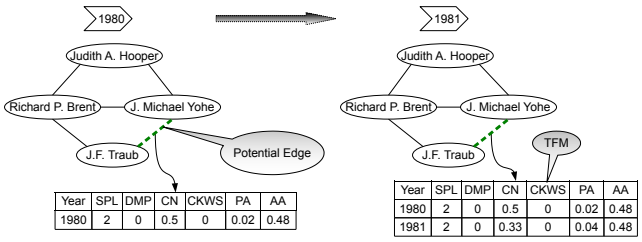
International Conference on Management of Data, Jun 1993,Washington, D.C.,USA,pp.207-216.
[20] S. Brin, R. Motwani, J.D. Ullman and S. Tsur, " Dynamic itemset counting and implication rules for market basket data", In Proc. of ACM SIGMOD Int'l conference on Management of Data ,1997
[21] Savasere A.,E. Omiecinski and S. Navathe," An efficient algorithm for mining association rules in large database", proc. of 21st international Conf. on Very Large Database,(ICLD 1995), Zurich, Swizerland,pp. 432-443.
[22] Yew-Kwong Woon, Wee-Keong Ng, and Ee-Peng Lim, "A support ordered trie for fast frequent item set discovery", IEEE Trans. On Knowledge And Data Engineering", Vol.16, No.7, July 2004
[23] A Mohd Khuzi et al." Identification of masses in digital mammogram using gray level co-occurrence matrices", Biomed Imaging Int'l J 2009; 5(3):e17
[24] Azlindawaty Mohd Khuzi, R. Besar and W. M. D. Wan Zaki, "Texture Features Selection for Masses Detection In Digital Mammogram", 4th Kuala Lumpur International Conference on Biomedical Engineering 2008 IFMBE Proceedings, 2008, Volume 21, Part 3, Part 8, 629-632, DOI: 10.1007/978-3-540-69139-6_157
[25] Brijesh Verma, Peter McLeod and Alan Klevansky, "Classification of benign and malign patterns in digital mammograms for the diagnosis of breast cancer", Expert System with Applications, 37, pp.3344-3351, 2010.
[26] C. Chen and G. Lee, "Image segmentation using multi-resolution wavelet analysis and expectation maximization (em) algorithm for digital mammography", International Journal of Imaging Systems and Technology, 8(5):491–504, 1997.
[27] D. Brazokovic and M. Neskovic. "Mammogram screening using multi resolution-based image segmentation", International Journal of Pattern Recognition and Artificial Intelligence, 7(6):1437–1460, 1993.
[28] Ferreira, CBR.; Borges, DL, "Automated mammogram classification using a multi resolution pattern recognition approach", Proceedings of XIV Brazilian Symposium on Computer Graphics and Image Processing, IEEE; Florianopolis, Brazil. 2001. p. 76-83.
[29] G. Piatetsky-Shapiro, Discovery, analysis, and presentation of strong rules, in: Knowledge Discovery in Databases, G. Piatetsky-Shapiro and W. Frawley, eds, AAAI/MIT Press, 1991, pp. 229–238.
[30] H. Li et al.," Fractal modeling and segmentation for the enhancement of micro calcifications in digital mammograms", *IEEE Trans. Medical Imaging*, 16(6):785–798, 1997.
[31] H. Li et al.," Marcov random field for tumor detection in digital mammography", IEEE Trans. Medical Imaging, 14(3):565–576, 1995.
[32] I. Christoyianni et al. "Fast detection of masses in computer-aided mammography", IEEE Signal Processing Magazine, pages 54–64, Jan 2000.
[33] "Image Processing The Fundamentals" Maria Petrou *University of SurreN Guildford, UK* Panagiota Bosdogianni *Technical Universify of Crete, Chania, Greece* JOHN WILEY & SONS, LTD
[34] http://www.breastcancerindia.net/
[35] http://globocan.iarc.fr/
[36] K.Goh,F. Chang and T. Chang, "SVM Binary classifier Ensembles for Image Classification", ACM International conference on Information and Knowledge Management ,Nov 2001
[37] Li Ke,Nannan Mu,Yan Kang "Mass computer-aided diagnosis method in mammogram based on texture features", Biomedical Engineering and Informatics (BMEI), 3rd International Conference, IEEE Explore, pp.146 - 149, November 2010, DOI: 10.1109/ BMEI.2010.5639662,
[38] Liyang Wei, Yongyi Yang and Robert M. Nishikawa," Micro calcification classification assisted by content-based image retrieval for breast cancer diagnosis", Pattern Recognition, 42, pp.1126-1132, 2009.
[39] Liu B, Hsu W and Ma Y, "Integrating classification and association rule mining", Proceedings of ACM SIGKDD International conference on Knowledge Discovery and Data Mining(KDD-98),pp. 80-86,1998
[40] Li W., Han J.,Pei J. , "CMAR : Accurate and efficient classification based on multiple class-association rules", IEEE International Conference on Data Mining,(2001)
[41] Maria-Luiza Antonie, Osmar R. Zaiane, Alexandru Coman, "Application of data mining techniques for medical image classification", Proceeding of Second International Workshop On multimedia Data Mining 9MDM/KDD'2001) in conjunction with ACM SIGKDD Conf., San Francisco, USA, Aug 26,2001;94-101
[42] Maria-Luiza Antonie, Osmar R. Zaiane, Alexandru Coman, "Associative classifiers for medical images", LNICS. Vol. 2797, MMCD, Berlin/Heidelberg: Springer; 2003 p. 68-83.
[43] Maria-Luiza Antonie, Osmar R. Zaiane , Alexandru Coman, "Mammography classification by an association rule-based classifier", Proceedings of the Third International Workshop on Multimedia Data Mining, pp. 62-69, 2002.
[44] Mehul P. Sampat, Mia K. Markey & Alan C. Bovik, "Computer-aided detection and diagnosis in mammography", Hand Book of image and video processing (second edition),2005,pages 1195-1217
[45] MIAS Database. http://peipa.essex.ac.uk/info/mias.htm
[46] Rabi Narayan Panda, Dr. Bijay Ketan Panigrahi, Dr. Manas Ranjan Patro "Feature Extraction for Classification of Micro calcifications and Mass Lesions in Mammograms" IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.5, May 2009
[47] R. C. Gonzalez. "Digital Image processing using Matlab" Pearson publication, 2005.
[48] R.Nithya and B.Santhi,"Classification of normal and abnormal patterns in digital mammograms for the diagnosis of breast cancer", International Journal of Computer Applications, vol.28, No.6, 2011.
[49] Rangayyan R.M., F.J. Ayres and J.E.L. Desautels, "A review of computer-aided diagnosis of breast cancer: Toward the detection of subtle signs", *Journal of the Franklin Institute*, Vol. 344, pp. 312-348, 2007.
[50] Ribeiro, MX.; Traina, AJM.; Balan, AGR.; Traina, C., Jr; Marques, PMA. SuGAR: "A framework to support mammogram diagnosis", IEEE CBMS 2007; Maribor, Slovenia. 2007. p. 47-52.
[51] R.Jensen,Qiang Shen ,"Semantics Preserving Dimensionality Reduction: Rough and Fuzzy-Rough Based Approches", IEEE transactions on Knowledge and Data Engineering,pp.1457-1471,2004
[52] Rojas.A and A K Nandi, "Detection of masses in mammograms via statistically based enhancement, multilevel-thresholding segmentation, and region selection", *Computerized Medical Imaging and Graphics*, Vol. 32, No.4, pp. 304-315, 2008.
[53] Sumeet Dua, Harpreet Sigh and H.W. Thompson, "Associative Classification of Mammograms using weighted rules", Expert Syst Appl.2009 July 1:36(5):9250-9259
[54] Surendiran .B and A.Vadivel, "An Automated Classification of Mammogram Masses using Statistical Measures", In *Proc. of 4th Indian International Conference on Artificial Intelligence (IICAI-09),* pp. 1473-1485, 2009.
[55] Shuyan Wang, Mingquan Zhou and Guohua Geng,"Application of Fuzzy Cluster analysis for medical image data mining", proceeding of the IEEE International Conference on Mechatronics & Automation Niagara Falls,Canada,pp.36-41,July 2005
[56] Sheshadri H. S and Kandaswamy A, "Breast Tissue Classification Using Statistical Feature Extraction of Mammograms". Medical Imaging and Information Sciences, Vol. 23, No.3, pp. 105–107, 2006.
[57] Tseng, SV.,Wang, M-H.; Su, J-H. "A new method for image classification by using multilevel association rules", Presented at ICDE 05; Tokyo. 2005. p. 1180-1187.
[58] T.Wang and N. Karayiannis. "Detection of micro calcification in digital mammograms using wavelets", *IEEE Trans. Medical Imaging*, 17(4):498–509, 1998.
[59] Yun J, Zhanhuai L, Yong W, Longbo Z. "Joining associative classifier for medical images", HIS. 2005

Feature Engineering for Supervised Link Prediction on Dynamic Social Networks

Jeyanthi Narasimhan[1], **and Lawrence Holder**[1]
[1] School of Electrical Engineering and Computer Science, Washington State University,
Pullman, WA 99164-2752, USA

Abstract— *Link prediction is an important network science problem in many domains such as social networks, chem/bio-informatics, etc. Most of these networks are dynamic in nature with patterns evolving over time. In such cases, it is necessary to incorporate time in the mining process in a seamless manner to aid in better prediction performance. We propose a two-step solution strategy to the link prediction problem in dynamic networks in this work. The first step involves a novel yet simple feature construction approach using a combination of domain and topological attributes of the graph. In the second phase, we perform unconstrained edge selection to identify potential candidates for prediction by any generic two-class learner. We design various experiments on a real world collaboration network and show the effectiveness of our approach.*

Keywords: Dynamic Graph Mining, Supervised Learning, Link Prediction, Feature Extraction, SVD



Year	SPL	DMP	CN	CKWS	PA	AA
1980	2	0	0.5	0	0.02	0.48

Year	SPL	DMP	CN	CKWS	PA	AA
1980	2	0	0.5	0	0.02	0.48
1981	2	0	0.33	0	0.04	0.48

Fig. 1: Time-Feature matrix (TFM) and a toy example with real DBLP authors. SPL: Shortest Path Length, DMP: Degree Mixing Probability (1), CN: Common Neighbors, CKWS: Common Keywords, PA: Preferential Attachment, AA: Adamic-Adar measure. Refer to Table 5 for related details. *J.Michael Yohe* and *J.F. Traub* are yet to collaborate during the years *1980* and *1981*, so *(J.Michael Yohe, J.F. Traub)* becomes a potential edge and its TFM is gathered over years till it forms.

1. Introduction

One of the graph mining tasks is Relationship Prediction or more commonly, Link Prediction (LP). It refers to predicting the likelihood of existence of a link between two entities of a network based on the existing links, node attribute information and other relevant details [1]. Instances of the LP problem can be found in application domains such as sociology, bio-chemistry, and online social networks. Communication networks can be studied under this context to disclose existing but missing communication between two people. Given the past network information of "follows" relationships from Twitter, it is possible to predict the future important person(s) in the network which is key to many business tactics like viral marketing [2]. Generally, any ecosystem, whether physical or abstract, can be mapped as networks to study the relationship formation patterns, and we are interested in finding an answer to the question: *"Is it possible to build features from the graph topological measures and also time information in such a way that any supervised learner is able to perform better on the LP task?"*

The existing approaches to the LP problem using supervised learning use a direct feature construction approach where each constituent element of a feature vector is found by measuring a global or local graph/node/edge metric. Since we are interested in dynamic networks, we offer a feature construction approach (Section 2) that is indirect in nature, yet easy to compute and includes time as an inherent component (referred hereafter as `MetaFeatures`). We also found that the existing work (Section 5) restrict their prediction to a specific set of nodes. For example, PathPredict [3] selects authors based on the number of papers and limit the training set to nodes that are directly reachable. We observe from figure 3 (Section 4.1) that majority of the edges that form are between initially unreachable nodes. Few other works ([4], [5]) restrict the nodes for potential edge prediction to those seen in training years, but since we are looking at dynamic graphs, imposing this constraint is not suitable here. We have attempted to provide a generic solution with these issues in mind. The contributions of this work include: 1) Novel meta-feature vector construction for the LP problem based on a well-established Linear Algebra technique. 2) Events like node arrival, edge formation, and deletion, with associated time information give rise to dynamic graphs, and such graphs can grow or shrink over time. We show the applicability of the meta-feature vector to dynamic graphs through experiments (Section 4). 3) We design several experiments to evaluate the above feature vector as a predictor of future link occurrences and compare with the state-of-the-art.

2. Model: Indirect discriminative feature construction

Our motivation for working with simple graph topological measures of a dynamic graph and using supervised learning for the LP problem is discussed below. Firstly, we think that there is still room for improvement in the homogeneous graph approach as invariably existing approaches use heuristics to work with only a small portion of the graph. Secondly, an ensemble of graph topological features is observed to be more effective for the problem at hand than using them separately [3]–[5]. Lastly, the importance of discriminative features in supervised learning cannot be stressed more, as any powerful learning algorithm tends to fail when not fed with good features. The necessity to combine time with the above three requirements, leads to our design of a time-feature matrix for meta-feature construction.

2.1 Features: An overview

Graph growth at the macro level is attributed to the node and edge arrivals at the micro level with continuous evolution of features. Each potential edge between any two nodes that have been in the network for some time carries with it what we call *track/historical information* (TI). This TI is maintained in a matrix format with the time in rows and static graph topological easy-to-construct measures in columns. A comprehensive list of static features suitable for the LP problem in general is given by [6]. Figure 1 shows the feature matrix for a small evolving graph between two specific authors from DBLP. We have used six features in most of the experiments - Some of the experiments use fewer features and their related details are explained in section 4.

Below is a brief overview of how each feature in this work is computed. Let $\mathcal{G}$ be the given undirected graph and $\mathcal{N}$ the set of all nodes at a given instant. Let $x, y \in \mathcal{N}$ be the authors or nodes of interest in $\mathcal{G}$. Let $\Gamma(x)$ be the set of neighbors of node x and $\mathcal{D}(x)$ its degree.

- **Common Neighbors**: $CN(x,y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\mathcal{N}|}$
- **Preferential Attachment**: $PA(x,y) = \frac{|\Gamma(x)| * |\Gamma(y)|}{|\mathcal{N}|}$
- **Adamic-Adar**: $AA(x,y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{log|\Gamma(z)|}$
- **Common Keywords**: After stopping and stemming [7]: $CKWS(x,y) = \frac{|x_{kws} \cap y_{kws}|}{|x_{kws} \cup y_{kws}|}$
- **Degree mixing probability**: Adapted from Networkx [8]. Defined as the joint probability of degrees of two nodes under consideration. For any two degrees, $\mathcal{D}_i$ and $\mathcal{D}_j$, this value is calculated as:

$$DMP(\mathcal{D}_i, \mathcal{D}_j) = \frac{\sum_{\substack{\forall x \\ \mathcal{D}(x)=\mathcal{D}_i}} \sum_{\substack{\forall y \in \Gamma(x) \\ \mathcal{D}(y)=\mathcal{D}_j}} |(\mathcal{D}(x), \mathcal{D}(y))|}{\sum_{\forall x} \sum_{\forall y \in \Gamma(x)} |(\mathcal{D}(x), \mathcal{D}(y))|} \quad (1)$$

- **Shortest Path Length**: It is the shortest distance (in hops) between any two reachable nodes in $\mathcal{G}$.

Table 1: Feature derivation from a Time-Feature Matrix (TFM). In column 2, $7 := 6\ (features) + 1\ (time)$

Graph growth-Span	TFM Shape	Constructed Vector
1 (second/minutes/. . .years)	1×7	1×6
2 (second/minutes/. . .years)	2×7	1×6
3 (second/minutes/. . .years)	3×7	1×6

Fig. 2: Typical supervised framework for solving the LP problem. Approaches differ predominantly in the way of subsetting the nodes for edge prediction. This leads to analysis on an induced subgraph.

2.2 Time-feature matrix (*TFM*): Theory

The novel meta-feature vector of our system is constructed in two steps. In the first stage, a matrix of features is calculated for each time unit and each potential edge as shown in Figure 1. Such a matrix can also be viewed as a *Multivariate Time Series* [9] holding the evolution of features of a potential edge. In the second stage, we compute the *dominant right singular vector* (meta-features) of the matrix calculated in stage one, after slicing out the time column. This way we convert multidimensional feature values to single-dimensional ones. Refer to Table 1 for a prototypical construction of feature vectors from the TF matrix. To see why we take only the dominant singular vector, let us revisit the SVD steps.

Any $t \times d$ real matrix A can be decomposed as $A = U\Sigma V^T$, where U is a $t \times t$ orthogonal matrix whose columns are the eigenvectors of AA^T, Σ is a $t \times d$ diagonal matrix with its diagonal entries in descending order (the diagonal entries are $\sigma_1, \sigma_2, \ldots$), and V is a $d \times d$ orthogonal matrix whose columns are the eigenvectors of A^TA. Let $\mathcal{R}(A)$ be the row space of A and $r(A)$ be its rank. From our experiments, through the *SVD* of TFMs, we found that irrespective of their shape (Table 1), $r(TFM) = 1$ and $\sigma_1 >> \sigma_2 \simeq 0$. In such cases, the following holds true:

$$\mathcal{R}(A) \subseteq a\boldsymbol{v}_1 \subset \Re^d,\ a > 0 \quad (2)$$

$$dim(\mathcal{R}(A)) = r(A) = 1 \quad (3)$$

where $V = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_d]$, $\boldsymbol{v}_i \in \Re^d$ and dim is the dimension of a vector space. It can be seen that the $\mathcal{R}(A)$

Table 2: Highlights of LP using supervised learning. *Any type of learning algorithm. +More than one dataset.

Highlights	Classification algorithm	Heterogeneous network	Feature construction	Dynamic network	Dataset
PathPredict	Logistic Regression	✓	✓	×	DBLP
[4]	SVM-RBF	×	✓	×	DBLP+
[5]	Random Forest	×	✓	×	Condmat+
MetaFeatures	*	×	✓	✓	DBLP

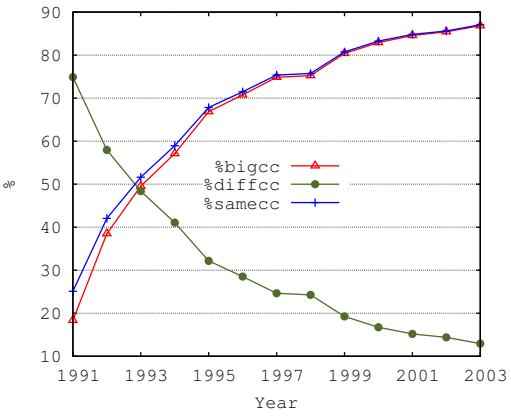


Fig. 3: Positive edges (with TI) Vs CCs in DBLP. *%bigcc*: proportion of positive edges that form in the BigCC, *%diffcc*: proportion of positive edges connecting two different CCs. Because of the affinity of BigCC with small CCs, *%samecc* curve is seen merging with *%bigcc*.

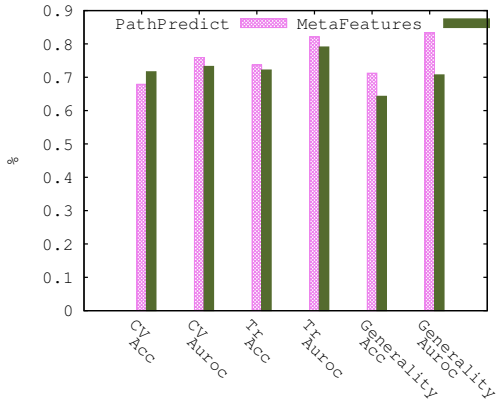


Fig. 4: Performance of meta-features based on Table 3 settings. The measures are CV: 10-fold cross-validation accuracy (Acc) and area under ROC ($Auroc$), Tr: Training set acc and auroc, and generalization performance on test years

is spanned by the first r columns of V, where $r = 1$ in our case. This clearly indicates that despite the shape of a TF matrix, all its rows are linearly dependent and they can be replaced by the single basis vector, $\boldsymbol{v}_1$. By utilizing $\boldsymbol{v}_1$, we get uniform lower-dimensional representation of all TFMs independent of their shape. We have also come across cases of an entire column being zero leading to a rank-deficient matrix, and we discuss this in our future work section

We hypothesize and validate by experiments that this TF matrix and its meta-features capture the discriminative information necessary for LP when cast as a binary classification problem. It should be noted that the time could be in any unit: seconds, minutes, years, etc., but should be uniform across all edges. Because of the SVD step, our method does indirect feature construction as opposed to the related work. In Table 1, for a graph of age $\breve{t}$, we construct TFM for all the potential edges. For those edges that appear during $\breve{t}-$th timestamp, we segregate them into one class and the potential edges become the second class yielding a two-class dataset. For these two classes of TFMs, we do feature extraction as explained in the previous paragraphs.

3. Evaluation

The two-class formulation of the LP problem in existing work [3]–[5] is given by Figure 2. Though our approach is more general, applicable to both static and dynamic graphs, for the sake of comparison, we replicated the experiments of one of the above works [3], except for a small change (discussed in section 4.1). We also used the same learning algorithms and metrics to compare the performance. We report our results on the DBLP [10] co-authorship network and compare with PathPredict [3]. We did some experiments for comparison with [4]and [5], but since these works do not address the generalization issue, which is important for us in a dynamic setting, we do not report the results here for brevity - TFM's performance was comparable with these works as well.

3.1 Experimental setup

Implementation is done in Python(Weave) [11] and C/C++ and uses Networkx graph library, LAPACK and LIBSVM [12] tools for related operations. Since the code involves multiprocessing routines, the execution is done on a 16-core 3.6GHz machine.

DBLP statistics: For the years 1937 to 2012, there are about 800K articles and 1.1M in-proceedings. The authors and edges are 1.1M and 4.1M in number. In the full grown graph, there are about 94K connected components and the largest component has 84% of the total nodes. We construct an undirected and weighted graph from this dataset for feature extraction. The graph growth is dynamic, and we allow for node and edge arrival, but not their deletion, hence this graph grows over time monotonically.

4. Results and Analysis

We conducted a series of experiments (Sections 4.1, 4.2), checking the generality and the skewed class distribution suitability to compare our results with PathPredict. The number of features in the TF matrix differs for each experiment and the reasons behind this are explained in the respective sections. For the curious reader, we provide table 2 showing the contrast between the link prediction solution strategies referenced in this work that use the supervised learning paradigm.

Table 3: Experimental Setup for Section 4.1

	Train years	Test Years	Graph type
PathPredict	1989 - 2002	1996 - 2009	Subgraph
MetaFeatures	1981 - 1988	1985 - 1992	BigCC

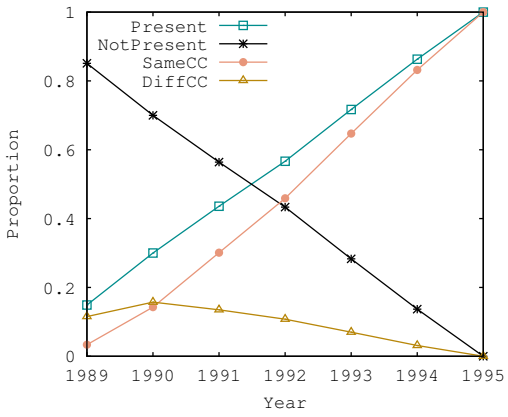


Fig. 5: DBLP (89 - 02). Training set author join patterns. $Present + Notpresent =$ total *#source authors* at the end of training years. Non-zero *diffCC* indicates many rank-deficient TFMs.

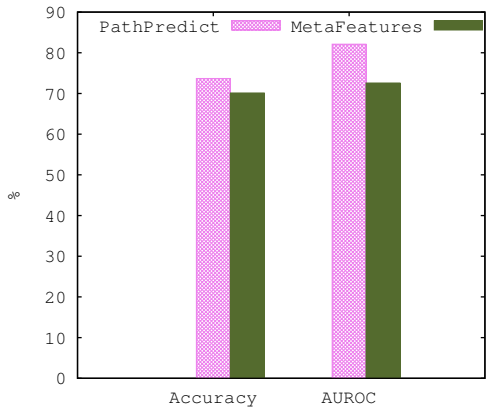


Fig. 6: DBLP (89 - 02). Performance comparison based on Table 4 settings. Accuracy and Auroc are calculated over the years shown in Figure 5 by adding the individual timestamp performance shown in Figures 7 & 8.

4.1 Experiment 1

Table 3 shows the differences in the setup between MetaFeatures and PathPredict for this experiment. We use the Logistic Regression implementation from LibLinear [13] to use the same learner as PathPredict. The approaches differ in the node selection process. While we do not restrict the source and target authors when considering potential edges, PathPredict picks close-to-prolific authors and their 2 or 3-hop authors as targets (indicated as subgraph in the Table 3 as this essentially reduces the graph to a subgraph of those nodes). BigCC in the table stands for the biggest connected component based on longitudinal growth (restricted to nodes in the bigCC of timestamp-1). To avoid negative entries in the TF table when calculating the SPL, we restrict the edges selection to bigCC. Implications of this are discussed in detail in future sections.

The importance of the bigCC in this work is stressed in Figure 3. The figure shows the pattern of edges that form in the graph for which we were able to collect the TI in relation to the connected components(*CCs*). It is clear from this figure that as the graph matures over years the positive edges form predominantly between two authors in the same CC, and all those components eventually get merged with the bigCC. As expected, the positive edges connecting two different CCs diminish quickly in number, but are significant in the initial stages. Hence, to accommodate all edges that form, however to avoid negative entries, we chose the bigCC based approach.

4.1.1 Results

Figure 4 shows cross-validation and training years accuracy, the area under the ROC curve and the model generality performance on DBLP. As can be seen, MetaFeatures stay close to current work in prediction despite our unconstrained source/target author selection, i.e., we consider two authors who are even more than 2 or 3-hops away for potential edge prediction.

4.2 Experiment 2

Here we set out to compare training set prediction performance between the two approaches, *MetaFeatures* and *PathPredict*. Table 4 shows the experimental setup. At this point, we would like to emphasize that it is common in the literature [3]–[5] to restrict the performance analysis to just training set based metrics like cross-validation accuracy. We believe that it is important to measure a solution to the LP problem by considering generalization performance as it is always possible to get good results by over-training on the training set. Whereas in experiment 1, we used 6 features including *SPL*, because of the disconnectivity of authors of potential edges, we had to remove the *SPL* feature from the TF table in this experiment to avoid feeding negative values to the SVD step. It is not only meaningless to have negative entries in SVD, the result of such a decomposition is hard to interpret since those negative values do not have physical significance. Indeed, we enter negative values to indicate that nodes are in different CCs and have a very large distance, but since mathematically negative values are smaller than their positive counter parts, their purpose is not served.

Refer to Figure 5 to see the proportion of authors of potential edges joining the graph in different CCs during the training years. Since we operate on growing graphs, the *present* and *notpresent* represent the potential authors that have joined the graph or not. The non-zero value of

Table 4: Experimental Setup. $*$: unconstrained node selection.

	Train years	Graph type	Learner
PathPredict	1989 - 2002	Subgraph	Logistic regression
MetaFeatures	1989 - 2002	$*$	Logistic regression

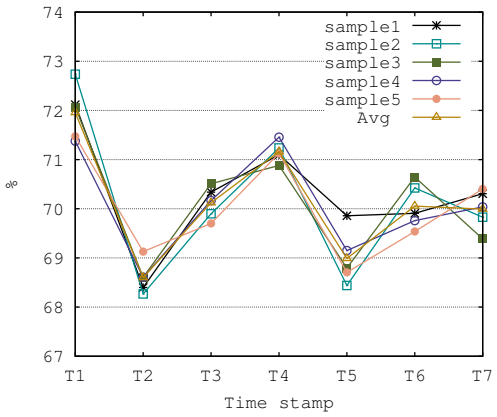


Fig. 7: DBLP (89 - 02). Accuracy Performance of Meta-Features over individual timestamps. Figure 6 shows the overall behavior.

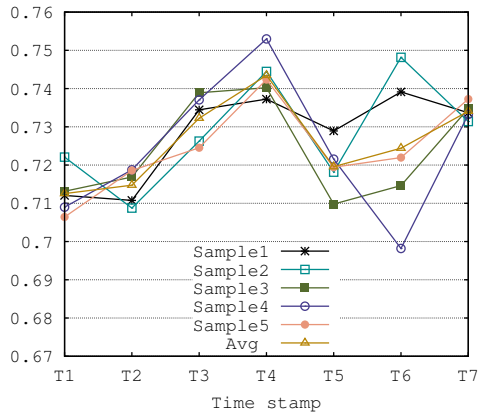


Fig. 8: DBLP (89 - 02). Auroc Performance of MetaFeatures over 7 timestamps shown in Figure 5.

the *DiffCC* curve shows that in the beginning a number of authors join the graph connecting different CCs. For example, about 14% of total #source authors are $present$ in the graph in the year 1989. Those TFMs are of shape 7x5. Of those, about 78% are in different CCs. Owing to the way we construct feature matrices, this is a significant detail because for many potential edges this leads to rank-deficient and sparse TFMs. The *sameCC* line in figure 5 represents the pattern of authors joining the graph in the same CC. Thus, *#present = #sameCC + #diffCC*.

4.2.1 Results

Figures 7 and 8 show the training set performance of MetaFeatures for the listed years. Since time information is an inherent component of our model, the figure shows the performance for each time stamp (in other words, potential edges have different TI). T1 indicates timestamp 1. Of all the edges of interest, we categorize the positive and negative edges into 7 categories depending on the number of years of information we collected on them. The five set of samples shown are created by under-sampling majority class edges, with positive edges (minority class) replicated each time (totally 5), thus these samples have equal number of edges from both classes in all the 5 subsets (this also reduces bias and allows for broader range of majority samples to be included in learning). For each sample, we set the hyperparameters after 10-fold cross validation. The LP problem is an ideal case of the extreme-skew class distribution based binary classification problem, and we conduct some experiments to this end in Section 4.3. Refer to Figure 6 for overall comparison. With just 5 features, MetaFeatures performs comparably (recall *SPL* is removed here). This shows that choosing the right graph topological measure helps to improve the performance. We plan to investigate this in detail with more complex features.

4.3 Experiment 3

This experiment was designed to solve the LP problem preserving its imbalanced nature. Figure 9 captures the acute level of imbalance between the two classes of edges - those that form (minority) and those that do not.

Traditionally, though the existing work recognizes this problem as such, the solution strategies are invariably provided after random sampling (under- sampling) of majority class samples - in this case, the edges that do not form. Both the experiments 1 and 2 (sections 4.1 and 4.2) follow this line of solving the problem with the TF table, but this experiment was designed to see if retaining the original distribution of classes helps for better prediction performance. Figure 10 shows that the positive edges are only a small portion of edges that form (which we know is a negligible portion of those that do not form). This indicates that a significant portion of edges form because of exogenous reasons (e.g., an author shifting to a different university) and their TI cannot be collected. Of those positive edges, a small portion are slings. In this work, we do not consider such edges for prediction, as they do not include two separate authors. For example, in figure 10, during the year 1990, of the total number of edges that formed, only about 3.5% were not slings and had TI associated with them.

Since we find various types of classification algorithms used in conjunction with the LP problem like Bagging with Random Forests [5] and SVMs with RBF [4], we experimented with the *polynomial kernel* (Figure 12) in combination with SVMs. The hyperparameters were set based on 10-fold stratified cross validation experiments. To make the SVM learner unbiased towards the majority class, we did cost-sensitive learning with minority class misclassifications highly penalized. We used LIBSVM for all the trials under this section. The DBLP years we used were from 1980 to 1985 and the metric is training set accuracy. The class distribution is shown in Figure 11 (skewed class

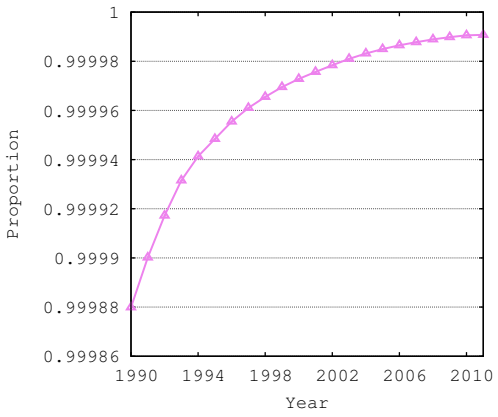


Fig. 9: DBLP over years. Increase in sparsity of the edges with TI in Figure 10.

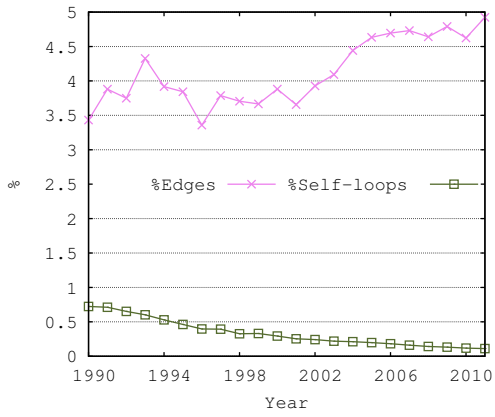


Fig. 10: DBLP. Fraction of edges that form in the graph. This directly controls the % of positive edges in Figure 10.

Fig. 9: DBLP over years. Increase in sparsity of the edges Fig. 10: DBLP. Fraction of edges that form with TI. *Edges*: have two distinct authors. *Self-loops*: Single author papers.

distribution is apparent in the figure).

In addition to the 6 features discussed in section 4.1, we also used *communicability* measure [8] in this experiment. Because *SPL* and *communicability* are path-based measures, we restricted our graph to the bigCC based longitudinal growth.

4.3.1 Results

Figure 12 shows the individual class based accuracy. Sometimes, we had to set the SVM hyperparameter C to even 1000 with misclassification cost of *pedges* class to 100. Though the accuracy values are high, we plan to investigate further here with other years for two reasons: one, there are variations in the results for the *pedges* case and two, it is a well-known fact that a high C will overfit the data. The accuracy for *medges* could be because of C or their sheer count. The last two pairs of bars show the effect of increased penalty to the minority class.

5. Related work

We broadly classify the solution strategies of the LP problem into 5 main categories: 1) Technique, 2) static/dynamic graphs, 3) heterogeneous networks, 4) LP flavor (candidate edges for prediction) and 5) the domain of application. Some of the literature is discussed in Section 1 and throughout the paper. For a solution using a regularized matrix factorization of the adjacency matrix of the graph to learn the latent features using stochastic gradient descent, refer to [14]. This solution is on static homogeneous graphs and domain of application is generic, however they restrict the prediction to nodes that are only two hops away which will not work in scenarios shown in figure 3. Techniques that use spectral theory [15] use graph kernels and model link prediction as spectral transformation of the Laplacian matrix. This approach is on static homogeneous graphs and works only with the bigCC edges. One main issue with the matrix methods ([14], [15]) is the difficulty in accommodating new nodes. While most of the works concentrate on finding new links, Huang and Lin [16] use a time series methodology to find the repetitive links. MRIP [17] solves the problem on heterogeneous networks and does temporal analysis, however their feature construction technique does not involve time directly. Our work aligns with the solution strategies that extracts proximity features from a homogeneous graph to apply supervised learning, but differs from the literature (Table 5) because we solve the problem for dynamic graphs. The table also shows the list of features used in MetaFeatures and in the related works. For a survey on the LP problem, refer to [1], [18].

Table 5: Representative list of features in this and other relevant work for Link Prediction using binary classification.

PathPredict [3]	[5]	[4]	MetaFeatures
• Path Count •Random Walk • Normalized Path Count	• Degree • Common Neighbors • PropFlow • Preferential-Attachment	• Sum of papers • Shortest Distance • Sum of neighbors • Second Shortest Distance	• Shortest-path length • Preferential Attachment • Adamic- Adar • Common Keywords

6. Conclusion and Future Work

We presented an indirect way of constructing feature vectors encapsulating time and tested its suitability for the LP problem framed in the binary classification setting. Through various experiments, we could see that having a correct set of features is important for better classification performance. We found that many TF matrices have zeros in an entire column, leading to singular matrices. This is a direct consequence of potential edges forming by connecting different connected components. We will have this problem as long as we include path and neighbor based measures.

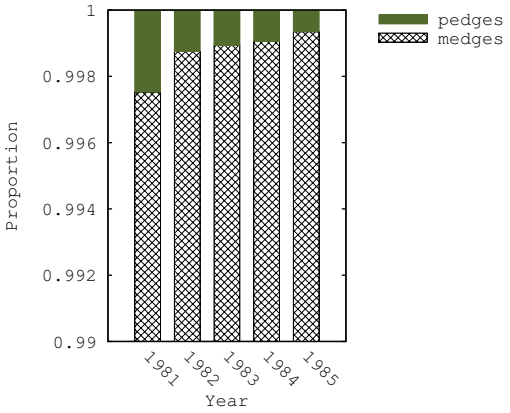


Fig. 11: Class distribution over the years 1980 - 1985 showing dominance of negative edges, edges with TI that never form (*medges*). Greater the age of the graph, larger the skew in class distributions because of sparsity (Figure 9). *pedges*: edges with TI that eventually form.

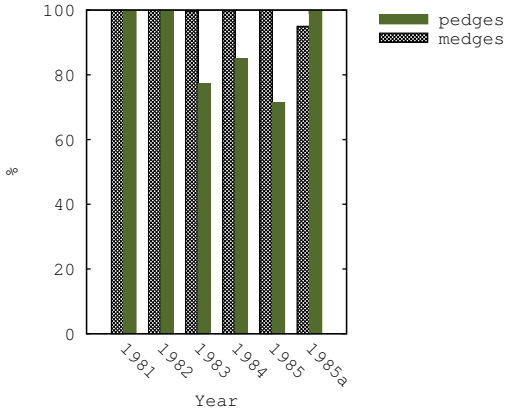


Fig. 12: Accuracy using MetaFeatures on skewed data. Despite the imbalance in class distribution (Figure 11), MetaFeatures perform well on the majority *medges* and comparably on *pedges*. *1985a*: with higher penalty for minority class misclassifications, shows improvement over no-penalty case (*1985*).

MetaFeatures performed comparably in most of the cases despite the unconstrained node selection, making it suitable for generic link prediction problems in social networks. Though we have predominantly aimed at comparing our results with the past work in this paper, it is important to note that even without any separate label acquisition period, our model can still verify link occurrence in the future in a pure dynamic setting. In that case, labels are acquired as the graph grows, giving us a training dataset for any of the future link predictions.

In our future work, in addition to verifying the performance of MetaFeatures with dynamic graphs from other domains, we plan to investigate the problem of rank-deficient TF matrices. There are two ways by which this problem could be addressed - one is to remove those features and complement them with new feature(s) that are robust against this problem (such a feature would have a non-zero value irrespective of the presence of authors in different CCs) or conduct experiments by segregating edges that connect and do not connect CCs.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant Nos. 1318913 and 1318957. The authors would like to thank the anonymous reviewers for their constructive comments and suggestions.

References

[1] L. Getoor and C. P. Diehl, "Link mining: a survey," *SIGKDD Explor. Newsl.*, vol. 7, no. 2, Dec. 2005.

[2] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in *Proceedings of eighth ACM SIGKDD international conference on Knowledge Discovery and Data Mining*, 2002.

[3] Y. Sun, R. Barber, M. Gupta, C. C. Aggarwal, and J. Han, "Co-author relationship prediction in heterogeneous bibliographic networks," in *ASONAM*, 2011.

[4] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *In Proc. of SDM 06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[5] R. Lichtenwalter, J. T. Lussier, and N. V. Chawla, "New perspectives and methods in link prediction," in *KDD*, 2010.

[6] D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks," in *Proceedings of the twelfth international conference on Information and knowledge management*, ser. CIKM '03, 2003.

[7] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*. O'Reilly Media Inc, 2009.

[8] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using NetworkX," in *Proceedings of the 7th Python in Science Conference (SciPy2008)*, Aug. 2008, pp. 11–15.

[9] K. Yang and C. Shahabi, "A PCA-based similarity measure for multivariate time series," in *Proceedings of the 2nd ACM international workshop on Multimedia databases*, 2004.

[10] M. Ley. (1993) Dblp.uni-trier.de: Computer science bibliography.

[11] E. Jones, T. Oliphant, P. Peterson *et al.*, "SciPy: Open source scientific tools for Python," 2001.

[12] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011.

[13] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, "LIBLINEAR: A library for large linear classification," *Journal of Machine Learning Research*, vol. 9, 2008.

[14] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in *ECML/PKDD*, 2011.

[15] J. Kunegis and A. Lommatzsch, "Learning spectral graph transformations for link prediction," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 561–568.

[16] Z. Huang and D. K. J. Lin, "The time-series link prediction problem with applications in communication surveillance," *INFORMS Journal on Computing*, pp. 286–286, 2009.

[17] Y. Yang and N. V. Chawla, "Link prediction in heterogeneous networks: Influence and time matters," in *Proceedings of the 2012 IEEE International Conference on Data Mining*, 2012.

[18] L. Lu and T. Zhou, "Link prediction in complex networks: A survey," *CoRR*, vol. abs/1010.0725, 2010.

A Flexible Feature Selection Framework for Improving Breast Cancer Classification from Sparse Spectral Count Proteomic Data

Lanfei Shi[1], **Himanshu Grover**[2], **Jeya B. Balasubramanian**[1]
Kumar Kolli[3], **Craig D. Shriver**[4], **Vanathi Gopalakrishnan**[1,3]
[1]Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA, USA
[2]Center for Health Informatics and Bioinformatics, New York University, New York, NY, USA
[3]Department of Biomedical Informatics, University of Pittsburgh, Pittsburgh, PA, USA
[4]Murtha Cancer Center, Walter Reed National Military Medical Center, Bethesda, Maryland, USA

Abstract— *Spectral count proteomic data, for disease prediction and biomarker discovery, has become increasingly popular owing to advantages of label-free quantification. However, such data brings great challenges for quantitative analysis because of its high variability and low sensitivity. We propose a flexible framework that uses Bayesian Rule Learning and Negative Binomial Regression as two representative feature selection methods to improve the predictive performance and identify differentially expressed biomarkers for case-control discrimination. We demonstrate the application of this framework to a new spectral counting dataset for breast cancer classification that we generated. We improved the classification performance of baseline SVM when using feature selection by 20.1% measured by averaged F-score across 20 runs of 10-fold cross-validation. Furthermore, we annotated the identified putative protein markers through functional analysis of biological pathways.*

Keywords: Biomaker Discovery, Spectral Count Data, Protein Quantitation, Feature Selection, Breast Cancer Classification

1. Introduction

Quantitative analysis of proteins in biological samples holds a lot of promise for discovery of diagnostic and prognostic disease biomarkers. Recently, label-free mass spectrometry (MS)-based methods have generated great interest in the proteomics community since they estimate protein abundance in a high-throughput fashion but without the need for time-consuming and expensive sample labeling steps [38], [28], [43], [23]. In this paper, we focus on one of the two categories in label-free proteomics quantification, called spectral counting (SC). SC method involves counting the number of MS/MS spectra in an experiment that is confidently assigned to peptides of a protein, and then using the count (typically after some normalization) as a surrogate for the protein's abundance. The SC of a protein has been shown to have a strong linear correlation with its abundance [28], [22], [2], [31], [23]. Its underlying rationale is that an increase in protein abundance typically results in an increase in the number of its proteolytic peptides and subsequently results in an increase in the number of identified MS/MS spectra (spectral count). Therefore, spectral count profiles can be used to identify differential protein expression patterns associated with disease and help to develop predictive models of disease status [4], [23]. From the point of view of machine learning, prediction of disease status entails building a classifier/model from training data to accurately predict whether a new patient (test case) has a certain disease or not. However, modeling SC data are non-trivial as a result of the inherent challenges and limitations of SC-based quantification [38], [4], [23]. First, SC data is strongly biased to detect abundant proteins, hampering the generation of a comprehensive list of differential markers. Second, low SC values are unreliable especially when working with a small number of replicates. Yet, typically a sizable percentage of identified proteins is characterized by low spectral counts. Third, peptides that map to multiple proteins complicate the assignment of spectral counts and identification of proteins. Proper strategies should be used to reduce the assignment errors. Fourth, limited replicates bring low reliability of variance estimation, resulting in wrongly selecting or overlooking a certain protein[29], [30]. These limitations - variability/noise and sparseness - increase difficulty in performing robust estimation and inference with predictive models.

Previous analyses [4] on SC data for detection of differentially expressed proteins include statistical tests, such as F-test and Analysis of Variance, and classification methods for disease prediction, such as Support Vector Machine (SVM) model. To tackle the challenges in classification of sparse spectral counting data with low reliability, we utilize the differential markers/features that reduce the noise-to-signal ratio and add predictive value for class discrimination. We identify these differential markers through feature selection. Generally, no single feature selection method can always provide optimal results [5]. Thus, we develop a flexible feature selection framework unconstrained by classification models. It explores the best combinations of feature subsets obtained from different feature selection methods through set calculations (intersection and union). To our knowledge, other domains have explored similar feature selection methods [37], [21], [40]. In gene expression classification,

intersection of different feature subsets has been shown to outperform the individual methods [37]. In stock prediction, the most discriminative features are obtained by calculating the intersection and union set [40]. These promising results inspired us to apply such feature set calculations to SC data.

A key component of our framework is the adoption of recently reported, useful proteomic biomarker discovery methods - Negative Binomial Regression (NBR) [43] and Bayesian Rule Learning (BRL) [11], [26]. We chose them as representatives since they both generate well-performing models for SC data and compensate each other in terms of feature independency assumption. NBR is a univariate filtering approach to feature selection that assumes each feature/protein is independent of others, while BRL is a multivariate wrapper method that takes the interactions among features/proteins into account in the modeling process. Note that we are not suggesting BRL and/or NBR to be the optimal choice for exploring SC data classification. Rather, these are only representatives, and can be substituted or supplemented with any other feature selection method or classification model.

We tested our framework on a new breast cancer SC dataset, which we processed and generated from clinical samples. The experiment showed that our feature selection framework improved classification performance by 20.1%. The identified markers were further validated to be predictive through functional analysis. Therefore, our main contributions of this paper are: 1) abstracting feature subset calculation into a general framework and applying the framework to a new domain - classification development for SC data, 2) adopting effective feature selection methods within the framework for this new domain, 3) applying the framework to a new proteomic dataset for discrimination of benign versus malignant cases of breast cancer in pre- and post-menopausal women, 4) generating and publishing this new SC dataset, and 5) annotating the identified markers through bioinformatics analyses.

The remainder of this article is organized as follows. Section 2 introduces our feature selection framework and feature selection methods of NBR and BRL in depth. Section 3 describes the experiment and results for applying the framework to the new SC dataset, along with the explanation of the procedure to process and generate SC data. Section 4 presents the functional analysis on the selected markers. Section 5 concludes with future work.

2. Methods

2.1 Feature Selection Framework

Our proposed framework contains three components: feature selection, feature subset operation, and classifier. The choice of feature selection methods and the classification model depends upon the data and the task. The framework illustrates the procedure of how we utilize the features extracted, from the feature selection methods we choose, to further generate a new feature subset that can enhance the classification performance.

Specifically, we first apply different feature selection methods that are appropriate for a given task and a given data type to extract multiple feature subsets. We then apply set union and intersection operations to generate additional combinations of feature subsets. The union operation potentially provides information that might have been missed by one of the approaches, while the intersection operation keeps only the selected features from multiple approaches, indicating better stability and discriminatory power [37], [40]. Finally, we send all the feature subsets - one for each feature selection method plus union and intersection - to the classifier respectively and pick the subset that gives us the best classification performance. The rationale behind this strategy is to best employ different feature selection methods such that their selected features complement and/or compensate for each other. The flexibility of our framework allows the choice of appropriate feature selection methods and/or classifiers based on the characteristics of datasets and tasks.

2.2 Negative Binomial Regression

As a result of the challenges in modeling SC data that we mentioned earlier, estimating variance becomes difficult [23]. Previous studies have identified Poisson distribution to be a proper model to estimate means instead, which assumes that the mean equals variance [7]. Later research on spectral count data mining has employed Negative Binomial regression for lung cancer classification modeling [43] with good performance. Negative Binomial Model uses an extra parameter for modeling over-dispersion, thus adding generality and flexibility in modeling count data.

In NBR, the dependent variable is the spectral count of a protein while the independent variable is an indicator variable, discriminating the benign versus malignant cases. To reduce the bias for large-abundance spectral count, we include the total spectral count for the protein across all the samples as an offset. Eq.(1) shows the mathematical representation of the model:

$$\log(SC_m^{P_i}) = \beta_0 + \beta_i \cdot X_m + offset(\log(\sum_{m=0}^{N}(SC_m^{P_i})) \quad (1)$$

X_m is an indicator variable of the case/control status for the sample m; P_i is the i_{th} protein; $SC_m^{P_i}$ is the spectral count of the i_{th} protein, in sample m. Accordingly, we conducted negative binomial regression for each protein i, over all samples to estimate the regression parameters. The p-value of coefficient β_i is the evidence of significance of the i_{th} protein/feature. We set the cut-off p-value for protein selection to be 0.05, thus considering features with p-values

smaller than 0.05 as statistically significant. These features comprise the selected feature set from the NBR method.

2.3 Bayesian Rule Learning

Bayesian rule learning is a useful data mining method for the discovery of biomarkers from high-dimensional biomedical data [11], [24], [26], [9]. Rule generating methods are useful in biomedicine because of their model interpretability, with modular knowledge representation and tractable inference. We can consider it as a decision tree method within the machine-learning framework. Bayesian rule learning (BRL) learns a constrained Bayesian network from the training data and infers a set of rules from them. A single rule from the inferred rule set matches the pattern seen in a single instance of the test data. The rule predicts the class with the highest posterior probability of the class given the learned model.

BRL requires specification of some parameters. For detailed explanation of the method and parameter set-up, please refer to [11], [24], [9]. Briefly, it has six searching algorithms: global- greedy, global- beam search, local decision tree- greedy, local decision tree- beam search, local decision graph- greedy, and local decision graph- beam search. Since BRL takes nominal input, BRL uses a heuristic effective Bayesian discretization method (EBD) as the discretization algorithm to transform continuous values to nominal values [26]. The discretizing parameter lambda is another parameter that influences the classification performance. Empirically, on a few publicly available biomedical datasets, a lambda value of 0.5 was observed to give the optimal classification performance when the number of attributes is large (thousands), and lambda value of 2.0 or greater, for datasets where the number of attributes is smaller (few hundreds) [26]. Accordingly, we run our experiments on different lambda values within reasonable range to get the best classification performance while EBD retains a stable selection of proteins. The chosen value of lambda, based on our experiments, is reported in the experiment section.

3. Experiments

Here, we describe the detailed experimental procedure used for generating an SC dataset for breast cancer from the clinical samples. We demonstrate our proposed feature selection framework using this new SC dataset.

3.1 Dataset

3.1.1 Clinical samples

The breast cancer serum samples used in this molecular profiling study consists of benign (n=40) and invasive (n=39) cases. The mean age of benign cases is 52.78 (range 31 -79 years) while the mean age of invasive cases is 53.65 (range 31 - 85 years). All the invasive cases were in stage 1. We obtain the serum samples, pre-surgically, from consenting subjects, enrolled in HIPAA-Compliant, IRB approved clinical protocols at CBCP (clinical breast care project), Walter Reed Army Medical Center (WRAMC) and are banked at the tissue repository of Windber Research Institute for molecular profiling research.

3.1.2 LC/MS/MS analysis

The serum samples (n=79) were subjected to immunodepletion on an IgY12 mixed antibody column (Beckman) using AKTA explorer HPLC to remove the twelve most abundant serum proteins. For the targeted removal of high abundant proteins, we utilized a volume of 25 μL from each serum sample. We followed the LC conditions mentioned in the protocol (Beckman) to collect the flow through fraction. The flow through fractions from the respective samples was concentrated using an AMICON 5K molecular weight cutoff filters (Millipore) and we estimated the protein concentrations using the BCA protein assay (PIERCE). We brought up the concentrated protein samples (25μg of protein for each sample) in 6.25μl of TFE, 2.5 μL of 20mM DTT, 1.25μL of 500mM ammonium bicarbonate to give a final volume of 25μL and 25mM amm bicarbonate. We heated the mixture at 60°C for 1hr, cooled and added 1μL of 200mM IAA, vortexed and incubated at room temperature in the dark for 1hr. We quenched the excess IAA by adding 2.5 μL of DTT and dilute the solution 7-fold with the 25mM amm bicarbonate and perform trypsin digestion by the adding trypsin at a ratio of 1:25 (w/w) enzyme:protein. We quenched the digestion mixture with 2 μL 2% formic acid and directly analyzed on LTQFT instrument.

We analyzed the protein digest samples in triplicates on a high performance LTQFT (Thermo Electron) mass spectrometer coupled to an online Surveyor LC system equipped with an autosampler. The LC system employed a trap column and a reverse phase analytical column connected via a 10-port switching valve. We loaded the samples onto a C18 trap column (Agilent) using sample pump at a flow rate of 10 μL/min (pre column splitting). Following the washing for 5 min, we switched the trap in-line with the analytical column (Biobasic C18, 180μm, 10cm) and used the MS pump for eluting the peptides (flow rate 1.5 μL; pre column splitting) onto an analytical column for further separation and online nano-ESI LC/MS/MS analysis. The gradient for reverse phase chromatography consisted of 2% B (Acetonitrile 0.1% formic acid) - 45% B for 85 min. We operated the LTQFT in a data-dependent mode and acquired the full FTMS spectrum with a resolution setting of 100K (at m/z 400). We set the parallel detection to analyze the top five intense peptides in the ITMS for peptide sequence information. We enabled the dynamic exclusion of precursor masses with a repeat count 2 and repeat duration 20 sec to exclude the previously analyzed peptide masses for 2 min.

3.1.3 Generation of SC data

We require several processing steps to generate SC data from clinical data, as described next. First, we assigned

peptides to large collections of experimental MS/MS spectra via database searching using the open-source software called Crux [32]. For each query MS/MS spectrum, Crux searched a database of peptides to extract candidates, which were then scored and ranked against the query spectrum using a scoring algorithm. Each sample had three replicates and we analyzed them independently during the process. We used the primary score from Crux to get top-scoring peptides as potential identifications.

Based on what peptides were identified (in other words, present in the sample), the next step was to compile a list of proteins present in the sample from this list of identified peptides. This was a challenging process due to a lot of redundancy in the human proteins. Specifically, in several cases a peptide could be a subsequence in multiple protein sequences, making presence of a specific protein in the sample ambiguous. We used the Barista program [39], which was based on a complex artificial neural network model, to compile a list of proteins present in the corresponding sample from the list of identified peptides. The false discovery rate (fdr) threshold was fixed at 1% for this step. More details on fdr control as well as the Barista algorithm can be found in the original reference [39].

From the results of application of Barista to all the 79 samples and 3 replicates, we compiled a global non-redundant list of all the proteins and peptides identified in the entire dataset. Next, we created a bipartite graph from this list, where each of the peptides and proteins formed a node and there was an edge from peptide to all the proteins it belonged to. As in the IDPicker algorithm [27], we then used a greedy set cover algorithm to compile the minimal or parsimonious list of protein nodes that explained (covered) all of the peptide nodes in the dataset. This resulted in a total of 189 proteins (or protein groups).

Finally, for each protein in the parsimonious list, the adjusted spectral count was computed as follows: a) counted the number of times a unique peptide of the protein is identified; b) For peptides that were shared across multiple proteins, their identification count was distributed across the sharing proteins in proportion of each sharing protein's unique count (from step 'a'), rounded to the nearest integer. Adjusted spectral count for each protein was the sum of unique counts and the distributed shared counts from 'a' and 'b' above.

The 79 samples contain 39 cases and 40 controls. We filtered out the features with low amount since spectral count less than 3 are typically considered unreliable. We followed the steps: 1) For each feature/protein i, we compared the number of samples whose spectral count is less than 3, $Num(Sample)^i_{SC<3}$, to the number of Case class instances, $Num(Case)_i$; 2) If $Num(Sample)^i_{SC<3} > Num(Case)_i$, this feature/protein is considered involving too much uncertainty and thus removed. Typical filtering method is simply resetting the low-abundance count value to zero, which modified the distribution of protein abundance. This step reduced the number of proteins from 189 down to 98. We used this filtered spectral counting data for all the subsequent experiments.

3.2 Experiment Setup

The experiment focused on demonstrating how to apply our feature selection framework to improve classification performance in SC data. Therefore, selecting the best classification model is beyond the scope of our paper. We evaluate each feature subsets' quality by their performance in classification. We chose SVM model because it is widely used in proteomics, and bioinformatics in general [4], [36]. We use 10 fold cross-validation to evaluate the classification performance. Given the limited size of our dataset, leave-one-out cross validation is also a viable option with benefit of providing more training data for building the models. However, 10 fold cross-validation offers more stable measurements using larger testing sets in comparison with the leave-one-out approach which is easily biased due to single testing sample in each round of model building [34].

We used WEKA 3 [12] platform to perform classifications. For the SVM model, we incorporated LibSVM3.17 [6] library into WEKA, and used linear kernel to avoid over-fitting given the much larger number of features than samples. We report F-score with Case class to be positive as the measurement of classification performance. F-score is the harmonic average of Precision and Recall. For each feature subset, we performed 10 fold cross validation on SVM models 20 times, by generating different random splits of data each time. We reported the average, and standard deviation over the 20 runs in the experiment result section.

We re-implemented the NBR algorithm in the R language. As described earlier, the offset chosen for each protein was the sum of spectral counts across all samples for that particular protein/feature. After running the regression models, we retained the proteins with p-value of regressions' estimated parameters less than 0.05 as statistically significant. For the BRL, after transforming the data to the BRL-compatible format, we used the in-house BRL implementation to generate all the rules and results [11]. The final classifier results were literally classification results (on training data and 10 fold cross-validation) from LibSVM that included the features that BRL selects to generate a model. Within the experiments of BRL, we used 20 fold cross validation to compare performance. We found the discretization parameter lambda to be very sensitive to the final model learned by BRL. We used the classification performance over cross validation to indicate how well the parameters were set. This cross validation had nothing to do with final classification using LibSVM. By experimenting within reasonable range according to different dimensions of features, lambda for original data with 198 features was finally set to 1.1 while filtered data with 98 features was set

Table 1: Identified Biomarkers/Selected Features

Protein	Mean (Case)	Std. (Case)	Median (Case)	Mean (Control)	Std. (Control)	Median (Control)	Range
IPI00930442.1	7.82	9.54	2	9.53	7.40	10	[0, 28]
IPI00896419.3	75.48	19.73	75	84.38	18.21	85	[23, 121]
IPI00847179.1	22.51	22.95	20	29.5	23.44	31	[0, 82]
IPI00304273.2	41.71	23.25	39	37.9	26.82	31.5	[0, 98]
IPI00032328.2	27.38	17.24	29	19.53	17.28	23.5	[0, 74]
IPI00215894.1	31.03	21.14	29	44.23	23.57	32	[0, 87]
IPI00946337.1/IPI00892870.1	11.10	14.30	10	17.1	21.47	13.5	[0, 87]
IPI00022895.7	78.92	8.36	81	75.55	8.21	75	[55, 94]
IPI00291866.5	47.97	14.29	50	49.9	8.95	52	[11, 70]
IPI00166729.4	38.87	6.45	40	38.58	4.97	39	[21, 52]
IPI00022426.1	22.13	7.11	21	20.33	5.36	20	[13, 37]
IPI00022394.2	6.41	3.47	6	5.2	3.71	5	[0, 14]
IPI00026199.2	6.69	3.31	6	5.33	3.33	5	[0, 15]
IPI00477992.1/IPI00643948.2/IPI00967849.1	12.54	5.90	13	12.38	3.91	12.5	[0, 22]

Table 2: SVM Classification Results using averaged F-scores

Features	Baseline	BRL	NBR	Union	Intersection
# of features	98	10	6	14	2
Averaged F-Score	0.512	0.661	0.674	0.642	0.713
STD	0.041	0.037	0.023	0.040	0.019

to 3.2. Through six different searching algorithms available, we found that using the same lambda, the proteins that each selected were identical. Thus we only reported BRL results obtained from local greedy searching method.

3.3 Experimental Results

By using the two feature selection methods and the parameters we described, we obtained 10 proteins from BRL and 6 proteins from NBR. There were 2 proteins in common. These 14 proteins comprise our identified discriminative protein markers, leading to the feature dimension's reduction rate of 85.7% from filtered data and 92.6% from original data. We list these 14 markers in Table 1 along with some descriptive statistics between Case and Control classes, namely range, mean and standard deviation. Functional analysis using bioinformatics techniques are discussed in the next session. The first two proteins are from the intersection of the two selected protein lists.

Table 2 lists averaged F-scores as classification performance using SVM with 10 fold cross validation. The F-score was obtained by averaging the 20 runs of cross validation, as described earlier, and standard deviation was calculated accordingly. The results show that using either feature selection approach improves classification performance, with NBR giving slightly better results.

The best result for this dataset is obtained using the intersection of two subsets. *We observe that using only the intersection set increased the classification performance by 20.1% over baseline (from 51.2% to 71.3%) where baseline represents the LibSVM using filtered SC data. Note that this experiment is to demonstrate the procedure and effectiveness of our feature selection framework. We do not claim the superiority of intersection over union from this experiment. It could be possible that union outperforms intersection in other SC datasets, and will be pursued in the future.*

3.4 Discussion

Feature selection methods for SC data are still emerging, thus we compared our framework with state-of-the-art feature selection methods including Information Gain [8], and ReliefF [41], [18], [35]. We used the top 10 ranked features obtained by these two feature selection methods respectively and performed SVM classification in WEKA [12]. We measured the performance consistently using 20 runs of 10-fold cross-validation. For features selected by Information Gain, the average F-score of classification was 0.627 with a standard deviation of 0.03, and for features selected by ReliefF, the average F-score of classification was 0.618 with a standard deviation of 0.031.

Our choice of BRL and NBR, both standalone, performs better than the two feature selection methods examined here, with the SC data. We tried several other feature selection methods available in WEKA as well, and did not see an improvement over BRL and NBR. The experiments not only demonstrated the effectiveness of BRL and NBR as feature selection methods for SC data classification, but also indicated that the intersection set operation in our framework gives larger improvement to the performance than the readily available feature selection methods.

Table 3: Functional analysis of 8 of the 11 genes

Functions Annotation	P-value	Molecules
Cholangiocarcinoma	4.27E-06	A1BG,AMBP,KNG1
Liver cancer	6.08E-04	A1BG,AMBP,ITIH4,KNG1
Digestive organ tumor	1.45E-03	A1BG,AMBP,AZGP1 GPX3,ITIH4,KNG1
Prostatic intraepithelial neoplasia	2.40E-03	GPX3,SERPING1
Serous ovarian carcinoma process	3.18E-03	GPX3,SERPING1
Ovarian cancer	1.06E-02	GPX3,ITIH4,SERPING1
Mucinous ovarian cancer	1.53E-02	GPX3
Clear-cell ovarian carcinoma	1.66E-02	GPX3
Incidence of liver tumor	1.78E-02	ITIH4
Acinar-cell carcinoma	1.85E-02	AZGP1
Rectum cancer	2.22E-02	GPX3
Pancreatic ductal adenocarcinoma	3.54E-02	AZGP1
Carcinoma	3.93E-02	A1BG,AMBP,AZGP1 C1QB, GPX3,ITIH4 KNG1,SERPING1
Small cell lung cancer	4.03E-02	KNG1

4. Functional Analysis

We performed the functional analysis for the 14 features using Ingenuity Pathway Analysis (IPA; Ingenuity Systems, www.ingenuity.com). The IPA mapped 13 out of the 14 features using their International Protein Index identifier to their gene symbols. Two pairs of features mapped to the same gene symbol. We perform functional analysis for the 11 gene symbols, using IPA.

The IPA identified significant association of the set of gene symbols with diseases including development disorder, hereditary disorder, immunological disease, cancer, and gastro-intestinal disease. Cancer was found to be associated with 8 of the 11 genes, and their functional annotation shown given in Table 3. A few interesting examples from these 8 genes include- ITIH4, the down-regulation of human ITIH4 mRNA in hepatoblastoma is associated with liver cancer in humans [42]; KNG1, the kininogen gene family that has been linked to a role in the adhesion of breast cancer cell lines [15], and SERPING1 (serpin peptidase inhibitor, C1 inhibitor), which has been associated to the function of binding of macrophage cancer cell lines [20].

The results are promising in that our framework can be applied to obtain predictive models with useful markers for breast cancer discrimination from SC data.

5. Conclusions & Future Work

We propose a feature selection framework to improve the performance of SC data classification by reducing the noise in the feature space. We acknowledge that there is no single feature selection method or classification algorithm that uniformly does well on all datasets and tasks. Instead, we strongly believe that it is more flexible and reasonable to provide a classification process or pipeline with an open-ended feature selection framework, leaving final decision of which features to use based on the classification results. In summary, our framework enables evaluation and choice of features associated with best classification performance by comparing the individual, union and intersection of all feature subsets extracted by different feature selection methods.

We included two representative feature selection methods, NBR and BRL, in our framework, and demonstrated its application to SC dataset for improving disease classification and biomarker discovery. We also released a new breast cancer dataset to the public, contributing to the limited proteomic data in the public domain on the disease. In terms of experimental results, the intersection of two feature subsets performed the best among other feature subsets, with 20.1% improvement of classification performance compared to the baseline model. Functional analysis validated the identified putative proteins as useful markers for breast cancer discrimination, in turn revealing the effectiveness of our framework and two feature selection methods in improving predictive models and identifying biomarkers.

Our work serves as an initial exploration on proteomic analysis of disease prediction and biomarker discovery using SC data. The promising results are encouraging for the application of our methods to additional SC datasets to test their generalizability. Several generally applicable feature selection methods have been demonstrated to be very effective in other domains [33], [21], [36], such as Random Forest [10], Weight vector of SVM [14], Neural network [3] and Lasso [16]. It would be interesting to explore these methods within our framework. Additionally, while we have mainly focused on the quality of the selected features in this work, it is also important to evaluate the stability of the features since it gives us better understanding on the features through providing additional information of variability [13], [25], [1]. Our framework will, in future, allow investigating this aspect in context of several feature selection methods. Moreover, we want to enhance our framework in a way that better combines or fuses all feature selection methods instead of just utilizing the feature sets they generate [19], [17].

Acknowledgment

The authors are grateful for support from the following grants from the National Institutes of Health: National Library of Medicine grant number R01LM010950, NIGMS grant number R01GM100387 and NCI grant number P50CA090440 as well as a previous grant from the Department of Defense/TATRC number W81XWH-05-2-0066. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

[1] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, and Y. Saeys. Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. *Bioinformatics*, 26(3):392–398, 2010.

[2] R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422(6928):198–207, 2003.

[3] G. Ball, S. Mian, F. Holding, R. Allibone, J. Lowe, S. Ali, G. Li, S. McCardle, I. O. Ellis, C. Creaser, et al. An integrated approach utilizing artificial neural networks and seldi mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers. *Bioinformatics*, 18(3):395–404, 2002.

[4] M. Bantscheff, M. Schirle, G. Sweetman, J. Rick, and B. Kuster. Quantitative mass spectrometry in proteomics: a critical review. *Analytical and bioanalytical chemistry*, 389(4):1017–1031, 2007.

[5] P. C. Carvalho, J. Hewel, V. C. Barbosa, and J. R. Yates III. Identifying differences in protein expression levels by spectral counting and feature selection. *Genetics and molecular research: GMR*, 7(2):342, 2008.

[6] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[7] H. Choi, D. Fermin, and A. I. Nesvizhskii. Significance analysis of spectral count data in label-free shotgun proteomics. *Molecular & cellular proteomics*, 7(12):2373–2385, 2008.

[8] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[9] C. S. Floudas, J. B. Balasubramanian, M. Romkes, and V. Gopalakrishnan. An empirical workflow for genome-wide single nucleotide polymorphism-based predictive modeling. *AMIA Summits on Translational Science Proceedings*, 2013:53, 2013.

[10] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine learning*, 63(1):3–42, 2006.

[11] V. Gopalakrishnan, J. L. Lustgarten, S. Visweswaran, and G. F. Cooper. Bayesian rule learning for biomedical data mining. *Bioinformatics*, 26(5):668–675, 2010.

[12] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[13] Z. He and W. Yu. Stable feature selection for biomarker discovery. *Computational biology and chemistry*, 34(4):215–225, 2010.

[14] K. Jong, E. Marchiori, M. Sebag, and A. Van Der Vaart. Feature selection in proteomic pattern data with support vector machines. In *Computational Intelligence in Bioinformatics and Computational Biology, 2004. CIBCB'04. Proceedings of the 2004 IEEE Symposium on*, pages 41–48. IEEE, 2004.

[15] M. Kawasaki, T. Maeda, K. Hanasawa, I. Ohkubo, and T. Tani. Effect of his-gly-lys motif derived from domain 5 of high molecular weight kininogen on suppression of cancer metastasis both in vitro and in vivo. *Journal of Biological Chemistry*, 278(49):49301–49307, 2003.

[16] Y. Kim and J. Kim. Gradient lasso for feature selection. In *Proceedings of the twenty-first international conference on Machine learning*, page 60. ACM, 2004.

[17] J. Kittler, M. Hatef, R. P. Duin, and J. Matas. On combining classifiers. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 20(3):226–239, 1998.

[18] I. Kononenko. Estimating attributes: analysis and extensions of relief. In *Machine Learning: ECML-94*, pages 171–182. Springer, 1994.

[19] L. I. Kuncheva. Switching between selection and fusion in combining classifiers: An experiment. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 32(2):146–156, 2002.

[20] D. Liu, S. Cai, X. Gu, J. Scafidi, X. Wu, and A. E. Davis. C1 inhibitor prevents endotoxin shock via a direct interaction with lipopolysaccharide. *The Journal of Immunology*, 171(5):2594–2601, 2003.

[21] H. Liu, J. Li, and L. Wong. A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. *Genome Informatics Series*, pages 51–60, 2002.

[22] H. Liu, R. G. Sadygov, and J. R. Yates. A model for random sampling and estimation of relative protein abundance in shotgun proteomics. *Analytical chemistry*, 76(14):4193–4201, 2004.

[23] D. H. Lundgren, S.-I. Hwang, L. Wu, and D. K. Han. Role of spectral counting in quantitative proteomics. *Expert review of proteomics*, 7(1):39–53, 2010.

[24] J. L. Lustgarten. A bayesian rule generation framework for'omic'biomedical data analysis. *Doctoral Dissertation*, 2009.

[25] J. L. Lustgarten, V. Gopalakrishnan, and S. Visweswaran. Measuring stability of feature selection in biomedical datasets. In *AMIA Annual Symposium Proceedings*, volume 2009, page 406. American Medical Informatics Association, 2009.

[26] J. L. Lustgarten, S. Visweswaran, V. Gopalakrishnan, and G. F. Cooper. Application of an efficient bayesian discretization method to biomedical data. *BMC bioinformatics*, 12(1):309, 2011.

[27] Z.-Q. Ma, S. Dasari, M. C. Chambers, M. D. Litton, S. M. Sobecki, L. J. Zimmerman, P. J. Halvey, B. Schilling, P. M. Drake, B. W. Gibson, et al. Idpicker 2.0: Improved protein assembly with high discrimination peptide identification filtering. *Journal of proteome research*, 8(8):3872–3881, 2009.

[28] K. A. Neilson, N. A. Ali, S. Muralidharan, M. Mirzaei, M. Mariani, G. Assadourian, A. Lee, S. C. Van Sluyter, and P. A. Haynes. Less label, more free: Approaches in label-free quantitative mass spectrometry. *Proteomics*, 11(4):535–553, 2011.

[29] A. I. Nesvizhskii and R. Aebersold. Interpretation of shotgun proteomic data the protein inference problem. *Molecular & Cellular Proteomics*, 4(10):1419–1440, 2005.

[30] A. I. Nesvizhskii, O. Vitek, and R. Aebersold. Analysis and validation of proteomic data generated by tandem mass spectrometry. *Nature methods*, 4(10), 2007.

[31] S.-E. Ong and M. Mann. Mass spectrometry–based proteomics turns quantitative. *Nature chemical biology*, 1(5):252–262, 2005.

[32] C. Y. Park, A. A. Klammer, L. KaĬLll, M. J. MacCoss, and W. S. Noble. Rapid and accurate peptide identification from tandem mass spectra. *Journal of proteome research*, 7(7):3022–3027, 2008.

[33] T. V. Pham, S. R. Piersma, M. Warmoes, and C. R. Jimenez. On the beta-binomial model for analysis of spectral count data in label-free tandem mass spectrometry-based proteomics. *Bioinformatics*, 26(3):363–369, 2010.

[34] P. Refaeilzadeh, L. Tang, and H. Liu. Cross-validation. In *Encyclopedia of Database Systems*, pages 532–538. Springer, 2009.

[35] M. Robnik-Šikonja and I. Kononenko. An adaptation of relief for attribute estimation in regression. In *Machine Learning: Proceedings of the Fourteenth International Conference (ICMLâĂŹ97)*, pages 296–304, 1997.

[36] Y. Saeys, I. Inza, and P. Larrañaga. A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517, 2007.

[37] S. C. Shah and A. Kusiak. Data mining and genetic algorithm based gene/snp selection. *Artificial intelligence in medicine*, 31(3):183–196, 2004.

[38] K. L. Simpson, A. D. Whetton, and C. Dive. Quantitative mass spectrometry-based techniques for clinical use: biomarker identification and quantification. *Journal of Chromatography B*, 877(13):1240–1249, 2009.

[39] M. Spivak, J. Weston, D. Tomazela, M. J. MacCoss, and W. S. Noble. Direct maximization of protein identifications from tandem mass spectra. *Molecular & Cellular Proteomics*, 11(2):M111–012161, 2012.

[40] C.-F. Tsai and Y.-C. Hsiao. Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches. *Decision Support Systems*, 50(1):258–269, 2010.

[41] D. Wettschereck and D. W. Aha. Weighting features. In *Case-Based Reasoning Research and Development*, pages 347–358. Springer, 1995.

[42] S.-i. Yamada, M. Ohira, H. Horie, K. Ando, H. Takayasu, Y. Suzuki, S. Sugano, T. Hirata, T. Goto, T. Matsunaga, et al. Expression profiling and differential screening between hepatoblastomas and the corresponding normal livers: identification of high expression of the plk1 oncogene as a poor-prognostic indicator of hepatoblastomas. *Oncogene*, 23(35):5901–5911, 2004.

[43] W. Zhu, J. W. Smith, and C.-M. Huang. Mass spectrometry-based label-free quantitative proteomics. *Journal of Biomedicine and Biotechnology*, 2010, 2009.

Optimization of an individual re-identification modeling process using biometric features

Alejandro Heredia-Langner[1], Brett G. Amidan[1], Shari Matzner[1], and Kristin H. Jarman[1]
[1]Pacific Northwest National Laboratory, 902 Battelle Boulevard, PO Box 999 Richland, Washington 99352 USA

Abstract— *We present results from the optimization of a re-identification process using two sets of biometric data obtained from the Civilian American and European Surface Anthropometry Resource Project (CAESAR) database. The datasets contain real measurements of features for 2378 individuals in a standing (43 features) and seated (16 features) position. A genetic algorithm (GA) was used to search a large combinatorial space where different features are available between the probe (seated) and gallery (standing) datasets. Multiple linear regression models are employed to estimate one set of features from the other. Results show that optimized model predictions obtained using less than half of the 43 gallery features and data from roughly 16% of the individuals available produce better re-identification rates than two other approaches that use all 43 gallery set features and information from all 2378 individuals.*
Key Words: Genetic Algorithm, Re-identification, CAESAR.

1. Introduction

Re-identification is the task of accurately recognizing a person that has been previously observed and for whom some information is available in a database. For example, an image obtained from a photograph or video can be employed to estimate measurements of certain body features or other characteristics, and those estimates used to interrogate a database in search of a match. In particular, the database may contain biometric information obtained using a controlled and systematic process that can be reliably used to identify an individual. Subsequently, the same or other measurements may only be obtainable under a different and more challenging set of circumstances.

Numerous research efforts have been conducted recently on person re-identification, including gait recognition [1], clothing appearance [2], and anthropometry [3], [4]. The work in references [2] and [5] provide two excellent surveys on person re-identification. Other research has also focused on finding anthropometric features for clustering individuals along gender [6] and in reducing the number of dimensions needed for clustering [10]. In this work, we present results for person re-identification using two sets of biometric data. The two sets of data were obtained by the Air Force Research Laboratory (AFRL) and form part of the Civilian American and European Surface Anthropometry Resource Project (CAESAR) database [7]. The datasets used in this work are 1D North American anthropometric measurements for 2384 individuals in a standing position and 2380 individuals in a seated position. The standing dataset contains measurements of 43 body features (measurements between two landmarks) and the seated dataset contains measurements of 16 body features, with values for both sets reported in millimeters. The two datasets have 2378 persons in common. Data for five body features are identified with very similar names in the two sets but, because the measures are obtained in standing or seated positions, the numerical values of the features with similar names are not equal for a given person. Some measurements are missing for some individuals in each of the two datasets. Because the standing dataset contains information for a larger number of body measurements and the largest number of individuals, it is used as the gallery set, the set containing sufficient information for unique identification, and the seated set is used as the probe (or secondary) data from which gallery set feature value estimates will be obtained. The names of the features in the standing set are shown in Table 1 and the names of the features in the seated set are shown in Table 2.

Table 1. Names of the 43 Features in the North American 1D Standing Set (Gallery Set)

Feature Name	Feature Name	Feature Name
Acromial Ht Stand Lt	Bitrochant.Brth Stand	Malleolus Med Rt
Acromial Ht Stand Rt	Bustpoint Brth	Neck Ht
Acromion-Radiale Len Lt	Cervicale Ht	Radiale-Stylion Lt
Acromion-Radiale Len Rt	Chest Ht Stand	Radiale-Stylion Rt
Ankle Ht Lt Malleolus,Lat	Elbow Ht Stand Lt	Sellion Supramenton
Ankle Ht Rt Malleolus,Lat	Elbow Ht Stand Rt	Sleeve Outseam Lt
Arm Inseam Lt	Foot Brth Lt	Sleeve Outseam Rt
Arm Inseam Rt	Foot Brth Rt	Sphyrion Ht Lt
Axilla Ht Lt	Infraorbitale Ht Lt	Sphyrion Ht Rt
Axilla Ht Rt	Infraorbitale Ht Rt	Suprasternale Ht
Biacromial Brth	Inter-pupillary Dst	Trochanterion Ht
Bicristale Brth	Interscye Dst Stand	Trochanterion Ht
Bigonial Brth	Knee Ht Stand Lt	Waist Back
Bispinous Brth	Knee Ht Stand Rt	
Bitragion Brth	Malleolus Med Lt	

Table 2. Names of the 16 Features in the North American 1D Seated Set (Probe Set)

Feature Name
Acromial Ht Sit Lt
Acromial Ht Sit Rt
Bi-lateral Femoral Epicondyle Brth Sit
Bi-lateral Humeral Epicondyle Brth Sit
Bitrochanteric Brth Sit
Buttock to Trochanter Lth
Femoral Epicondyle Lat to Malleolus Lat Lt
Femoral Epicondyle Lat to Malleolus Lat Rt
Infraorbitale Ht Sit Lt
Infroarbitale Ht Sit Rt
Trochanter to Femoral Epicondyle Lat Lt
Trochanter to Femoral Epicondyle Lat Rt
Trochanter to Seated Surface Lt
Trochanter to Seated Surface Rt
Elbow Ht Sit Lt
Elbow Ht Sit Rt

2. Materials and Methods

The process of re-identifying an individual by searching a database is fairly simple. A numerical vector with values for some body features from an unknown individual is compared against the corresponding values in a database. Standardized distances between the vector of the unknown individual and every person in the database are calculated and ranked. Standardized distance metrics are often used in re-identification because body feature measurements vary in magnitude. The individual in the database with the smallest distance, ideally zero, to the vector from the unknown person is reported as the closest match. This matching process creates a single, real-valued metric that determines how similar any two subjects are. If a correct match is found as the top-ranked standardized distance, it is said to have a Rank of 1. If the correct match is found, say, in the fifth position of ranked standardized distances, it is said to have a Rank of 5.

Using standardized Euclidean distance between two vectors of body measures as a metric to identify an individual is reasonable because sets of body measurements for a person tend to be unique. In the absence of noise, very few features in the gallery set are needed to unambiguously identify an individual in the database. Most combinations using only two of the 43 gallery features available provide perfect discrimination under conditions of unchanging measurements and the fixed number of individuals in the database. Not surprisingly, using a larger number of features results in better identification power, measured as better separation between all pairs of distinct individuals in the gallery set. Figure 1 shows distributions of the minimum standardized Euclidean distance among all pairs of individuals in the gallery when one thousand samples with 5, 10, 15, 20, 25, 30, and 35 features from the gallery set are selected at random and used to calculate pairwise standardized Euclidean distances.

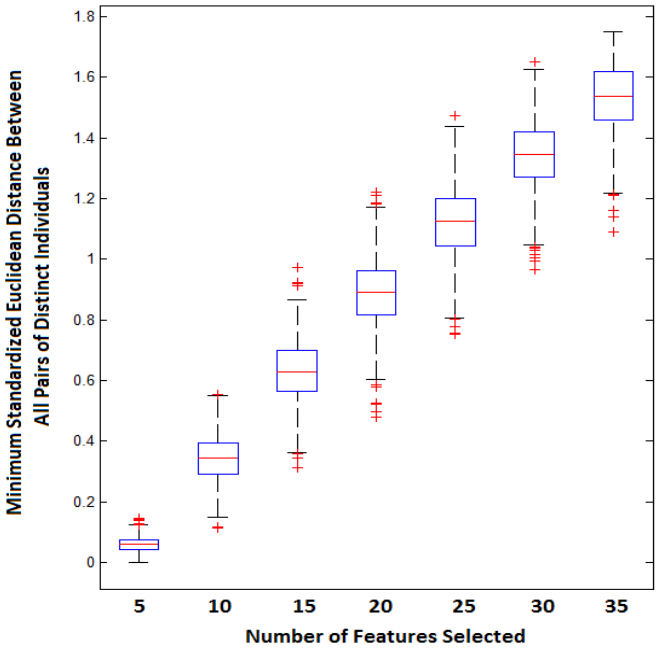


Figure 1. Box plots of the minimum standardized Euclidean distance among all distinct pairs of 2384 individuals in the gallery data for 1000 samples where the number of features shown in the x-axis was selected at random from the 43 available. The central mark in each box is the median and the edges are the 25 and 75 percentiles, the edges extend to the most extreme values not considered outliers and the crosses are outliers.

Figure 1 makes clear that, although some feature combinations provide better separation between all pairs of distinct individuals in the gallery, it is likely that any set with 10 or more features will be sufficient for perfect identification. Naturally, it is a much greater challenge to obtain estimates of gallery set measurements from data collected in a different, possibly uncontrolled way.

The re-identification question using two different sets of data becomes a feature selection problem. As Figure 1 shows, very few features are needed for perfect re-identification if the features in the gallery set are known or can be estimated with very high accuracy. Figure 1 also shows that, in general, using a larger number of features is better. If features in the gallery set can only be estimated from a secondary source of data with some degree of error, how many features are needed and what level of re-identification can we expect to achieve? Is there a subset of features that is better for re-identification purposes?

Because 2378 of the individuals are common to both the gallery and probe datasets, it is easy to study the relationship between pairs of features in the two sets. Simple linear correlation coefficients between every single feature in the gallery set and every single feature in the probe set range between -0.18 and 0.96, with most of the pairings (496/688) having simple linear correlation values under 0.6. This indicates that, with some exceptions, few features in the probe set are good linear predictors, on their own, of features

in the gallery set. Figure 1 shows that a relatively small number of features are sufficient to establish the identity of an individual in the gallery set so, a naïve approach would be to employ individual features in the probe set to predict gallery set values, and using those estimates for identification.

Gallery set feature values were estimated using, for each, the single most highly correlated feature in the probe set as a predictor. This approach is intuitively appealing because only the best possible individual predictor is used, and probe set features containing little or no useful information as predictors of gallery features are ignored to the extent possible. Unfortunately, results from this approach are disappointing, as only 120 individuals end up with a Rank of 5 or better.

The relationship between features in the probe and gallery sets may be more complex than predictions from one simple linear regression model may be able to convey. A more sophisticated approach involves the use of multiple linear regression models to obtain estimates of gallery set features. In this approach, a linear model relating a feature in the gallery set to one or more features in the probe set is built using a training set of randomly selected individuals. These multiple linear regression models are built using a forward stepwise procedure (p-value to enter of 0.05 and 0.1 to remove), one for each feature in the gallery set, and all using information from the same set of individuals. The models can then be used to predict a vector with gallery set feature estimates using probe set data as input.

As we have shown, accurate re-identification can be carried out with a relatively small number of gallery set features. This means that, as long as some predicted gallery measures are available, finding the best possible match with the gallery data is possible. Naturally, the accuracy of the match will depend on the quality and the quantity of the feature estimates. The problem then becomes one of finding an optimum set of features that will result in a maximum identification rate. Other parameters, such as the quality of predictions and the size of the training set used to build the multiple linear regression models, could also affect the correct identification rate.

Searching through this space for an optimal set of parameters in an exhaustive way is not practically feasible. The number of combinations of two or more features in the gallery set is of the order of 8.8×10^{12}. If we consider also the size of the training set used to create the prediction models and the quality of the fits for an estimate to be considered good as two more parameters to be optimized, the size of the problem space becomes even more intractable.

To find a solution, a genetic algorithm (GA) was implemented. Genetic Algorithms are a heuristic optimization technique, loosely based on the Darwinian theory of evolution, in which selective pressure is exerted on an evolving population of solutions (or chromosomes) through mechanisms of recombination, selection and mutation [8],[9]. Repeated application of the GA mechanisms forces improvement in the fitness (or objective function) value of the population until convergence is reached.

The form of a GA solution for the re-identification problem considered here consists of a vector (or chromosome) with 45 entries. The first 43 are binary (1/0) entries indicating whether the corresponding feature in the gallery set will be estimated and used in the matching process or not. The 44^{th} entry in the chromosome is an R^2 threshold, indicating that only multiple linear regression models that match or exceed this threshold with the training set will be used to create a vector of estimated gallery feature values. The last entry in the chromosome is the size of the training set (number of individuals) used to build the multiple linear regression models. To ensure that only multiple linear regression models with at least a moderately good fit were considered, it was decided to limit the search of R^2 threshold values to the [0.5, 1] range. The size of the randomly selected training set was also limited to remain between 100 and 500 individuals. The limits in the size of the training set were imposed to determine if it is possible to build models that produce reasonably good and reliable predictions without having to use all the data available.

The GA creates an initial population of solutions at random called the parent population. A population of offspring solutions is obtained by combining the contents of chromosomes in the parent population. Evaluation of every solution in the offspring population is carried out by choosing a training set of individuals, of the size indicated by the solution, selected at random. A multiple linear regression model for each of the gallery features that have a '1' in a solution is built, employing a stepwise procedure, using data in the training set while all the features in the probe set remain available to build the models. To avoid overfitting, the models are limited to purely linear terms. After the models have been built, data for every individual in the probe set is used to predict values for the appropriate features in the gallery set, provided that the multiple linear regression model for that feature has an R^2 value that is equal or greater than the threshold indicated by the GA solution. The resulting vector of gallery set feature estimates is compared to all the available gallery data by computing standardized Euclidean distances. The standardized Euclidean distances are ranked in ascending order and the position where the correct ID is found is stored. For example, if the top match corresponds to the correct identity, this individual has a Rank of 1. However, if the top four matches for an individual are wrong (the individual is not one of these four persons in the gallery set) and the correct identity is found as the fifth match, this individual has a Rank of 5. The fitness value of the chromosome is obtained by adding the number of individuals with Rank 5 or better.

After the offspring solutions have been evaluated, their fitness values are sorted and the best solutions are selected to become the new parent population. To ensure that all feasible chromosome entries remain available, and to help

avoid premature convergence, a mutation mechanism is applied to the new parent population. Mutation consists of making random changes to a small number of individuals, also selected at random, in the new parent population. The mechanisms of recombination, evaluation, selection and mutation are applied repeatedly until some measure of convergence is achieved. In general, the fitness value of the best solution (or solutions) is used to determine if the GA has converged. When the fitness value of the best solution remains unchanged generation after generation, we say that the algorithm has converged. The re-identification algorithm described here was implemented in MatLab (R2013)[11].

3. Results

Table 3 shows the gallery set features selected by the GA with multiple regression models that match or exceed the R^2 threshold selected by the algorithm. The values of R^2 threshold and training set size selected by the GA are 0.87 and 389 respectively.

Table 3. Identities of the Features in the North American 1D Standing Set Selected by the GA

Feature Name	Feature Name
Acromial Ht Stand Lt	Infraorbitale Ht Lt Stand
Acromial Ht Stand Rt	Infraorbitale Ht Rt Stand
Acromion-Radiale Length Lt	Knee Ht Stand Rt
Acromion-Radiale Length Rt	Sleeve Outseam Len Lt
Axilla Ht Lt	Trochanterion Ht Lt
Bitrochanteric Brth Stand	Trochanterion Ht Rt

Results from Table 3 show that only a subset of less than half of the features in the standing set are selected by the GA as optimal for building an anthropometric signature for the available individuals. Over repeated independently started GA runs, most of the features in Table 3 are selected again, indicating that there is a subset of features that are robust for re-identification purposes under the approach used in this work. Even though it appears as if some of the features in Table 3 are redundant -both the left and right measurements of four features are selected by the algorithm- removing even just one or two features from those shown in Table 3, results in an average decline of about 5% in the number of re-identifications with Rank 5 or better. Removing all four 'right' features in Table 3 and keeping only the 'left' when both are present, results in an average decrease of 16% in the number of re-identifications with Rank 5 or better.

Figure 2 shows the cumulative proportion of individuals plotted against their re-identification rank for feature vectors estimated in three different ways:

1) Ten GA solutions using the features shown in Table 3 and multiple linear regression models built using ten training sets of 389 randomly chosen individuals and an R^2 threshold of 0.87 (solid lines).

2) Estimates for all 43 features in the gallery set, each estimated using a multiple linear regression model where all individuals are used to build the models and all the probe set variables are potentially available (dashed line).

3) The approach described in the first part of this paper, using all 43 features and simple linear regression models built using all individuals. Each feature in the gallery set is estimated using the simple linear model with the best R^2 of those available (dot and dash line).

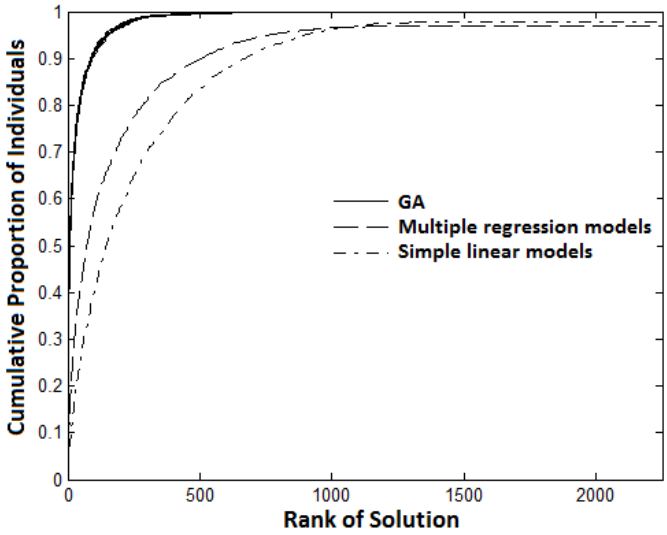


Figure 2. Plots of cumulative proportion of individuals (y-axis) for a given rank (x-axis) for ten solutions using the features found with the GA (solid lines), multiple linear regression models for all 43 gallery set features using information from all 2378 individuals (dashed line) and the best simple linear regression model for each feature in the gallery set (dot and dash line).

Figure 2 shows that the re-identification predictions found using the parameters reported by the GA exhibit better performance than multiple linear regression models obtained using all gallery set features and all individuals and much better than results obtained using the best simple linear regression model for each gallery set feature. This indicates that better quality re-identification results can be obtained by relying on a subset of accurately estimated features and that reliable predictive models for those features can be obtained using data from relatively few individuals.

Notice that, because the identity of the individuals selected to build the multiple linear regression models changes from generation to generation, even if the identity of the gallery set features chosen remains unchanged, the evolving solutions are robust against the particular subset of individuals used to build the models. Only solutions that perform well generation after generation, that is, solutions that maintain a high fitness value with relatively little

variability, will be maintained by the GA. This puts pressure on the algorithm to select gallery set features that can be reliably estimated without over-fitting.

Despite the encouraging results shown in Figure 2, the re-identification task remains challenging. Only an average of about 19% of the 2378 individuals receive a Rank of 1 using the GA solution over repeated runs using randomly selected subsets of 389 individuals to build the predictive models, and an average of 42% receive a Rank of 5 or better. Still, these results indicate that this re-identification approach may prove useful for greatly narrowing down the pool of individuals in a database that require closer inspection.

4. Conclusions and Future Work

We have presented a methodology to develop predictive models for biometric features linking two sets of distinct data involving 2378 individuals. Individuals in the gallery set can be unambiguously identified using only a few biometric measures if these measures are known, or can be estimated, with high accuracy. However, estimation of biometric features in a gallery set using as predictors data gathered under different circumstances presents a number of challenges. Investigating an adequate set of gallery features that can be predicted using features in a probe set is difficult because the combinatorial space is very large. In addition, the predictive models sought should be of enough quality (producing relatively accurate predictions) and should be robust to the particular subset of data used to build them.

A genetic algorithm (GA) was used to explore the problem space, searching for a group of gallery set features that could be linearly related to the features in the probe set. Results indicate that the GA selects less than half of the gallery set features to make a re-identification and that this approach produces better results than two other approaches that use information for all features and all individuals available.

The methodology presented in this paper could prove useful when incorporated into a re-identification system that is constantly updated. Biometric information from new individuals, or information from new biometric features, can be added and the algorithm trained again, helping in the development of a system that is robust and scalable.

In future work, we plan to investigate if re-identification performance can be improved by limiting the search to individuals that fit a profile consistent with an estimated vector of gallery set features. We are also exploring new modeling approaches, including feature transformations and different matching metrics. We are interested in studying the possibility of finding gallery set features that may be exchangeable to help in cases when one of the features selected by the GA is not available, and in determining the robustness of the multiple regression models when applied to a new set of data.

Acknowledgment

The research described in this paper is part of the Signatures Discovery Initiative at Pacific Northwest National Laboratory. It was conducted under the Laboratory Directed Research and Development Program at PNNL, a multiprogram national laboratory operated by Battelle for the U.S. Department of Energy. This research was also made possible by the Air Force Research Laboratory, who supplied the CAESAR data and provided valuable technical input.

References

[1] Han, J., Bhanu, B. (2006). Individual Recognition using Gait Energy Image, IEEE Transactions on Pattern Analysis and Machine Intelligence, 28, 316-322.

[2] Satta, R. (2013). Appearance Descriptors for Person Re-Identification: a Comprehensive Review. arXiv e-print 1307.5748. URL http://arxiv.org/abs/1307.5748

[3] Ober, D.B., Neugebauer, S.P., and Sallee, P.A. (2010). Training and Feature-Reduction Techniques for Human Identification using Anthropometry. Biometrics: Theory Applications and Systems, Fourth IEEE International Conference on Biometrics. September 27-29, 2010, Washington, D.C.

[4] Godil, A., Grother, P., and Ressler, S. (2003). Human Identification from Body Shape. IEEE Fourth International Conference on 3-D Digital Imaging and Modeling. October 6-10, Banff, Canada.

[5] Bedagkar-Gala, A., and Shah, S.K. (2014). A Survey of Approaches and Trends in Person Re-Identification. Image and Vision Computing. Accepted Manuscript.

[6] Fouts, A., Rizki, M., Tamburino, L., Mendoza-Schrock, O. (2011). Evolving Robust Gender Classification Features for CAESAR Data. Proceedings of the IEEE 2010 National Aerospace and Electronics Conference. 20-22 July 2011, Dayton, OH.

[7] CAESAR: Civilian American and European Surface Anthropometry Resource Project. http://store.sae.org/caesar/

[8] Goldberg, D.E. (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading, MA.

[9] Holland, J.H. (1992). Adaptation in Natural and Artificial Systems. MIT Press, Cambridge, MA.

[10] Patrick, J., Clouse, H.S., Mendoza-Schrock, O., and Arnold, G. (2010). A Limited Comparative Study of Dimension Reduction Techniques on CAESAR. Proceedings of the IEEE 2010 National Aerospace and Electronics Conference. 14-16 July, 2010, Fairborn OH.

[11] MatLab. Version 8.1.0.604 (R2013a). The Mathworks Inc., Natick, MA, 2013.

More Reliable Over-sampled Synthetic Data Instances By Using Artificial Neural Networks for a Minority Class

Hyontai Sug[1] and Douglas D. Dankel II
Department of Computer and Information Science and Engineering, University of Florida, Gainesville, Florida, U.S.A.

Abstract - *Over-sampling for a minor class is a plausible strategy for better classification when we have imbalanced target data sets in data mining. To over-sampling for minor classes SMOTE, one of the representative over-sampling methods uses artificially generated instances. But, there is some possibility that the classes of artificially generated instances may belong to different classes. In this paper we showed empirically how we may surmount the problem by resorting to some different and more reliable data mining algorithms like artificial neural network. Experiments with a data set called anneal showed that we may select better artificial instances for more reliable data mining models.*

Keywords: Synthetic over-sampling, decision tree, artificial neural networks.

1 Introduction

Over-sampling can be a good strategy in data mining, when data collectability is limited. The method is especially applied to imbalanced data sets to find more reliable classifiers for minor classes [1]. The accurate classification of these minor classes is more important than major classes, because we are often more interested in these rare cases. Minor classes are classes having a relatively smaller number of instances in the target data sets. There are two kinds of over-sampling method: simple over-sampling and artificially generating minor instances. SMOTE[2] is one of the representative over-sampling method based on artificial instance generation for minor class. Because of its importance several slightly modified methods based on SMOTE have been suggested also [3, 4]. SMOTE generates instances for popular data mining algorithms like decision tree and rule generator, and success was reported in its use. But, incorrect training instances are easy to lead to incorrect classifiers. So, generating correct instances is important.

The true class of the artificially generated instances, however, may be questioned, because they are not real data. One possible solution is to rely on the opinion of a domain expert, but, they may not be always available. A second possibility is to rely on the data themselves. There are many data mining algorithms available, and depending on the particular data mining algorithm used, each data set may have a different performance. For example, decision trees and artificial neural networks may have a different performance for the same data set, because decision tree algorithms are based on greedy search methods, while artificial neural networks are based on repeated and gradual training methods. In other words, decision tree algorithms have more tendency of being satisfied with local optima compared to artificial neural networks. As a result, it is known that decision trees have poorer performance than artificial neural networks in many cases [5, 6]. So, if we can find better data mining algorithms than the original target data mining algorithm of SMOTE for a given data set, we may use them to test the artificially generated instances. In section 2 we discuss our experiment method, and in section 3 conclusions are provided.

2 Method and experiment

2.1 Method

Many data mining algorithms exist. For example, SMOTE tries to generate better decision tree of C4.5 [7], rules of RIPPER [8], and naive Bayes model. It uses artificially generated instances of the minor class. The artificial instances are made based on the K-nearest neighbors algorithm and randomization on continues values. As a result, there is some possibility that the class of artificially generated instances is not correct. Anyway, the performance of the system was checked with a10-fold cross validation, and a success was reported.

On the other hand, we know the fact that depending on the data mining algorithm used, each data set may have a different performance. So, we want to check the class of artificially generated instances by SMOTE using more a accurate classifier, if it is available. In the following experiments, we first check the performance of three different data mining algorithms using data sets generated from SMOTE, and then check the quality of the over-sampled data sets.

.

[1] visiting from div. of com. & info. eng., Dongseo U., Korea.

2.2 Experiment

An experiment was performed using a data set called 'anneal' in the UCI machine learning repository [9]. The data set contains a total 798 instances with 38 attributes. Among the attributes are 6 continuously valued, and 3 integer valued, and 29 nominal attributes. There are 6 classes with the number of instances for each class being 8, 99, 684, 0, 67, and 40 for class 1 ~ class 6 respectively. So class 1 is considered a minor class for the experiment.

Three different data mining algorithms were used for the first experiment: C4.5, RIPPER, and multilayer perceptron(MLP). Table 1 shows the accuracy and true positive rate of each class of the data set for each algorithm. C4.5 and RIPPER were chosen because they showed better performances in the original paper of SMOTE [2]. The weka data mining package was used [10] with default parameters of C4.5 and RIPPER, and training time of 2000 for MLP. The experiment was based on 10-fold cross validation. There are no instances of class 4, so its true positive rate is 0 in the table.

TABEL I

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS

		C4.5	RIPPER	MLP
Accuracy(%)		90.98	94.7661	99.4432
True positive rate	Class 1	0.5	0.625	0.75
	Class 2	0.636	0.788	1
	Class 3	0.958	0.969	0.999
	Class 4	0	0	0
	Class 5	0.925	1	1
	Class 6	0.825	0.95	0.95

From this experiment, we can conclude that MLP is the best overall, and it is also the best classifier for the instances of class 1.

After the first experiment, four different percentages (100%, 200%, 300%, and 400%) of artificial instances of the minor class 1 were made using SMOTE. This generated 8, 16, 24, 32 more artificial instances of class 1. As a result, the number of instances in class 1 changed to the values shown in table 2.

TABLE II

CHANGE IN THE NUMBER OF INSTANCES AS OVER-SAMPLING RATE CHANGES

	Over-sampling rate				
	original	100%	200%	300%	400%
Class 1	8	16	24	32	40
Class 2	99				
Class 3	684				
Class 4	0				
Class 5	67				
Class 6	40				

Tables 3, 4, 5, and 6 show the result of the experiment using artificial data sets in over-sampling rate of 100%, 200%, 300%, and 400% of class 1 respectively.

TABLE III

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET WITH 100% MORE ARTIFICIAL INSTANCES OF CLASS 1

		C4.5	RIPPER	MLP
Accuracy(%)		92.7152	95.3642	97.6821
True positive rate	Class 1	0.625	0.813	0.75
	Class 2	0.626	0.828	1
	Class 3	0.977	0.994	0.994
	Class 4	0	0	0
	Class 5	0.985	1	1
	Class 6	0.85	0.9	0.675

TABLE IV

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET WITH 200% MORE ARTIFICIAL INSTANCES OF CLASS 1

		C4.5	RIPPER	MLP
Accuracy(%)		92.9978	95.6236	97.7204
True positive	Class 1	0.708	0.875	0.958
	Class 2	0.687	0.818	1

rate	Class 3	0.971	0.975	0.996
	Class 4	0	0	0
	Class 5	0.97	1	1
	Class 6	0.9	0.95	0.575

TABLE V

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET WITH 300% MORE ARTIFICIAL INSTANCES OF CLASS 1

		C4.5	RIPPER	MLP
Accuracy(%)		93.6009	95.8785	98.4816
True positive rate	Class 1	0.875	0.938	0.969
	Class 2	0.778	0.828	1
	Class 3	0.961	0.963	1
	Class 4	0	0	0
	Class 5	0.955	1	1
	Class 6	0.925	0.95	0.675

TABLE VI

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET WITH 400% MORE ARTIFICIAL INSTANCES OF CLASS 1

		C4.5	RIPPER	MLP
Accuracy(%)		94.086	94.4086	97.0968
True positive rate	Class 1	0.925	0.925	0.975
	Class 2	0.758	0.778	1
	Class 3	0.971	0.963	1
	Class 4	0	0	0
	Class 5	0.925	1	1
	Class 6	0.925	0.95	0.35

If we compare the results of the different experiments, C4.5 seems to improve, and RIPPER becomes more balanced in its true positive rate for each class. But, MLP comes to have the worse accuracies as the percentage of the artificial instances of class 1. Moreover, as the number of instances in class 1 increases, the increase in the true positive rate of class 1 makes that of class 6 decrease. So, we doubt the quality of the artificial data. To check this we performed two more experiments: the first using all the artificial instances that are true positives with the MLP from the original data set, and the second using all the artificial instances that are false positives with the MLP from the original data set. Table 7 shows the false positive rate for class 1 for each over-sampling rate.

TABLE VII

FALSE POSITIVE RATE OF CLASS 1 OF OVER-SAMPLED DATA SETS WITH RESPECT TO MLP AS OVER-SAMPLING RATE CHANGES

Over-sampling rate(%)	False positive rate(%)
100	63
200	63
300	67
400	72

The second set of experiments was performed with artificial instances that MLP says belong to class 1. The MLP has the highest true positive rate for class 1 as shown in table 1. There are 26 different instances of true positives in all the over-sampled data sets, resulting in the total number of class 1 instances becoming 34. Table 8 shows the result.

TABLE VIII

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET HAVING TRUE POSITIVE ARTIFICIAL INSTANCES OF CLASS 1 BASED ON MLP OF THE ORIGINAL DATA SET

		C4.5	RIPPER	MLP
Accuracy(%)		92.2078	96.1039	99.1342
True positive rate	Class 1	0.912	0.941	0.971
	Class 2	0.586	0.848	0.99
	Class 3	0.966	0.975	0.994
	Class 4	0	0	0
	Class 5	0.97	1	1
	Class 6	0.925	0.95	0.95

The third set of experiment was performed with artificial instances of the false positive instances of class 1 with respect to MLP of original data set, because MLP has the highest true positive rate for class 1 as shown in table 1. There are 41 different instances of false positives in all the over-sampled data sets, resulting in the total number of class 1 instances becoming 49. Table 9 shows the result.

TABLE IX

ACCURACY OF THE THREE DIFFERENT DATA MINING ALGORITHMS FOR THE DATA SET HAVING TRUE POSITIVE ARTIFICIAL INSTANCES OF CLASS 1 BASED ON MLP OF THE ORIGINAL DATA SET

		C4.5	RIPPER	MLP
Accuracy(%)		93.6102	95.6337	97.1246
True positive rate	Class 1	0.959	0.959	0.98
	Class 2	0.727	0.798	1
	Class 3	0.965	0.975	0.999
	Class 4	0	0	0
	Class 5	0.955	1	1
	Class 6	0.9	0.95	0.375

Comparing table 8 and table 9, C4.5 and RIPPER do not show much difference in their true positive rates for each class. But, MLP has worse accuracies in table 8. As the number of instances of class 1 increases, the true positive rate of class 6 decreases.

Comparing table 8 and table 9, C4.5 and RIPPER do not show much difference in their true positive rates for each class. But, MLP has worse accuracies in table 8. As the number of instances of class 1 increases, the true positive rate of class 6 decreases.

3 Conclusions

Over-sampling is a common strategy to cope with the situation of insufficient data especially for minor classes. SOMTE has been considered a good methodology of over-sampling minor classes. But, there is some possibility that the artificially generated instances may belong to different classes, even though they are made based on nearest neighbors. As a result, the trained data mining models may not perform as expected for the unseen cases. In this paper we showed the possibility of surmounting this problem by resorting to different and more reliable data mining algorithms. Experiments for a data set showed a promising result. Further experiments will be performed to validate these results using additional data sets.

4 References

[1] H. He. "Learning from Imbalanced Data," IEEE Trans. Knowledge and Data Engineering, vol. 21, issue 9, pp. 1263-1284, September 2009.

[2] N.V. Chawla, K.W. Dowyer, L. O. Hall, W. P. Kegelmeyer. "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321-357, 2002.

[3] H. Han, W. Wang, B. Mao. "Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning," LNCS, vol. 3644, pp. 878-887, 2005.

[4] D. Zhang, W. Liu, X. Gong, H. Jin. "A Novel Improved SMOTE Resampling Algorithm Based on Fractal," Journal of Computational Information Systems, vol. 7, no. 6, pp. 2204-2211, 2011.

[5] Y. Kim. "Comparison of the decision tree, artificial neural network, and linear regression methods based on the number and types of independent variables and sample size," Expert Systems with Applications: An International Journal, vol. 34, issue 2, pp. 1227-1234, February 2008.

[6] L.O. Hall, X. Liu, K.W. Bowyer, R. Banfield. "Why are neural networks sometimes much more accurate than decision trees: an analysis on bio-informatics problem," in Proc. 2003 IEEE International Conference on Systems, Man and Cybernetics, vol. 3, pp. 2851-2856, 2003.

[7] J.R. Quinlan. "C4.5: Programs for Machine Learning". Morgan Kaufmann Publishers, Inc., 1993.

[8] W.W. Cohen. "Fast Effective Rule Induction," in Proc. 12th International Conf. on Machine Learning, pp. 115–123, 1995.

[9] A. Frank and A. Suncion. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Sciences, 2010.

[10] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten. "The WEKA Data Mining Software: An Update," SIGKDD Explorations, vol. 11, issue 1, 2009.

Entropy Based Adaptive Outlier Detection Technique for Data Streams

Yogita[1], Durga Toshniwal[1], and Bhavani Kumar Eshwar[2]
[1]Department of Computer Science and Engineering, IIT Roorkee, India
[2]IBM India Software Labs, Bangalore, India

Abstract— *Outlier detection in data streams is an immensely enthralling problem in many application areas such as network intrusion detection, faulty sensor detection, fraud detection in online financial transactions etc. Majority of existing outlier detection techniques have been mainly designed for static datasets and require a global view and multiple scans of data which is not feasible in case of streaming data. In this paper, we propose an entropy based outlier detection technique for streaming data exploiting the fact that presence of an anomalous data object highly increases the entropy of normal data clustering. It maintains clusters of streaming data and finds change in its entropy on incoming data object. If increment in entropy is very large then the data object is marked as candidate outlier and its anomalous behaviour confirmed over multiple sliding windows to minimize the false alarms. The proposed method is incremental and dynamically updates clustering structure and entropy statistics to deal with heavy volume and concept evolution of data streams. The proposed scheme has been evaluated on both synthetic and real world data. Experimental results prove its effectiveness on following performance measures: outlier detection rate, false alarm rate and running time.*

Keywords: Concept Evolution; Data Streams; Entropy; Outlier Detection

1. Introduction

Outlier detection identifies such data objects that significantly deviate from rest of data [1], [2]. In many applications, outliers are more interesting than the normal patterns of data for example network intrusion detection, fraud detection, fault diagnosis, finding criminal activities in electronic commerce etc. Data streams are potentially infinite sequence of data objects [3] and are produced by many applications such real-time surveillance, environmental monitoring, medical systems, communication networks, online banking, internet traffic etc. Outlier detection in streaming data faces following challenges [4]:

- Entire data stream cannot be stored for multiple scans because of its terrific volume.
- Outlier detection model needs to be updated with incoming data to handle the dynamic nature of data streams.
- High speed of data streams imposes the limitation of memory space and processing time on outlier mining techniques.

A lot of literature is available for outlier detection in static datasets which can be classified as distance based, density based and clustering based methods [5], [6], [7], [8], [9], [10]. But most of existing techniques do not fit in streaming data environment because of their assumption of availability of whole data in memory for multiple scans. Some work has also been done toward distance and density based outlier detection in data streams [11], [12], [13]. These works involve computation of nearest neighbors and depend upon the choice of different parameter.

The entropy is a powerful mechanism for measurement of information content or uncertainty of a variable [14]. It is also referred as a measure of randomness of a system. Concept of entropy is very intuitive for outlier detection because presence of outliers increases the entropy (randomness) of dataset [15], [16], [17], [18] and this increment can be used to measure the outliersness of an object. Concept of entropy has also been used in a number of literature works for clustering data [19], [20] but in the present work our focus on outlier detection as oppose finding quality clusters.

In this paper, we propose an outlier detection technique for streaming data which uses the concept of entropy to avoid pair wise distance computations and nearest neighbour parameter setting. It makes use of fact that presence of an outlier is likely to increases the entropy of clusters comprising normal data. To exploit this fact, it maintains clusters of the normal data objects and for each incoming new data object finds the change in the entropy of clusters [21]. If increment in entropy is very large then the data object is marked as candidate outlier and their anomalous behaviour confirmed over multiple sliding windows to minimize the false alarms [11], [22]. The proposed method is incremental and dynamically updates clustering structure and entropy statistics to deal with heavy volume and concept evolution in data streams. We have implemented and validated proposed technique on both synthetic and real world datasets [23].

The rest of this paper is organized as follows: Section 2 talks about related work. The proposed outlier detection technique has been presented in section 3. The section 4 focuses on the experimental results and conclusion has been given in section 5.

2. Related Work

Outlier detection has its application in large number of domains that's why it has been a topic of importance for research community. A good survey of it can be find in [1], [2]. Majority of this literature focuses on static dataset. Distance-based outlier detection approaches are presented in [5], [24]. Their definitions of outlier are simple and intuitive but requires user to specify parameter k and d which could be difficult to determine. This idea is further extended in [7], [10], where the outlier factor of each data object is calculated as the sum of distances from its k^{th} nearest neighbors. Density based outlier methods are proposed by Breunig et al. in [6] which captures local outlierness of an object and by Tao et al. in [8], it unifies density-based clustering and outlier detection in [8]. Tony et al. [25] proposed the first isolation method for outlier detection called as Isolation Forest (iForest) which detects outliers purely based on the concept of isolation without using any distance or density measure.

Concepts of distance, density and clustering based outlier detection have also been extended to data streams. Two distance based approaches for finding outliers in data streams are given in [11], [12]. Yogita et al. have proposed an unsupervised approach for identifying outliers evolving data streams by weighting attributes in clustering [26]. Most of these methods have applied sliding window model for dealing with data streams. Concept of entropy has been used a lot for solving clustering and outlier detection problem in static [17], [18], [20] and clustering problem in streaming data [21], [19]. But its use for finding outliers in data streams is very intuitive and needs further exploration. Liu and Lu proposed an entropy-based method to approximate the number of outliers for a spatial data set in [15] by using a function of local contrast and local contrast probability of non-spatial and spatial attributes. An entropy-based approach to discover covert timing channels is proposed by Steven et al. in [27] based on the fact that the creation of a covert timing channel affects the entropy of original process. By using information entropy model to measure the uncertainty in rough sets framework Jiang et al. presented a new definition of outliers known as IE (Information Entropy)-based outliers [16].

3. The Proposed Method

This section presents the proposed adaptive outlier detection scheme. Initially preliminary concepts and notations are introduced and then deviation criterion and proposed method is discussed. In the end of this section time and space complexity of proposed method are analysed.

3.1 Preliminary Concepts and Notations

1) Data Stream: A data stream is an infinite sequence of data objects x_1, x_2,..,x_n,.., arriving at time stamps T_1,T_2,..,T_n,.., Each data object is a multidimensional point with *m* attributes. Data streams are of tremendous volume and flows at very high speed. In this work sliding window model has been used to process streaming data. It stores only a percentage of data in memory at a time. After processing the current window data, only sufficient summaries of data are maintained in memory and detailed data is discarded.

2) Cluster Summary Structure: A cluster Summary structure CS of a cluster C at time *t* is defined as follows:

$$CS = (FT, \delta t) \tag{1}$$

where FT is the frequency table which stores the frequency of each attribute value pair of every attribute in the cluster C. A similar data structure is used in [17] for maintaining the attribute value frequency of complete dataset. It can be updated incrementally on assigning a new data object to the cluster by incrementing the frequency of corresponding attribute value pair of each attribute. δt stores the timestamp of the data object that is least recently added to the cluster.

3) Entropy: It is the measure of information and uncertainty or randomness of a variable [14]. Let x is a random variable and S(x) is the set of values that variable x can take and P(x) represents the probability function of random variable x, then entropy E(x) is defined as given by equation (2).

$$E(x) = -\sum_{x \epsilon S(x)} P(x) log_2(P(x)) \tag{2}$$

The entropy of a multivariate vector X = (x_1,...,x_i,..., x_m) having m attributes and x_i is a discrete random variable, can be calculated as defined by equation (3).

$$E(X) = -\sum_{x1 \epsilon S(x1)} ... \sum_{xi \epsilon S(xi)} ... \sum_{xm \epsilon S(xm)} P(X) log_2(P(X)) \tag{3}$$

where P(x_1,...x_i,..., x_m) is the multivariate probability distribution function.

3.2 Deviation Criterion for Outlier Detection

Given a data stream and clusters of normal data, outlier detection aims to identify such data objects that deviate heavily from the clusters based on a deviation criterion. A deviation criterion is very important factor for outlier detection and measures outlierness of a data object. In this paper, we have proposed an entropy based deviation criterion PCE(X) as defined in equation (4). It gives the percentage change in the entropy of clustering Cl on assigning a data object X to a nearest cluster (nearest cluster means that cluster for which increase in entropy is minimal out of all clusters). As outliers highly increases the entropy of clustering so value of PCE will be large and positive for

them. As oppose to outliers value of PCE for normal data objects, will be either negative or small and positive.

$$PCE(X) = \left(\frac{E(Cl + X) - E(Cl)}{E(C)} \times 100 \right) \quad (4)$$

Where E(Cl) is the entropy of a clustering Cl and E(Cl + X) is the new entropy of clustering Cl on assigning data object X to nearest cluster C. E(Cl) is defined by the equation (5) and it represents the weighted sum of entropies of all the clusters [21].

$$E(Cl) = \sum_{k} \frac{|Ck|}{|D|} (E(Ck) \quad (5)$$

To simplify the computation of entropy of a cluster in streaming environment, we have assumed the independence of the attributes of the data stream. This assumption transforms the equation (3) into equation (6) and entropy of a cluster Ck can be calculated using it where xi is a data object belongs to cluster Ck.

$$E(Ck) = E(x1) + E(x2)... + E(xi)... + E(xr) \quad (6)$$

PCE is based upon the fact that when a data object is assigned to a cluster, there may be increase or decrease in the entropy of cluster and respectively in the total entropy of the clustering. Intuitively, if data object is inherently similar to cluster then entropy (randomness) of cluster will either decrease or it will increase slightly, while on assigning a dissimilar (outlying) one it will increase very high. So the percentage change in entropy of clustering (PCE) is a justified criterion for differentiating between outliers and normal data objects.

3.3 Proposed Method in Detail

The pictorial representation of proposed technique is given in Fig. 1. It comprises mainly four modules that are detailed below. Data streams are of tremendous volume and flows at very high speed that's why cannot be stored in memory for processing. We have used sliding window model to process streaming data. It stores only a percentage of data in memory at a time for processing. After processing the current window data, only sufficient summaries of data are maintained in memory and detailed data is discarded to vacate space for next window data.

1) Initialization: In proposed outlier detection technique, normal behaviour of data is represented by clusters. So this module initializes the normal behaviour (clusters) by performing clustering on sampled normal data (It should not contain outlying objects).These clusters are outputted to candidate outlier detection module only once, when processing of streaming data starts. For this initial clustering any clustering algorithm can be freely chosen.

2) Candidate Outlier Detection: Candidate outlier is an object which satisfies deviation criterion. We have used an entropy based deviation criterion PCE defined in equation (4). Algorithm for candidate outlier detection is shown in Algorithm1. It assigns incoming data objects to clusters following the approach given in [21].

Algorithm 1 Candidate Outlier Detection

Input: Clustering (Cl) - Comprises k Clusters, Current Window Data, Threshold
Output: Candidate Outliers

1: Repeat for all data objects of current window
2: X = Read next data object in window
3: Temporally assign X to cluster Ci such that on assigning X increase in entropy of Ci is minimum out of all other clusters Cj where j = 1....j....K.
4: Calculate the PCE(X)
5: **if** PCE(X) $>$ Threshold **then**
6: X is candidate outlier, initialize counter of X to one and save both X and counter into candidate outlier repository to further verify its deviation in outlier detection module
7: **else**
8: X is normal data object output it to updating module to update cluster Ci
9: **end if**
10: End Repeat

In candidate outlier detection algorithm, all objects are classified either as candidate outlier or normal data object. To keep a count of number of windows an object found as candidate outlier, a counter is associated with it and counter is set to one initially. Candidate outliers along with counter are stored in candidate repository. Later on outlier detection module takes candidate outliers from repository to verify their outlying nature (deviation) over multiple data windows. This multiple times verification is done because a candidate outlier may be part of an emerging cluster and showing deviation temporarily instead of being actually an outlier [11], [22]. To consider the local as well as global characteristics of data in outlier detection we have used both increments in individual cluster entropy in step 3 and PCE (percentage change in entropy of clustering) in step 5 of algorithm 1 respectively as criterion to decide upon outlying nature (deviation) of data object.

3) Updating: Concept evolution occurs when new classes come to existence and old may extinct from streaming data over time. It crop ups due to the change in the underlying process which is producing streams. In proposed method, smooth evolution of concepts has been addressed in following way:

- Proposed method is made adaptive by dynamically updating the data clustering and entropy statistics with incoming data streams. This is explained in this section itself.
- Outlying nature of an object is verified over multiple

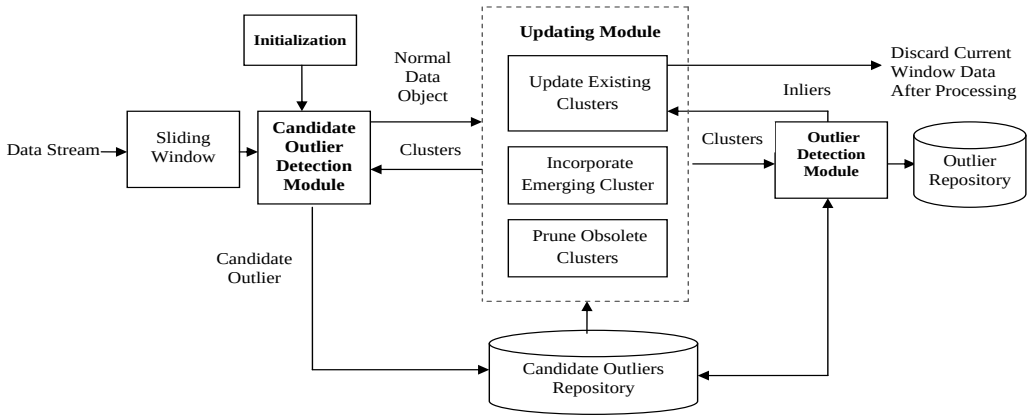


Figure 1: Proposed Adaptive Outlier Detection Technique

data windows before declaring it as outlier because it may be part of an emerging cluster and showing deviation temporarily instead of being actually an outlier. This is the part of outlier detection module.

There are following three types of possible updates to clustering structure and entropy statistics:

1) Update Existing Clusters: Whenever normal data object from candidate outlier module and inliers from outlier module inputted to update module then following are the steps of update procedure:

- Assign object X to cluster C_i such that on assigning X increase in entropy of C_i is minimum out of all other clusters C_j where j = 1....j....K.
- Update cluster summary structure of cluster C_i.
- Calculate the new entropy of clustering by using equation (5)
- Discard the data object X

2) Incorporate Emerging Clusters: Candidate outliers from candidate repository are clustered periodically. If size of any clusters is large enough and its entropy value is smaller than a threshold, it means that cluster is representing a new class in incoming data and hence must be incorporated in clustering as clustering based outlier detection approach assume that outliers are small in numbers and occur in sparse spaces. So following actions are taken

- K = k +1, where k is the number of clusters
- Initialize cluster summary structure of cluster C_k +1.
- Calculate the new entropy of clustering by using equation (5)

3) Prune Obsolete Clusters: If no data object is assigned to a cluster C_i from a long time, it signifies that cluster C_i no more exists in data streams. A times factor δt is associated with each cluster that stores timestamp of data object that is least recently added to the cluster. If difference between current timestamp and δt is greater than a threshold then cluster C_i is deleted. And following actions are taken

- Delete cluster summary structure of cluster C_i.
- K = k -1, where k is the number of clusters
- Calculate the new entropy of clustering by using equation (5)

Cluster summary structure for a cluster comprises FT and δt. FT is the frequency table which stores the frequency of each attribute value pair of every attribute in the cluster C. It can be updated incrementally on assigning a new data object to the cluster by incrementing the frequency of corresponding attribute value pair of each attribute. δt stores the timestamp of the data object that is least recently added to the cluster.

4) Outlier Detection: A candidate outlier turns to a real outlier if it fulfils the deviation criterion continuously over w sliding windows [11], [22]. We have used an entropy based deviation Criterion PCE that is defined in equation (4). Outlying nature of a candidate outlier is confirmed over multiple data windows before flagging it as real outlier because it may be part of an emerging cluster and showing deviation temporarily instead of being real outlier. To keep a count of number of windows over which an object found as candidate outlier, a counter is associated with each candidate and it is set to one initially when candidate outlier is detected first time in candidate outlier detection module. Algorithm for Outlier Detection Module is given in Algorithm 2.

If a candidate outlier found to be an inlier in outlier detection module then it is output to updating module. Because an inlier represents the normal behaviour so it must be incorporated in clustering of data stream.

3.4 Time Complexity

Let *k* is the maximum number of clusters that can occur at a time in data stream, *d* is the dimensions of data, *n* is the window size and c is the maximum number of candidate outlier may occur at a time and D is data stream size. So there

Algorithm 2 Outlier Detection

Input: Clustering (Cl) - Comprises k Clusters, Candidate Outliers, Threshold
Output: Real Outliers

1: Repeat for all candidate outliers
2: Read next candidate outlier from repository
3: Calculate the PCE(candidate outlier)
4: **if** PCE(candidate outlier) > Threshold & Counter(candidate outlier) == W **then**
5: Declare candidate outlier as real outlier, remove from candidate outlier repository and save to outlier repository
6: **else**
7: **if** PCE(candidate outlier) > Threshold & Counter(candidate outlier)< W **then**
8: Update Counter(candidate outlier) = Counter(candidate outlier) + 1
9: **else**
10: Candidate outlier is an inlier (normal data object), Remove it from candidate repository and output to updating module
11: **end if**
12: **end if**
13: End Repeat

will be total D/n data windows for processing. For processing of each window there are three modules. Initialization module works only once and on a small sampled data so its running time can be considered constant I in analysis. Time taken by candidate outlier detection module will be k*n*(1 + d). Updating will take k*d*(n-c) +d*p*(k+1+c*kc*i) and in outlier detection c*k*(1+d) time will be elapsed. Here *i* is the iteration in clustering of candidate outliers, k_c is the number of clusters in candidate outliers set and p is the number windows after which candidate outliers will be clustered. As we have I, d, p and c constant and i and kc will be very small because c is constant. So total running time of proposed scheme can be stated as follows:

$$Running Time = (D/n) \times k \times (n+1) \qquad (7)$$

As k is the number of clusters and it will not increase corresponding to data stream size so time complexity of proposed outlier detection technique will be of order O(D) which show that time complexity is of linear order with data stream size.

3.5 Space Complexity

Memory space used in storing a variable dependents upon the operating system specification, let's for the present work consider windows vista operating system and let k is the number of clusters that can occur at a time in data stream, d is the dimensions of data, m is the maximum number of domain values for a attribute and c is the maximum number of candidate outlier may occur at a time. Proposed method requires to store only following information

- Summary structure of cluster
- Candidate outliers
- Current window data

Space required to store summary structure of k clusters is 4*k*d*m bytes, current window data will take 4*d*n bytes and for candidate outliers 4*c*(d + 1) bytes will be needed. Hence Total memory space used =4*k*d*m bytes + 4*d*n bytes + 4*c*(d + 1). We have d and m constant so total space used will be of order O(k + n + c). Size of window n, number of clusters k and number of candidate outliers are very small as compare to size of data stream so space complexity of proposed method will be of order O(C) where C is a small constant. In this section, it is concluded that proposed method is efficient in terms of space complexity as it is required for data streams.

4. Experimental Results

We have done implementations in matlab R2010a and experiments are conducted on synthetic as well as real data sets. Real data sets are taken from UCI machine learning repository [23]. Threshold values are set by conducting experiments on a subset of data.

4.1 Data Sets

We have worked on two real datasets and two synthetic datasets. For experimental purpose a time stamp is added to each record in all datasets that specify the order of processing of streaming data. 10% samples of each dataset are used in initialization phase and rest are processed over sliding data window.

1) KDDCUP'99 Data Set: This data set was first time used in ACM KDD CUP Challenge of year 1999. After that it has been highly referenced for verification of outlier detection techniques. It is a computer network intrusion detection dataset. Each record represents a network connection which was simulated in a military network environment and labelled as either normal or an intrusion. It consists of 22 simulated attacks of following categories: DOS, R2L, U2R, and PROBE. We have removed class labels for experiments. It consists of total 41 attributes out of which are 7 categorical and 34 numeric attributes and approx. 4,898,431 connection records. In this original form, dataset is not suitable for outlier detection because the percentage of attacks is unrealistically higher than normal records. So we have sampled 10% subset of original data that consist of 489843 records. In sampled dataset attack records are 4895 that are approximately 1% of sampled data set. Numeric attributes are discretized to categorical attributes using equal width binning.

2) Mushroom Data Sets: It contains 8,124 data records of 23 species of mushrooms over 22 categorical attributes.

Table 1: Performance of proposed method in terms of outlier detection rate and false alarm rate

Dataset	Total Outliers	Total Normal Objects	No. of Outliers Detected	No. of False Alarms	Outlier Detection Rate	False Alarm Rate
KDDCUP'99	4895	484948	4187	2354	0.855	0.0049
Mushroom	164	8124	152	99	0.927	0.0122
DS1	500	15000	473	106	0.966	0.0071
DS2	1000	100000	897	113	0.947	0.0011

There are two classes of mushrooms: poisonous (48.2%) and edible (51.8%). We have planted 2% outliers in data sets based upon the frequency of each domain value of attributes.

3) Synthetic Data Set: Synthetic datasets are very useful for performance analysis as it is easy to control data parameters. We have generated two synthetic datasets using GAClust [28] data generator. It is freely available online. First dataset is named as DS1. It consists of 15000 records, 5 categorical attributes and 5 clusters. One cluster contains only 500 records and it is considered as outlier class in our experiments. Second synthetic data set is named as DS2, it comprises 100000 data objects, 5 attributes, 5 clusters and 1% randomly generated outliers.

4.2 Metrics for performance Evaluation

To assess the performance of proposed method along with running time we have also examined following two other metrics: outlier detection rate and false alarm rate. Outlier detection rate refers to the ratio of numbers of actual outliers detected to the total number of outliers in data (refer eq. (8)). False alarm rate is the ratio of numbers of normal data objects that are mistakenly flagged as outlier to the total number of normal data objects (refer eq. (9)).

$$\text{Outlier Detection Rate} = \frac{\text{Number of Outlier Detected}}{\text{Total Outlier}} \quad (8)$$

$$\text{False Alarm Rate} = \frac{\text{Number of False Alarms}}{\text{Total Normal Data Objects}} \quad (9)$$

4.3 Performance Evaluation

1)Outlier Detection Rate & False Alarm Rate

It can be analysed from Table1 that outlier detection rate of proposed technique vary from 0.855 to 0.966 on different datasets which shows its effectiveness. It has been resulted due the incorporation of both local as well as global characteristics of data in outlier detection by using both individual cluster entropy and PCE (percentage change in entropy of clustering) to decide upon outlying nature (deviation) of data object.

False alarm rate of an outlier detection method must be as low as possible because dealing with false alarms require extra effort and expensive for user and system. False alarm rate of proposed method on all four datasets is given in Table 1 and its value is small too as should be. The proposed method has checked an object exceptional behavior over multiple windows before declaring it as an outlier which leads to lower false alarms.

2)Effect of Dataset Size on Running Time

Data streams processing methods must be efficient in terms of running time to meet the challenge of high speed and tremendous volume of streaming data. To validate the efficiency of proposed technique in terms of running time, experiments are done by increasing size of dataset DS2 and KDD CUP dataset (having very large size) and results are shown in Fig 2.

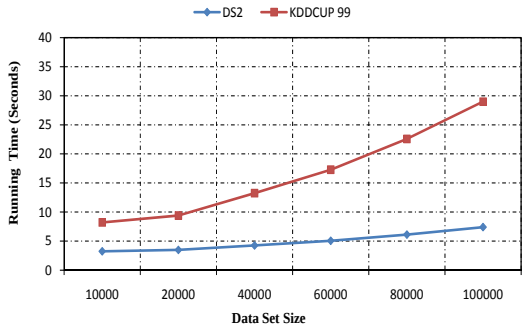


Figure 2: Effect of Dataset Size on Running Time

It can be concluded from Fig. 2 that running time increases linearly with dataset size. It shows the scalability of proposed method. It is achieved by processing streaming data in sliding window (which require only single scan of data) model and incrementally. Difference between running time of two dataset is the result of their different number of dimensions.

3) Increasing Percentage of Outliers verses Detection Rate

In this experiment, a subset of size 5000 of data set DS1 has been used. Outliers are placed in increasing percentages in the data. These outliers are of following two types: group outliers and point outliers. The percentages of outliers correspond to initial data set size.

It can be seen from Fig.3 that outlier detection rate of proposed method is regular with increasing percentages of outliers up to a level because it considers small clusters as outliers groups in place of normal clusters following the criterion of [9]. There is a rapid fall in detection rate after

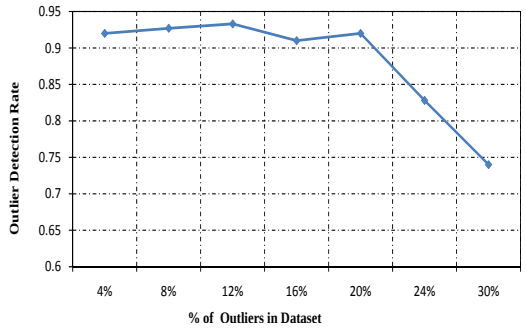


Figure 3: Effect of Increasing Percentage of Outliers on Outlier Detection Rate

20% outliers and this fall is obvious too as these objects do not have anomalous behaviour any more as per clustering based outlier detection approach which assume that outliers are only a small fraction of whole data. But for this analysis, we have still considered them as anomalous.

5. Conclusion

In this work, we have proposed an outlier detection method for data streams using the concept of entropy. It is made incremental and adaptive to handle dynamic nature of data streams. The proposed technique has been validated on both synthetic and real world datasets. Experimental results prove its effectiveness on outlier detection rate, false alarm rate and running time performance metrics. It also uses memory space efficiently as its space complexity is of order O(C). It can be applied in a large number of fields such as banking databases, medical databases, network intrusion detection and weather prediction.

In future, we will compare the proposed techniques to other exiting techniques and analyze the effect of threshold value of deviation criterion on detection rate. An extension of this work for mixed dataset is in progress.

References

[1] V. Hodge and J. Austin, "A survey of outlier detection methodologies," *Artif. Intell. Rev.*, vol. 22, no. 2, pp. 85–126, Oct. 2004.

[2] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, July 2009.

[3] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, J. Kacprzyk and L. C. Jain, Eds. Morgan Kaufmann, 2006, vol. 54, no. Second Edition.

[4] C. Aggarwal, Ed., *Data Streams – Models and Algorithms*. Springer, 2007.

[5] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '00. New York, NY, USA: ACM, 2000, pp. 427–438.

[6] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '00. New York, NY, USA: ACM, 2000, pp. 93–104.

[7] F. Angiulli and C. Pizzuti, "Fast outlier detection in high dimensional spaces," in *Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery*, ser. PKDD '02. London, UK, UK: Springer-Verlag, 2002, pp. 15–26.

[8] Y. Tao and D. Pi, "Unifying density-based clustering and outlier detection," in *Proceedings of the 2009 Second International Workshop on Knowledge Discovery and Data Mining*, ser. WKDD '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 644–647.

[9] Z. He, X. Xu, and S. Deng, "Discovering cluster based local outliers," *Pattern Recognition Letters*, vol. 2003, pp. 9–10, 2003.

[10] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-based detection and prediction of outliers," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 2, pp. 145–160, 2006.

[11] F. Angiulli and F. Fassetti, "Detecting distance-based outliers in streams of data," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, ser. CIKM '07, New York, NY, USA, 2007, pp. 811–820.

[12] M. S. Sadik and L. Gruenwald, *DBOD-DS : Distance Based Outlier Detection for Data Streams*. Springer, 2011, vol. 6261, p. 122Ű136.

[13] D. Pokrajac, A. Lazarevic, and L. J. Latecki, "Incremental local outlier detection for data streams," in *CIDM*, 2007, pp. 504–515.

[14] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July, October 1948.

[15] X. Liu, C.-T. Lu, and F. Chen, "An entropy-based method for assessing the number of spatial outliers," in *IRI*, 2008, pp. 244–249.

[16] F. Jiang, Y. Sui, and C. Cao, "An information entropy-based approach to outlier detection in rough sets," *Expert Syst. Appl.*, vol. 37, no. 9, pp. 6338–6344, Sept. 2010.

[17] A. Koufakou, E. G. Ortiz, M. Georgiopoulos, G. C. Anagnostopoulos, and K. M. Reynolds, "A scalable and efficient outlier detection strategy for categorical data." in *ICTAI (2)*. IEEE Computer Society, 2007, pp. 210–217.

[18] Z. He, S. Deng, and X. Xu, "An optimization model for outlier detection in categorical data," in *Advances in Intelligent Computing*, ser. Lecture Notes in Computer Science, 2005, vol. 3644, pp. 400–409.

[19] S. Wang, Y. Fan, C. Zhang, H. Xu, X. Hao, and Y. Hu, "Entropy based clustering of data streams with mixed numeric and categorical values." in *ACIS-ICIS*. IEEE Computer Society, 2008, pp. 140–145.

[20] P. Andritsos, P. Tsaparas, R. J. Miller, and K. C. Sevcik, "LIMBO: Scalable Clustering of Categorical Data," in *Adv. Database Technol. - EDBT 2004*, 2004, pp. 531–532.

[21] D. Barbará, Y. Li, and J. Couto, "Coolcat: An entropy-based algorithm for categorical clustering," in *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, ser. CIKM '02, New York, NY, USA, 2002, pp. 582–589.

[22] M. Elahi, K. Li, W. Nisar, X. Lv, and H. Wang, "Efficient clustering-based outlier detection algorithm for dynamic data stream," in *Proceedings of the 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery - Volume 05*, ser. FSKD '08. Washington, DC, USA: IEEE Computer Society, 2008, pp. 298–304.

[23] A. Frank and A. Asuncion, "UCI machine learning repository," 2010. [Online]. Available: http://archive.ics.uci.edu/ml

[24] E. M. Knorr and R. T. Ng, "Algorithms for mining distance-based outliers in large datasets," in *Proceedings of the 24rd International Conference on Very Large Data Bases*, ser. VLDB '98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 392–403.

[25] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation-based anomaly detection," *ACM Trans. Knowl. Discov. Data*, vol. 6, no. 1, pp. 3:1–3:39, Mar. 2012.

[26] Yogita and D. Toshniwal, "A framework for outlier detection in evolving data streams by weighting attributes in clustering," in *Proceedings of the 2nd International Conference on Communication Computing and Security*, India, 2012.

[27] S. Gianvecchio and H. Wang, "An entropy-based approach to detecting covert timing channels," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 6, pp. 785–797, 2011.

[28] D. Cristofor and D. Simovici, "Finding median partitions using information-theoretical-based genetic algorithms," *Journal of Universal Computer Science*, vol. 8, pp. 153–172.

SESSION

WEB, TEXT, MULTIMEDIA MINING

Chair(s)

Drs. Robert Stahlbock
Peter Geczy
Gary M. Weiss

Sub-Net Identification for Text Refinement and Text Meaning Discovery in Social Media Analytics

Dinesh Batra, Gurpreet Bawa, Vivek Anand
Analytics, Annik Technology Services Pvt. Ltd., Gurgaon, Haryana, India

Abstract - *The distribution, viewing and exchange of text documents can be seen and modeled as a conversation. Traditionally, the "conversation" has been in one direction (from publisher to consumer); recently, especially with the advent of social media (such as Twitter, Facebook and Blogger.com), the conversation incorporates many more, multi-directional linkages. This development has led to an increased use of such forms of analysis as social network analysis. This development is so pervasive that one of the foundational technologies of Google, the Pagerank algorithm, is a form of social network analysis that was virtually unknown as little as 10 years ago.*

The Pagerank algorithm is one example of many recent advances in computer classification that have made it possible to index and search for document content across very large collections of unstructured data. The method that is proposed here is designed to take advantage of the social context of text-mediated conversations and to extend these analytics in the service of superior text content characterization. Like the Pagerank algorithm, this method makes significant use of social network analysis; in addition, many social characteristics of text conversations are exploited in ways that is completely missing in most forms of current textual content discovery methods.

1 Introduction

Unsupervised (machine automated) sense discovery in unstructured (text) documents currently relies on three methods:

1) In all cases, text is first parsed using a variety of natural language parsing techniques

2) Word meaning is derived by referring to dictionaries (and more recently meaning-based ontology's) or

3) Word groups are identified by a variety of statistical techniques which attempt to pull together words in a similar context so that, when they appear together, they will be easily (and reliably) understood

While our method employs the three methods described above, it specifically provides an extension to the third method; i.e. it provides superior unsupervised meaning discovery through the identification of "sub-net conversations". We employ the conversational properties of text documents to calculate the associated properties of the conversational network – and identification of local sub-nets. This extension leads to a significant expansion of text sense and meaning discovery. For example, in preliminary tests we were able to identify an additional set of term tokens that were associated with a discovered topic in text. *This preface experiment, this additional vocabulary enabled us to increase the classification rate for the given topic by 14%.*

No other technique updates the first pass of discovered vocabulary from the top level corpus with vocabulary that is mined from population sub-segments (sub-nets) as we do. Because of this these general purpose techniques miss many "fine grain" variations in expression (as well as neologisms) that we automatically include.

Our method views text documents as network-based, social conversations: in this framework all textual documents have potentially multi-sided, asynchronous sharing (publishing and consuming) mechanisms (this even applies to traditional reference documents). In particular, social media exchanges – including the exchange of text messages and blogs -- are viewed from the perspective of this message exchange framework.

The novelty of the method proposed here lies in its explicit rigorous definition of the individual, social and operational mechanisms that are employed in order to effectively extract meaningful document sense descriptions in various conversational contexts:

1) The social, individual and community characteristics of the conversation are defined.

2) The message exchange framework – i.e. the conversation – is defined in terms of its social network, associated social network roles (e.g. leader, followers) and associated determination of community membership (node centralities).

3) Where possible, the social attributes of the conversation participants such as age and gender are identified. In addition, the community type of the conversation is characterized; for example, does the community have a personal, professional, political or leisure orientation?

4) Because new terms are manufactured constantly, and because existing words are used in new ways – with different senses – constantly, a comprehensive collection of root forms, synonyms, and elaborated forms of a given topic are collected, manipulated and analyzed (we call these forms "hypernyms").

5) The combination of specific contexts – described in 1 – 3, above, and associated "hypernyms" – described in 4, above, gives rise to *predictive structures* which describe

contextual effects that extend the generality and specificity of the expressions that are normally formed using current unsupervised text discovery methods (as described in third bullet of this section, above).

The combination of terms in the predictive structures track the "memetic" variations of word meanings and *can therefore be used to map different senses, in different contexts, at different times, to the same general category of discussion.* This capability prevents the proliferation of seemingly new classes of discussion and enables the effective comparison of topics of conversation across different groups which, though using different vocabularies, are actually discussing the same thing.

2 Text Documents as "Conversations"

Our method is based on the conversational nature of text documents. Even in the traditional text document publishing model there is an implied conversation between the publisher and consumer; more recently, with electronic transmission, document sourcing, distributing and sharing is much more multi-sided and multi-directional as in many typical conversations. Therefore, all text documents are fundamentally conversational in nature. The recognition of text documents as conversation -- and the associated inclusion of conversational analytics in text document analysis -- is unusual but, as claimed here, is extremely useful since the analysis and interpretation of the textual content and meaning can be informed by the social context that surrounds the conversation. *Most text analytical environments deal with text structure, syntax and semantics, hence do not include the additional information provided by social context. Our method corrects this deficiency and thus provides more information in the analysis and interpretation of text then would otherwise be available*.

3 Sub-Nets as a Primary Unit of Analysis

The proposed method places the discovery of meaning and interpretation of text corpora in the context of a social network, where the information exchange that is reflected in text documents is viewed as a (potentially) multi-sided "conversation" and where this "conversation" takes place among social network members who are participants in a local (social network) sub-net that has specific characteristics that define the nature of the sub-net community and the associated behavioral attributes of the "conversation".

These sub-nets have formal definitional characteristics that are calculated using a variety of social analytical, semantic, participant and conversation identification techniques that are, in turn, used in the derivation of the sub-net definition. The approach described here proposes to determine the context-dependent sub-nets of the social network that suggest word sense variations as well as new words that either change or refine the meaning of the general topic of conversation ("memetic variants").

The social context and conversation behavior of text document exchange are not usually computed in common text meaning discovery systems: one of the most powerful and widely employed methods for text topic discovery – Latent Dirichlet Allocation – employs no concept of social network textual meaning derivation whatsoever. Typical techniques are therefore more "brittle" than the sub-net approaches; i.e. have fewer expressions associated with a given meaning so cannot as effectively track meaning over time and ignore both subtle and gross changes in the expression of a given meaning (in various sub-groups) over time

4 Hypernym Calculation and Memetic Variations

Term combinations – called "hypernyms" – which tend to indicate memetic sense refinements or extensions that are context-dependent are identified using our method. "Hypernyms", as used in our method, have two distinguishing characteristics:

1) The context for the identification of a hypernym is defined by the specific sub-net that it occurs in. While the normal synonym is universal in its application – for example, the use of "big" as a synonym for "large" may apply without restriction – hypernyms are specifically defined in a particular sub-net context. For example, we may say that "big" and "large" are synonyms for the "dimension topic" as identified among sub-net participants who discussed "dress size" in today's email entries.

2) A further distinguishing characteristic is that hypernyms are formed through the decomposition of term aggregates in a specific hierarchical fashion. Hypernyms are identified at the highest through lowest levels of aggregations in the following order: class (or category), topic, noun phrase, named entity, part of speech, bigram, raw term.

This granular decomposition characteristic is illustrated in the figure below:

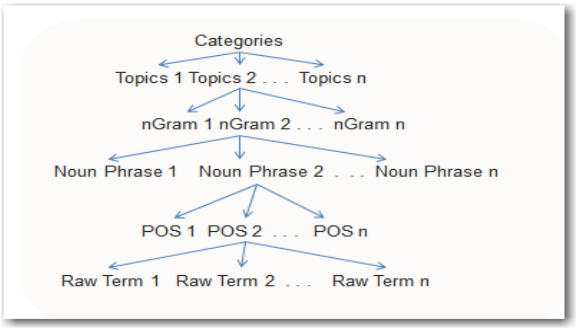


Figure 1: Hierarchy of Term Components Used to Identify Hypernyms (Ordered by Granularity)

The net effect of the hypernym calculation method is to form a rich set of potential markers of a given topic or concept so as to detect variations in term usage and meaning – at various levels of granularities – in a way that is not normally found and would not be otherwise possible.

5 Physical and virtual threads

Both Physical and virtual sub-net conversation "threads" can be identified. For example, physical threads in an email exchange are identified by examining the "from" and "to" fields in the email, the "subject" and any associated "cc" fields, for example. Similar methods are used to identify threads in blog conversations and other social media mechanisms such as forums and user groups. Determining the existence of a virtual thread is more abstract: virtual sub-nets may be identified by grouping documents according to semantic similarities, common topics of conversation, or common date-time or geographic ties (indicating a "virtual" link even if no documents have been explicitly exchanged).

Just as the detection of social networks is never used to extract text meaning or interpretation in typical text analytical tools, neither is the computation of virtual or physical links.

6 Social Network Analytics

Physical and virtual threads are encoded as links (or edges) between nodes (or vertices); hence the associated conversations may be analyzed by a range of social network analytic techniques. Of particular importance is the calculation of the sub-nets that characterize a given collection of text documents. Number of other metrics that are used in sub-net and individual node characterization, for example: Centrality, Betweeness, Influence

Since these metrics are produced on a standard scale they can be summed and averaged across various sub-nets (and hence can be used to compare sub-net attributes).

The community detection of sub-nets in the collection of text documents that are conceived of as the "host" for the various conversations that are monitored in our approach is a primary feature and capability of this conversational approach to text analytics. It is axiomatic that vocabulary and term usage is socially determined since language is a tool for social communication. While the word "snow" might suffice in a general conversation, it is likely that a group of skiers will use such terms as "corn", "sugar", "hard pack", "boilerplate" and so on to describe the many varieties of snow (yet the generic term "snow" may be completely missing from the conversation).

This core capability of social context analytics is missing from the major forms of natural language processing & text topic derivation that are in use today. The ability to situate a conversation in a social context goes a long way towards the precise allocation of meaning to phrases like "bank", for example, that could be referring to a "river bank" or a "savings bank". While it is normal for most text analytic systems to explore the associated vocabulary of the embedded term – to find associated instances of "water" or "financial institutions", it is unknown to characterize the sub-net as "water resource related" or "financial institution related", for example. This example is meant to illustrate the differences between our method and the normal procedure and will suggest how our method is at least different, and possibly superior.

7 Sub-Net Characteristics Refinement

Regardless of how the top-level sub-net is derived (as this may depend on the procedure settings and whether physical or virtual links have been defined) a number of other criteria are used to further specify the sub-net attributes, the characteristics of the individual participants and, if necessary, these may be used to further sub-set the sub-net into smaller collections. These refinements are used to create sub-nets that conform to "Dunbar's Number"; i.e. the size ranges from 100 – 250 participants. (Dunbar's number is considered an upper limit on the effective range of meaningful social interactions). Sub-net characteristics:

1) Individual indicators: Age, gender.
2) Social (sub-net) indicators: Influence, recreational interest.
3) Psychological indicators: Mood state, personality type (e.g. introversion vs. extroversion).
4) Behavioral indicators: Message recency, frequency, acceleration, size.

As indicated above, the sub-net use and capability of our method is both unique and powerful. Although the ability to calculate sub-nets using social network connection metrics alone is useful it is more powerful, and more illuminating to either amplify or refine the sub-net metrics produced by procedure with more salient characteristics that have a bearing on individual behaviors and psychological characteristics, the characteristics of social connections, and the operational (behavioral) characteristics of the messages that are exchanged (frequency, size, volume, and so on). This is a useful set of information that can be used to constrain the use boundaries of a particular term use or, more generally, can be used to establish synonyms, metaphors and other indicators of potential meaning and the evolution of meaning in vocabulary in general. This capability does not exist in generally-available text topic detection approaches.

8 Method Process Flow

As shown in Figure 2 there are 8 steps to the general process.

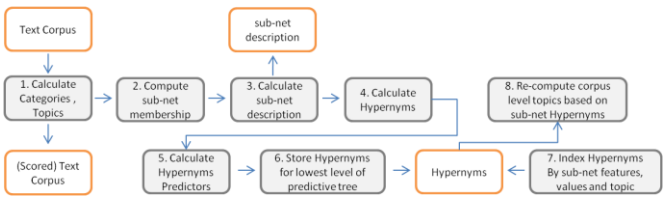


Figure 2: Process to Generate Text Topic with Meaningful Variations in Vocabulary and Mode of Expression

The first step, discussed in earlier sections, begins with one of the standard methods of calculating categories or topics. It should be noted that this proposed method can be used to adaptively expand both categories and topics. An

example of how categories and topics can be derived from the top-level (entire) text corpus is provided in the illustration of Step 1. Subsequent steps only show the operation of the proposed method with topics. Topics are more complex constructs than classifications and there is always a direct, unambiguous mapping of topics to classifications. Therefore, any method that is illustrated for topics can always be faithfully and accurately applied to categories as well.

8.1 Calculate Categories, Topics

An example that shows the classifications and topics that can be derived from a text corpus is shown below. The classifications were automatically discovered using SAS's Enterprise Content Classification "automatically detect sub-categories" feature. The topics were automatically discovered using SAS's Text Miner "Text Topics" node (see "References" under "LSA").

Example of Category and Topic Derivation

Table 1: Extracted Topics (And Associated Terms)

All Terms	Category?	Topics				
		T1	T2	T3	T4	T5
Dell		dell	3d hd	120hz	dell xps	screens
Alienware		duo	intel	core	alienware m11x	inch
Laptop	Yes					
Gaming		high-end	r3 game laptop	gaming		
alienware m17x		dell alienware m17x	+unleash	alienware system		
eBay		ebay	deal	+price	+well deal	alienware keyboards
Aurora						
dell xps						
120hz						
3d hd						
refresh						
screens	Yes					

Topic designators are general purpose description mechanisms that apply to the entire text corpus. In Table 1 we show 3 columns of results. The first column – "All Terms" – lists the terms in the document ordered by frequency. "Dell" is the most frequent term in this set of documents.

The second column – labeled "Category?" – describes whether the term belongs to one of the discovered categories or not. There are two categories shown in Table 1: a category for "laptop" and a category for "screens".

The third column – labeled "Topics" – shows the discovered topics 1 – 5. Discovered topics that contain one or more terms that appear in the document collection (shown in Column 1) appear on the same row as the first term that they contain. Table 1 shows the topic "dell, 3d hd, 120hz, dell xps, screens". This topic appears on the first row opposite the term "dell". Topic 5, which occurs on row 2 of Table 1, contains topic 5. This topic contains the terms "duo, intel, core, alienware m11x, inch".

Once the topic description of a Text Corpus have been discovered then the topic features can be used to develop a scoring model that can then apply the topic designators to the source collection of text documents. The topic designators can also be used to apply topic descriptions to unseen documents. This process is illustrated in Figure 3.

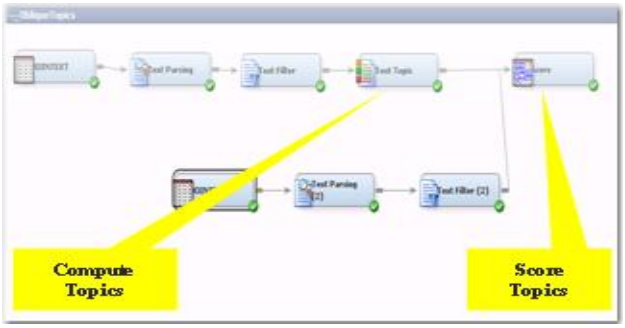


Figure 3: Example Process Flow Diagram Showing Calculation of Text Topics and Application of Topic Indicators to the Source Data

8.2 Compute sub-net membership

Once the discovered topics have been applied to the text corpus then we can take the next step of discovering the relevant sub-net attributes that are likely to apply. We compute the initial membership of the social community that characterizes the conversation in two ways:

1.Node-link characteristics ("physical")

2.Semantic link characteristics ("virtual")

The first method is labeled "physical" because the calculation depends on observed behavior that points to the occurrence of an actual message exchange between two nodes (for example, the appearance of a common "threadid" field in the database).

The second method is labeled "virtual" because the calculation depends on an inferred link that is not directly observed and which depends upon a common shared characteristic (in our case we compute semantic similarity and determine that two nodes are linked when they share the same semantic features).

Node-link calculations: Here we seek to observe the existence of a shared message exchange between participants. In blog sites and Twitter, for example, we will seek to discover whether participants are linked by a "thread". In blogs, thread links exist when individuals post messages to a common blog title. In Twitter, thread links exist when individuals reply to others or retweet others' original messages. Similar mechanisms appear in most forms of electronic messaging and, when available, can be used to compute "communities of interest" based on shared conversational threads. Once the nodes (individual participants) and connections (links or edges) have been defined then the metric characteristics of the network can be used to derive community membership. Community membership is derived by grouping nodes into communities such that there is a higher density of links within communities than between them.

The implementation used in SAS's Proc Optgraph is based on the paper "Fast unfolding of communities in large networks" (http://findcommunities.googlepages.com/).

Semantic link calculations. In our method we may determine linkages between participants based on shared semantic characteristics. *In this "quantum" view of connectivity between nodes we do not expect to observe a physical connection (nor is it necessary for nodes to be members of a common network or to be in a shared geography, for example). A connection is determined to exist if two or more nodes carry the same semantic content during a shared time frame.*

Shared semantic content is determined by computing singular value factorizations of the conversation that appears on the text corpus. Svd scores are calculated for all the members in the corpus level conversation and these scores are, in turn, clustered together. In our definition all members in the same cluster are defined as "similar" and therefore share the same semantic link with one another. Each cluster is therefore a separate (and mutually-exclusive) community.

We use 4 dimensional attributes to calculate relevant social group discriminating characteristics. The characteristics of these 4 dimensions are used to determine the specific sub-net discriminators that are used to derive appropriate hypernyms for any given topic that is detected in the text corpus.

8.3 Calculate sub-net descriptors

Each sub-net is uniquely defined by the input fields that are used to create it and the field values that are used to form the branches of a decision tree that defines the hierarchical branch attributes of the sub-net. The hierarchical branch attributes are formed by applying 3 types of sub-net discriminators:

(1) individual, (2) social and (3) operational

Textual expression emerges from a process that is visualized in Figure 4.

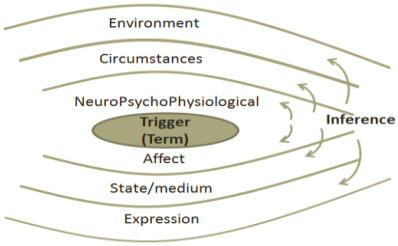


Figure 4: Environmental Context of Conversational Expression

The middle part of Figure 4 illustrates the trigger event – in this case the use of a particular expressive term – that results from the top-down flow of the environment and circumstances of the conversation. The environmental context is translated through the conceptual apparatus (neurochemistry, psychology, physiology) of the message originator and results in the generation of a given expression, as shown in the bottom part of Figure 4.

We use the sub-net descriptors indicated here to operationalize the environment and circumstances that affect the triggering expression-generating event. These sub-net discriminators are as follows:

1) Individual characteristics include: Age, Gender, Education, Marital status, Interests, Affiliations and memberships, Psychological (e.g. behavioral profile)

2) Social organization characteristics include: Geographic and temporal location, Social role: Leader, follower, marginal, Social Influence, Community size, density, dispersion, Community character (for example, friends, family, business associates, social, recreational and spiritual groups)

The messages that are exchanged in the conversation also have operational characteristics that can impact the sense characterization.

3) Operational characteristics include: Message recency, Message frequency, Conversation acceleration rate, Message mood state, Type of exchange: personal, professional

The specific form of the sub-net is calculated using the predictive modeling capability. This capability shows the relationship between a target value (the Topic) and various input values (individual, social and operational inputs). The decision tree is formed by searching for important relationships between the topics that have been extracted at the top-level text corpus and the three sets of field values that have been calculated as sub-net discriminators.

A causal sequence is implied in the unfolding of the sub-net characteristics. As shown in Figure 3, an element of the conversation is filtered through the participants' individual characteristics, social characteristics and message characteristics, in that order. (Clearly, the individual background will influence social choices and this context will further be influenced by the operational features of the messages that are exchanged in the conversation.)

8.4 Calculate sub-net discriminators

As indicated above we use Topics as the targets in our sub-net detection and the calculated individual, social and operational inputs described immediately above. We can easily identify combinations of sub-net attributes using supervised decision trees. An example of this approach is shown in Figure 5.

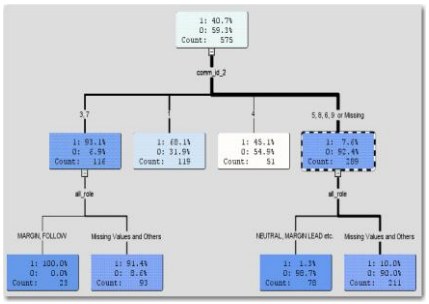


Figure 5: Sub-net Attribute Identification (Showing the Predictive Sub-classification of Text Topic 1)

In Figure 5 we see that, while discussion of Topic 1 accounts for about 40% of the conversations in our example, this conversation is mostly confined to Communities 3 and 7 (shown on the immediate left as the first node of the first level of the decision tree). In communities 3 and 7 we see

that the combined incidence of Topic 1 is about 93%. In this example we do not specify whether this is "physical" or "virtual" community membership (it could be either metric).

On the next level of the tree, we see the appearance of one of the social inputs (community role). When the community role is identified as "Marginal" or "Follower" we see that the incidence of Topic 1 increases to 100% (shown in the left-most node at the bottom left of the diagram. With other community roles – which includes the "leader" role -- the incidence of Topic 1 discussion decreases to 91% (a significant difference).

The advantages of the decision tree methodology are that it allows us:

1) To control the sequence of branching (to preserve the top-down nature of individual, social and operational feature unfolding);

2) To compute whether a particular discriminating feature is significant (in a statistical sense) or not; and,

3) To build arbitrary combinations of values on the branch partitions (so as to form sub-nets with the most discriminating characteristics identified on the branch labels).

Although this is a simple example it illustrates the general principal of computing sub-nets – here shown as the characteristics of the bottom nodes of the decision tree display – as discriminating features that define homogenous sub-groups of the population. It is among these sub-groups that we expect to see variations in terms and modes of expression that point to subtle and gross changes in vocabulary used to describe a given topic or concept. In some cases, the changes in vocabulary could be leading indicators of new forms of language; i.e. they are indicators of "memetic changes" in vocabulary that need to be tracked in order to ensure consistent topic tracking over time (Leskovec et. al., 2009).

In any case, vocabulary from these bottom-level nodes is harvested and placed in a table for potential re-use in re-computing the occurrence of the associated topic based on the co-occurring "hypernyms" (topic 1 is shown in the illustration but the method applies to all topics).

A sub-net descriptor table, illustrated in Table 2, is created to store the defining attributes of the sub-net. The defining attributes are equivalent to the lowest level branch separation rules in the diagram.

Table 2: Subnet predictor table

Sub-net ID	Precondition	Outcome
1	if comm_id_2 IS 1	then TextTopic_1 = 0.68
2	if comm_id_2 IS 4	then TextTopic_1 = 0.45
3	if comm_id_2 IS 3, 7 AND all_role IS MARGIN, FOLLOW	then TextTopic_1 = 1.00
4	if comm_id_2 IS 3, 7 AND all_role equals Missing	then TextTopic_1 = 0.91
5	if comm_id_2 IS 5, 8, 6, 9 or MISSING AND all_role IS NEUTRAL, MARGIN, LEADER	then TextTopic_1 = 0.01
6	if comm_id_2 IS 5, 8, 6, 9 or MISSING AND all_role equals Missing	then TextTopic_1 = 0.10

8.5 Calculate Hypernyms

In discovery searching for term equivalents to a given topic in a social grouping or sub-net we use an unsupervised method where we do not know exactly what we are looking for. Our strategy is to compute meaningful combinations and permutations of the vocabulary that is used in this sub-net so as to maximize our ability to detect alternative and emerging expressions. We call these computed combinations and permutations "hypernyms".

"Hypernyms" are like synonyms in that they are alternative wordings for the same identifier (just as "tiny" and "small" are synonyms). Unlike synonyms, hypernyms may involve bigrams, collocations and other compound terms that do not exist as a contiguous entity in the text stream.

Table 3: Hierarchy of Term Components Used to Identify Hypernyms (Ordered by Granularity)

Hypernym Source	Example
Collocations	Games + computer
sequential collocations	Computer games
Bigram	computergames
noun phrases	Gaming computer
named entities	SAS Institute Inc.
part of speech	Programmer -- noun
raw term	software

Table 3 shows the various term components that are used to identify hypernyms. Bigrams are two contiguous terms with spaces and other term separators removed. Noun phrases and parts of speech are used as lower level precursors to bigrams (as are the raw terms themselves). Collocations include term combinations that, while separate, are nevertheless present in the same textual context at a higher-than-average probability.

8.6 Store Hypernyms for lowest level of predictive tree

The example in Figure 5 presents the only significant discriminating attributes that were found. According to the method described here, the resulting sub-nets represent specific social communities where the interpretation of Topic 1 is likely to be different (alternatively, completely new topics are likely to emerge).

Once the sub-net discriminators are identified, it is possible to extract significant hypernyms that are constructed ahead of time using the decomposition hierarchy of term components described in Table 3, above. The extracted hypernyms for our example shown in Figure 4 are displayed here in Table 4.

Table 4: Hypernyms by Sub-net Attributes

Margin and Follower	*Other Roles*
alienware worth	desktop
alx review	heavy-duty
am11x	laptop
area-51 alx	small gaming
area-51 m11x	supply cord
breathing beast	high performing
Brilliance	widescreen

These hypernyms can be then be used to update the Topic 1 definition (by adding these terms as part of the topic computation algorithm). At this point the corpus of documents can be re-scored.

Sub-net Topic Detection: Once the subnets have been identified then, in addition to updates of the Topic definition, we may choose to create entirely new topics for the sub-net. At this point the method is recursive, so the approach of finding topics at the text corpus level (top-level) is simply re-produced for each of the identified sub-nets.

8.7 Index hypernyms by sub-net features, values and topic

Once the hypernyms for the sub-nets have been identified they are stored in a hypernym table that can subsequently be used to re-score and update the occurrence of the matching topic in either the original text corpus or in an entirely new text corpus.

Table 4: Example of hypernym table storage by Topic

Sub-net ID	Topic	Hypernyms
3	1	alienware worth
3	1	alx review
3	1	am11x
3	1	area-51 alx
3	1	area-51 m11x
3	1	breathing beast
3	1	brilliance
4	1	desktop
4	1	heavy-duty
4	1	laptop
4	1	small gaming
4	1	supply cord
4	1	high performing
4	1	widescreen

8.8 Re-compute Corpus Level Topics based on sub-net hypernyms

The scoring model for the re-computation of the text topic occurrence is the same as the process that is illustrated in Figure 3. The sole difference is that in the re-computation scenario we add terms taken from the hypernym table to re-compute the occurrence of a topic. The term definition for Text Topic 1 is shown in Figure 6.

EMWS1.TextTopic4_initTopics

Role	Term	_Weight_	Topic
	alienware worth	2.000	3d hd,120hz,hd,+screen,de
	a1x review	2.000	3d hd,120hz,hd,+screen,de
PERSON	sandy bridge	1.727	3d hd,120hz,hd,+screen,de
	am11x	2.000	3d hd,120hz,hd,+screen,de
Noun	cpu	1.708	3d hd,120hz,hd,+screen,de
Noun	120hz	1.761	3d hd,120hz,hd,+screen,de
Prop	hd	1.761	3d hd,120hz,hd,+screen,de
	area-51 alx	2.000	3d hd,120hz,hd,+screen,de
Noun	area-51 m11x	2.000	3d hd,120hz,hd,+screen,de
PROP_MISC	dell xps	1.729	3d hd,120hz,hd,+screen,de
	breathing beast	2.000	3d hd,120hz,hd,+screen,de
Punct		0.365	3d hd,120hz,hd,+screen,de
Prop	xps	1.712	3d hd,120hz,hd,+screen,de
Verb	refresh	1.699	3d hd,120hz,hd,+screen,de
Punct		1.385	3d hd,120hz,hd,+screen,de

Figure 6: Example text topic definition with new terms added

When the new topic definition was applied in our example we observed an increase in the number of documents that contained Topic 1 from 132 to 224. This is an increase of about 14%. These two figures are shown in the report table from the re-computation step: under the last column, labeled "Category", we see that 224 documents contained the "User" topic that we defined while the next row, labeled "Multiple", shows the 132 documents that were identified with the original, unmodified Text Topic 1.

Topics

Topic ID	Document Cutoff	Term Cutoff	Topic	Number of Terms	# Docs	Category
1	0.030	0.001	3d hd,120h...	38	224	User
2	0.060	0.176	3d hd,120h...	21	132	Multiple
3	0.060	0.366	+ebay store...	30	142	Multiple
4	0.060	0.154	duo,core,+a...	33	73	Multiple
5	0.060	0.183	dell alienw...	40	105	Multiple
6	0.060	0.183	+ces,r3,hig...	22	102	Multiple
7	0.060	0.181	store,+ebay...	36	117	Multiple

Figure 7: Re-score incidence of text topic 1

9 Conclusions

This technique is useful in the search and tracking of themes, topics and concepts in any text collection, most especially electronically-encoded text such as found on the internet, in social media and blog archives. This is a very general-purpose method since it enables the automatic expansion of search terms and definitions to include variations in expression used by different sub-populations and new expressions, neologisms and so on.

10 Contact Information

Your comments and questions are valued and encouraged.

Contact the authors at:

Dinesh Batra, Dinesh.Batra@anniksystems.com
Gurpreet Singh Bawa, Gurpreet.Bawa@anniksystems.com
Vivek Anand, Vivek.Anand@anniksystems.com

11 References

[1] Leskovec, L. Backstrom, J. Kleinberg. Meme-tracking and the Dynamics of the News Cycle. ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining, 2009, Google (http://memetracker.org).
[2] Acronym for Latent Semantic Analysis. Both Latent Dirichlet Allocation (LDA) (http://www.cs.princeton.edu/~blei/papers/BleiNgJordan2003.pdf)
[3] The SAS Text Miner Topic Node (http://www.sas.com/resources/factsheet/text-miner-factsheet1.pdf)

Semi-automatic Metadata Extraction from Scientific Journal Article for Full-text XML Conversion

Sukyoung Kim, Yoonsung Cho, and Kihong Ahn

***Abstract*—By the increasing continuous academic researches, the volume of scientific articles has dramatically reached unpredictable level. To facilitate archive and publication, many scientific journals in Korea are actively adapting Open Access (OA) policy. In addition, it has more attractable than commercial printing of companies that freely provide the full text of article published in scholarly journal through web to user. Because of difficulty to convert automatically unstructured format such as pdf document into full-text, which is structured with accuracy, the most full text conversion works in scholarly journal publisher have been conducted with human interaction. To deal with the problem of low reliability in the automatic metadata extraction and help with minimum human interaction, we propose semi-automated metadata extraction method based on rule-based method and machine learning method. In this experiment, we verified the performance under 26 different journals in Open Access Korea Central (OAK Central). We only cover two part (elements of front and back) as part of an effort to convert full-text xml based on JATS v1.0. As a result, our proposed method reached F1 = 94.1% in front and F1 = 92.5% in back.**

I. Introduction

World Wide Web played a huge role in change from the center of traditional paper media into the center of electronic publishing on circulating structure of scholarly journal article. Electronic publishing has promoted research activity in various research fields internationally by facilitating the expeditious publication. Journals and conference proceeding that exist today publish about 2.5 million articles per years have peer-reviewed, propelled by this advantage [1]. But electronic journals are forged mainly with commercial printing of companies, half of whole scholarly journal article are produced by the top-5 commercial companies [3]. However, these publishers typically provide their contents with the client only who subscribe to their publishers. In order to have competitive power with commercial printing of companies and have stature, non-commercial scholarly publisher have to offer differentiated interoperability service. The kinds of differentiated interoperability service are Open Access (OA), JATS XML, Pubreader, DOI, CrossCheck, CrossMarkm FundRef, Open Researcher and Contributor ID (ORCID), QR code, mobile application and multimedia [3]. From among these, it is critical for differentiated core strategy to provide OA policy, which provides unrestricted online access to articles published in scholarly journals and JATS XML, which describe the content and metadata of journal articles on Web. Especially, providing full text of journal article on Web can maximize not only visibility and accessibility of the scientific journal article but also provides high quality service in view of Digital Library when the well-structured data such as XML is more tractable in managing collections of scholarly articles than unstructured data such as pdf. As Fig. 1, many of the internal and external scholarly journals have been in accordance with OA and full text XML based on JATS. However, because of difficulty to convert automatically unstructured format such as pdf document into full-text, which is structured with accuracy, the most full text conversion works in scholarly journal publisher still have been conducted with human interaction. To reduce time and coast and help with minimum human interaction, most of scholarly journals in Korea are faced with the necessity of automatic metadata extraction and conversion method with accuracy.

Automating this conversion work requires programmatic access to the typographical layout of elements page as well as to their logical/rhetorical function within article [6]. Therefore, we view this subject from Natural Language Processing (NLP) and Information Extraction techniques. Previous research in automatic metadata extraction from pdf format of scholarly journal articles is divided into two main categories. The first is the rule-based method. This method, if rule set is constructed about target domains in advance, gives best performance, but it does not perform well when changed target domain. On the other hand, the second, machine learning method is most popular in information extraction fields. It is not necessary for considering changed target domain when sufficient volume of training set is constructed. With these advantages, most of relevant researches are based on the machine learning methods.

To reduce time and cost by the human-curated XML conversion method, in this paper, we propose a semi-automatic metadata extraction module based on the rule-based method and the machine learning method. Our aim is to deal with the problem of low reliability in the automatic metadata extraction and help with minimum human interaction. From this point of view, we focus on

Sukyoung Kim is with Department of Computer Engineering, Hanbat National University, Daejeon, South Korea (e-mail: kimsk@hanbat.ac.kr).

YoonSung Cho is with Department of Computer Engineering, Hanbat National University, Daejeon, South Korea (e-mail: dibbul5456@gmail.com).

Kihong Ahn is with Department of Computer Engineering, Hanbat National University, Daejeon, South Korea (e-mail: khahn@hanbat.ac.kr).

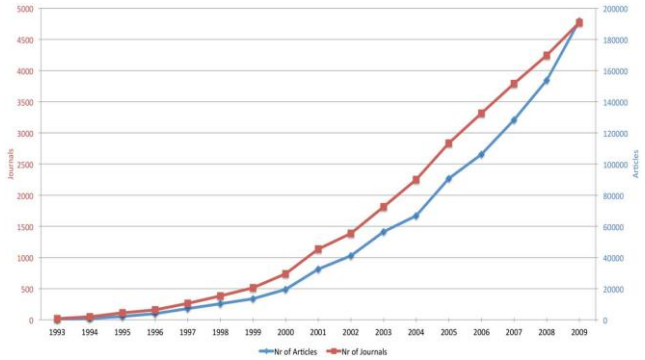


Fig. 1. The development of open access publishing
("The Development of Open Access Journal Publishing from 1993 to 2009," Mikeal Laakso et al, 2011, PLoS ONE, 6, p. 6.)

precision to have high reliability in the automatic methods.

This paper is organized as follows: section 2 introduces related work and section 3 presents our semi-automatic conversion method, and section 4 shows error analysis and evaluation. Finally, we give our conclusion and discussion for future directions.

II. Related Work

A. Journal Article Tag Suit

The National Center for Biotechnology Information (NCBI) originally created the Journal Archiving and Interchange Tag Suite with the intent of providing a common format in which publishers and archives could exchange journal content [22]. The JATS v1.0 is a revision of the NLM Journal Archiving and Interchange Tag Suite version 3.0. JATS 1.0 provides a common XML format in which publishers and archives can exchange journal content. This XML is mainly consists of front, body, and back. The element of front describes article header information, which is usually content on first page. The second, body is describes the main body, which includes contents from Introduction to the front of Reference. The back describes whole reference.

B. Rule-based Approach

The rule-based methods generate extraction rules by using knowledge about the target domain and use the rules to extract metadata from pdf documents. This approach is difficult to guarantee the performance when the target domain is changed. But the best performing systems are often handcrafted [19]. The CiteSeer system as the first search engine for scientific literature to incorporate Autonomous Citation Indexing use rule-based metadata extraction system [19]. Also, to overcome weakness in new application domain, after the rule templates about diverse document set are defined and these template are used to classify new document by assigning it to a group of documents of similar layout [23]. The Layout-Aware PDF Text Extraction (LA-PDFText) provides an open source system that extract text blocks from pdf document and classifier them into logical units based on rule [8].

C. Machine Learning Approach

The machine learning methods are popular methods in many text processing fields. This approach is flexible in new document domain but requires that a sufficient training data is available and the task manually labels a set of training data [19]. In Previous research, with enhanced state transition probability, full second order Hidden Markov Models (HMM) proposed [21]. However, HMMs are based on the assumption that features of the model they represent are not independent from each other. To solve this label bias problem and include a wide variety of arbitrary, non-independent feature of the input, Conditional Random Fields (CRFs) method is proposed [22]. With this advantage, The SectLabel extracted logical structure by using richer representation of the document that includes features from Optical Character Recognition (OCR) [7]. To relax segmentation uncertainty and improve extraction performance, [10] proposed co-reference information extraction method. Another method, Support Vector Machine (SVM) was used. To improve the line classification, an iterative convergence procedure was proposed [12]. Also, there are hybrid methods, which mixed one or more machine learning methods. [15] proposed metadata extraction method based on measurement fusion rule. In this experiments, the three learning method such as HMM, SVM and CRF are used.

III. Semi-Automatic Extraction Method

To convert pdf format into well-structured XML automatically, extraction model have to recovery text structure in pdf document by analyzing spatial and layout feature of raw texts. Than The metadata set defined by JATS 1.0 are (article-title, author, affiliation, pub-date, license, abstract, and volume, etc.). The problem to cover range of publication format style (geometric information, layout feature, font feature) from heterogeneous journals still cause that the results tagged as specific class are misleading. Although many of the previous works takes these features into account, the extraction result still is poor in noisy input data (domain is out of space or input data is out of sequence). We consider that it can greatly influence accuracy to refine the input data before applying the machine learning method instead of using full text on pdf document directly into the input data of the machine learning method. Our proposed method applied the rule-base method to refine input data and pre-empt the problem, which is misleading by labeling text chunk block as basic type. Also we adopted machine learning method to label each individual text line to a target class such as element defined in this paper according to JATS 1.0. A brief summary of our proposed work is as follow:

A) An open-source tool, LA-PDFText based on rule, extracts text blocks on pdf documents and determine each type of text block.
B) Block segmentation and line feature are constructed in order to split text block that might be belong to multiple class into text lines.
C) In phase of machine learning, each individual text line and token is labeled as target class with the Condition Random Fields (CRFs) model.

A. Rule-based Text Block Classification

There are various open sources to detect text line or block. Apahce Tika and Apache PDFBox return simple string in pdf document without additional features. CrossRef pdf2xml provides xml format of text with line spatial feature and font feature. The most recent open source of these is the LA-PDFText. It is an open-source tool for accurately extracting text from full-text scientific articles [8]. This open source developed to help with Natural Language Processing (NLP) developer who has to extract text from pdf documents. LA-PDFText extracts not only text block from pdf document but also provides the interface that developer can generate custom rules to classify the type of text block. Fig. 2 is the result of text block classification with LA-PDFText. In this experiments, we designed the rules, which are customized for each journal by analyzing whole format of journals. The used rules for block type classification are in Table 2.

B. Block Segmentation and Line Feature Generation

Text block that separated by the rule spilt cause to consider just layout feature and font feature of text, so it lacks amount of information to parsing by the end result(JATS elements). A text line in Text block can be labeled to target element, or one or more elements. Therefore, First, text block separate a text line unit and once again, to do classification by each line target element. The best case is labeled to one of target element to clarify by rule which decided about text block. In Article-title's case, layout and font feature is clearly different from other text so, in including text for text block case, is almost single class. But Most of text block is labeled to one or more multi class, so it needs segmentation work.

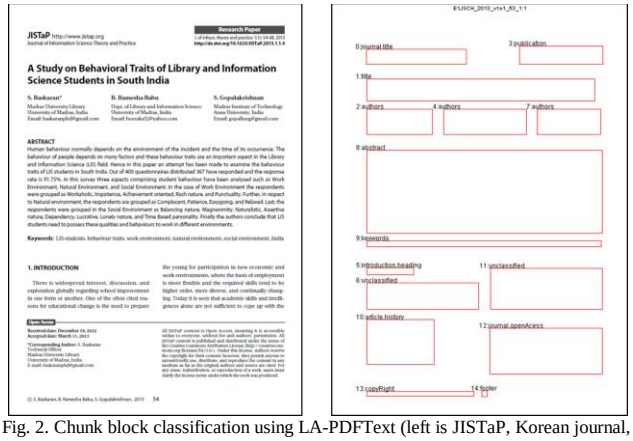


Fig. 2. Chunk block classification using LA-PDFText (left is JISTaP, Korean journal, right is the result of chunk block classification)

TABLE I
THE LIST OF USED RULE FOR TYPE CLASSIFICATION

Block Type	Condition
title	*pageNumber==1* *inTopHalf==true* *mostPopularFontSize>=15* *readLeftRightMidLine()=="MIDLINE"*
publication	*readNumberofLine()<=5* *pageNumber==1* *inTophalf==true* *regularExpression==true ("[vol|no]")*
author	*pageNumber==1* *inTopGalf==true* *regularExpresesion==true ("^([A-Za0z\\.]*))...* *fontSize=="large_font"*
openAccess	*redNumberofLine()>=5* *pageNumber==1* *inTopHalf==true* *regularExpression==true (e.g. 'This is an open Access')*
keyword	*pageNumber==1* *regularExpression==true ("^(Key|Index)")*
	. . .
abstract	*inTropHalf==true* *readLeftRightMidLine()=="MIDLINE"* *readNumberofLine()>=5* *RegularExpression==true ("^(Abst|ABSTR)")*

This is a rule sample, which are used to detect text chunk block (sample journal : GISTaP)

1) Block Segmentation

Text block that separated by the rule spilt cause to consider just layout feature and font feature of text, so it lacks amount of information to parsing by the end result(JATS elements). A text line in Text block can be labeled to target element, or one or more elements. Therefore, First, text block separate a text line unit and once again, to do classification by each line target element. The best case is labeled to one of target element to clarify by rule which decided about text block.

TABLE II
THE LIST OF USED FEATURE

Feature	Description	Scope
INITNUM	First letter starts with digit	Front, Back
INITCAP	First letter starts with capital	Front, Back
ALLCAP	All letter is capital	Front, Back
LARFONT	Font size is large	Front, Back
NORFONT	Front size is normal	Front, Back
EMAIL	Character is with e-mail pattern	Front, Back
DATE	Character is with time expression	Front, Back
PAGE	Character is with page pattern	Front, Back
DOI	Character is with doi pttern	Front
NUMBER	Character including digit	Front
FONT-SIZE	Font size	Front, Back
FONT-STYLE	Font style	Front, Back
ISSN	Character is with issn pattern	Front
PAGE	The page including current character	Front
LINE_LONG	Text line is long	Back
LINE_MDLLE	Text line is middle	Back
LINE_SHORT	Text line is short	Back
PERSON	Character is with person name	Front, Back
ORGANIZATION	Character is with organization	Front, Back
LOCATION	Character is with location	Front, Back
LAST-DOT	Last letter end with dot	Back
LAST-COMMA	Last letter end with comma	Back
BLOCK-TYPE	Text block type including current text	Front, Back

In Article-title's case, layout and font feature is clearly different from other text so, in including text for text block of case, is almost single class. But Most of text block is labeled to one or more multi class, so it needs segmentation work.

2) Line Feature Generation

It is point of performance of Machine-learning that wise

choice of feature set [10]. Combination to key feature which can classify the class is more helpful for performance quality than collecting various feature set. In this experiment, 23 kinds of feature are used. Table 2 is list of feature used in this experiment. Lexical feature, font feature, word entity feature, chunk type feature and so on of text line is used. The type of text block which is already determined by established rule clarify indirect target element of the text. Such a text block classification through rule-based method is very useful in extracting key feature.

C. Machine Learning-based Classification

The module of semi-automatic metadata extraction combined with rule-based and machine-learning is illustrated in Fig. 3. In this step, text line set divided in previous step is need to classification as each target element again. In this experiment, we use Conditional Random Field (CRF) in order to classify appropriate target element by considering a lot of context of text line and feature of each text. CRF is applied on field to solve sequence labeling problem because it can solve the label bias problem of Hidden Markov Model (HMM) and support arbitrary dependent feature and joint inference over entire sequence. The classification model we propose is divided into two models to extract front metadata and back metadata. First, model which tag elements of front area defined by JATS. Second, model which tag elements of back area.

1) Front Classification

Front area that defined by JATS is consist of 18 kinds of sub-element in <journal-meta> and 39 kinds of sub-element in <article-meta>. In First column of Table 6, in this experiment, there are 18 kinds of element that defined target class. Out of text block that determined single class by 18 previous rules, the other text block that has multiple classes is applied front CRF model.

2) Back Classification

Back part is consists of seven elements. Nature of LA-PDFText that finds the continuous text block, all of reference body is cognized one block by text layout and font feature. To extract each individual reference block, we make two kinds of classification step. One is looking up and separating individual reference blocks in all of reference body. Previously, separate the reference text in body by line, tracking down first and last line, separating to individual reference. The other is dividing individual reference as token unit. Because of reference has a multi class on one text line. After divided text token set as token unit generating feature-vector, eventual seven elements (author, title, source, year, volume, number and page) are classified.

IV. Error Analysis and Evaluation

A. Experimental Data

In this paper, the dataset to test our semi-automatic metadata extraction module is collected from Open Access Korea Central (OAK Central), which is a free archive of scholarly & scientific journal literature in Korea. We downloaded 560 papers from 26 different journals in OAK Central. In whole dataset, 260 papers are used for training set and rest 300 papers are used for testing set.

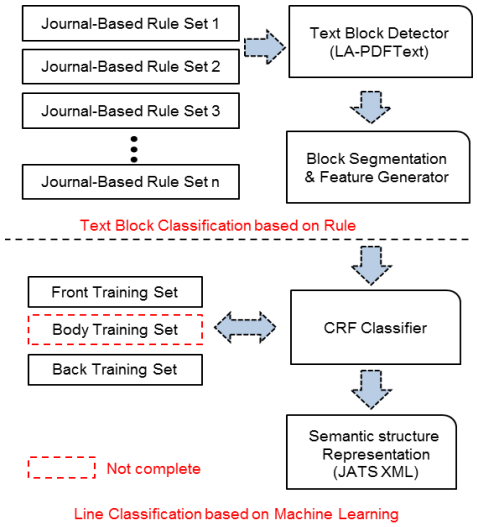


Fig. 3. Overview of Semi-Automatic Metadata Extraction Module

B. Evaluation Metrics

To verify our proposed method, we used basic measure, Precision (P), Recall (R), and F1-measure. This thesis defines the following:.

A: The number of true positive elements (e.g. 'title' token tagged as 'title').

B: The number of false negative elements (e.g. non-title token tagged as 'title')

C: The number of false positive elements (e.g. 'title' token tagged as anything but 'title')

Precision (P) = A / (A + C)
Recall (R) = A / (A + B)
F1 = (2 * P * R) / (P + R)

C. Evaluation of Extraction Result

Table 4 is precision, recall and F1-measure of front elements. Before applying machine learning method, using rule-based methods gets high precision on abstracting text block and Classifying basic type about text block. Text block which belong single class is tagged as JATS element, not applied CRF. Typical single class includes title, doi, abstract, accepted-date and received date. If we analysis single and multiple class about text block previously and use refined data in machine learning method, can expect reliable output. Table 5 is a result of back part. In the case of back part, we extract metadata only using machine learning method.

TABLE VII
CONFUSION MATRIX OF FRONT ELEMENTS

	j-id	*j-title*	*a-title*	*a-doi*	*c-group*	*aff.*	*a-date*	*r-date*	*c-holde*	*license*	*abstra.*	*volume*	*issue*	*other*
j-id	18	0	0	1	0	0	0	0	0	0	0	0	0	0
j-title	0	76	0	0	0	0	0	0	0	0	0	0	0	22
a-title	0	0	98	0	0	0	0	0	0	0	0	0	0	0
a-doi	1	0	0	80	0	0	0	0	0	0	0	0	0	13
c-group	0	0	0	0	226	14	0	0	0	0	3	0	0	34
aff.	0	0	0	0	0	160	0	0	0	0	3	0	0	31
acc-date	0	0	0	0	0	0	89	0	0	0	0	0	0	0
rec-date	0	0	0	0	0	0	0	89	0	0	0	0	0	0
co-holder	0	0	0	0	0	0	0	0	60	0	0	0	0	18
license	0	0	0	0	0	0	0	0	0	54	0	0	0	4
abstract	0	0	0	0	0	0	0	0	0	0	95	0	0	3
volume	0	0	0	0	0	0	0	0	0	0	0	98	0	1
issue	0	0	0	0	0	0	0	0	0	0	0	0	88	1

D.Error Analysis

Table 7 is classification confusion matrix about front area of JATS XML. Table 6 is classification confusion matrix about back area. We need separating as each individual reference and classifying as JATS back elements from individual reference because extracted reference body by rule-based method includes all of reference area. Eventually, classifying as JATS back elements take a lot of effect on output of separated from reference body. Most errors of reference are originated in separating as individual reference procedure. It is case that two references are classified one reference and one reference is classified two references.

V. CONCLUSION AND FUTURE WORK

Appearance of JATS which standard Open Access (OA) policy shows further need for the research that extracts structured data and analysis semantic logical structure. Many scholarly journal publishers in Korea still rely on manual information extraction owing to low reliability in the automatic metadata extraction system. To deal with the problem of low reliability in the automatic metadata extraction and help with minimum human interaction, we propose semi-automated metadata extraction method based on rule-based method and machine learning method.

The model proposed in this experiment mixed rule-based and machine-learning is verified under scholarly publisher that treat small scale journal, not search engine or digital library which highlights scalability. We verified the performance under 26 different journals in Open Access Korea Central (OAK Central). Thus, proposed model prosecute text block classification and tags block which type of clear single class as target class, do not use machine-learning method. If we do not analyze logical structure of journal and only treat all of text in article with machine-learning, we hardly see a result of extraction for service. If we remove ambiguity of text block through rule-based method and apply machine-learning method after refining data, we can expect reliable high quality data.

Several issues remain to be investigated. First, the portion of body in JATS is difficulty to extract metadata such as figure extraction, table recognition, and mathematical expression automatically. Second, our experiments only are conducted in small specific journals (26 journals in OAK Central) to get high reliability and help with minimum human interaction. Finally, open data integration problem are occurred because we use the rule-based methods

TABLE IV
THE ACCURACY OF FRONT ELEMENTS

Article-front	**P**recision	**R**ecall	**F**1
Element	**96.47**	89.35	**92.51**
journal-id	94.74	94.74	94.74
journal-title	100.0	77.55	87.36
article-title	100.0	100.0	100.0
article-id(doi)	98.77	85.11	91.43
contrib-group	100.0	81.59	89.86
affiliation	91.95	82.47	86.96
accepted-date	100.0	100.0	100.0
received-date	100.0	100.0	100.0
copyright-holder	100.0	76.92	86.96
license	100.0	93.10	86.43
abstract	94.06	96.94	95.48
volume	100.0	98.99	99.49
issue	100.0	98.88	99.44

Precision, recall and F1-measure of the front from OAK Central data set (%)

TABLE V
THE ACCURACY OF BACK ELEMENTS

Article-front	**P**recision	**R**ecall	**F**1
Average	**96.80**	92.00	**94.15**
person-group	98.49	97.12	97.80
article-title	85.05	97.12	90.69
page	98.58	93.78	96.12
year	99.22	95.12	97.12
source	97.28	80.74	88.24
volume	98.96	90.60	94.59
number	100.0	89.53	94.48

Precision, recall and F1-measure of the front OAK Central data set (%)

TABLE VI
CONFUSION MATRIX OF FRONT ELEMENTS

	name	*title*	*page*	*year*	*sou.*	*vol.*	*num.*	*other*
name	3135	42	0	0	0	0	0	51
title	18	3241	0	0	33	0	0	42
page	3	9	2712	24	9	6	0	129
year	3	18	3	3039	3	3	0	126
source	24	477	15	0	2679	0	0	123
volume	0	6	21	0	24	1995	0	156
number	0	0	0	0	6	12	462	36

REFERENCES

[1] Y. Gargouri, C. Hajjem, V. Larivière, Y. Gingras, L. Carr, T. Brody and S. Harnad, "Self-selected or mandated, open access increases citation impact for higher quality research," *Plos one*, 5(10), 2010, e13636.

[2] M. Laakso, P. Welling, , H. Bukvova, L. Nyman, B. C. Björk and T. Hedlund, "The development of open access journal publishing from 1993 to 2009," *PloS one*, 6(6), 2011, e20961.

[3] S. Huh, "ScienceCentral: open access full-text archive of scientific journals based on Journal Article Tag Suite regardless of their languages," *Biochemia medica*, 23(3), 2013, p.235-236.

[4] B. C. Björk, P. Welling, M. Laakso, P. Majlender, T. Hedlund and G. Guðnason, "Open access to the scientific journal literature: situation 2009," *PloS one*, 5(6), 2010, e11273.

[5] A. Constantin, S. Pettifer, and A. Voronkov, "PDFX: fully-automated PDF-to-XML conversion of scientific literature," *Proceedings of the 2013 ACM symposium on Document engineering*, ACM, 2013, p.177-180.

[6] R. Kern, K. Jack, M. Hristakeva and M. Granitzer, "TeamBeam Meta-Data Extraction from Scientific Literature," *D-Lib Magazine*, 18(7), 2012, 1.

[7] M. T. Luong, T. D. Nguyen and M. Y. Kan, "Logical structure recovery in scholarly articles with rich document features," *International Journal of Digital Library Systems (IJDLS)*, 1(4), 2010, p.1-23.

[8] C. Ramakrishnan, A. Patnia, E. H. Hovy and G. A. Burns, "Layout-aware text extraction from full-text PDF of scientific articles," *Source code for biology and medicine*, 7(1), 2012, 7.

[9] M. Granitzer, M. Hristakeva, R. Knight, K. Jack, and R. Kern., "A comparison of layout based bibliographic metadata extraction techniques," in *Proc. the 2nd International Conference on Web Intelligence, Mining and Semantics*. ACM, June, 2012, p. 19.

[10] F. Peng, and A. McCallum, "Information extraction from research papers using conditional random fields," *Information Processing & Management*, 42(4), 2006, p.963-979.

[11] S. Klampfl and R. Kern, "An Unsupervised Machine Learning Approach to Body Text and Table of Contents Extraction from Digital Scientific Articles," *Research and Advanced Technology for Digital Libraries*, Springer Berlin Heidelberg, 2013, p. 144-155.

[12] H. Han, C. L . Giles, E. Manavoglu, H. Zha, Z. Zhang and E. A. Fox, "Automatic document metadata extraction using support vector machines," *Digital Libraries, Joint Conference on*. IEEE, 2003, p. 37-48.

[13] J. Chen and H Chen. "A Structured Information Extraction Algorithm for Scientific Papers based on Feature Rules Learning," *Journal of Software (1796217X)*, 8(1), 2013.

[14] D. Tkaczyk, L. Bolikowski, A. Czeczko and K. Rusek, "A modular metadata extraction system for born-digital articles.," *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*, IEEE, 2012, p. 11-16.

[15] J. Zhao and H. Liu, "Metadata Extraction Approach of PDF Documents Based on Measurement Fusion," *Journal of Multimedia*, 8(6), 2013

[16] P. A. Praczyk and J. Nogueras-Iso, "Automatic extraction of figures from scientific publications in high-energy physics," Information Technology and Libraries, 32(4), 2013, p. 25-52.

[17] G. Eysenbach, "Citation advantage of open access articles," PLoS biology, 4(5), 2006, e157.

[18] K. Antelman, "Do open-access articles have a greater research impact," College & research libraries, 65(5), 2004, p.372-382.

[19] I. G. Councill, C. L. Giles, E. Di Iorio, M. Gori, M. Maggini and A. Pucci, "Towards next generation citeseer: A flexible architecture for digital library deployment," *Research and Advanced Technology for Digital Libraries*, Springer Berlin Heidelberg, 2006, p.111-122.

[20] Z. Fuzhi, and Z. Zhao, "A Metadata Extraction Approach from Papers Based on Meta-learning," 2013.

[21] B. Ojokoh, M. Zhang and J. Tang, "A trigram hidden Markov model for metadata extraction from heterogeneous references," *Information Sciences* 181(9), p.1538-1551.

[22] J. Lafferty, A. McCallum and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[23] J. Beck, "NISO Z39. 96The Journal Article Tag Suite (JATS): What Happened to the NLM DTDs?," *The journal of electronic publishing: JEP*, 14(1), 2011.

[24] P. Flynn, L. Zhou, K. Maly, S. Zeil and M. Zubair, "Automated template-based metadata extraction architecture," *Asian Digital Libraries, Looking Back 10 Years and Forging New Frontiers*, Springer Berlin Heidelberg, 2007, p.327-336.

The Authorship of Audacity: Data Mining and Stylometric Analysis of Barack Obama Speeches

Jonathan Herz
School of Engineering and
Applied Science
George Washington University
Washington, DC 20052
Email: jonathanherz@gwmail.gwu.edu

Abdelghani Bellaachia
School of Engineering and
Applied Science
George Washington University
Washington, DC 20052
Email: bell@gwu.edu

***Abstract*—We explore the feasibility of identifying authorship among President Obama's principal speechwriters with the interesting result that, yes, we can. This task is difficult because there are few training examples, multiple authors (four), and because these authors consciously attempt to emulate a single style - the President's. Four corpuses are created to compare different text pre-processing techniques. On each, function word frequencies are analyzed with ANOVA to select discriminating feature vectors. Using leave-one-out cross-validation, K-nearest neighbors achieves the best classification accuracy, 78%. One interesting result is that the new head White House lead speechwriter, Cody Keenan, is not distinguishable from the other principal speechwriters beyond pure chance. Classification accuracy is improved to 90% after removing his work from our corpus.**

Key words: authorship attribution, text mining, classification, statistical analysis, stylometry.

I. Introduction

A. Stylometry

Stylometry is the study of identifying authorship of texts based on information contained in the text itself. Many methods have been proposed, but they all share the assumption that every author leaves behind quantifiable and distinctive markers that allow them to be identified from a pool of possible authors [1]. Some of the more popular features that have been used in the problem domain include various types of word frequencies[2, 3], n-gram frequencies, punctuation[4], and aggregate measurements such as average sentence length [5]. Unlike many text classification problems, authorship identification uses function word frequencies as feature vectors rather than ignoring them as stop words.

B. The Data

We assembled a corpus of 37 speeches and addresses delivered by President Obama for which we can assign primary authorship to one of four principal speechwriters. These speeches span both his time as candidate and sitting president. One of the challenges in this experiment was finding attributions of speeches, since presidential speechwriters are, in the words of FDR, expected to "have a passion for anonymity."

The four principal speechwriters are Jon Favreau, Cody Keenan, Adam Frankel, and Ben Rhodes. Jon Favreau was Obama's lead speechwriter from 2005 - 2013, and has the most speeches in the corpus attributed to him as primary author: 13 in total. Cody Keenan, the current lead speechwriter, joined the team in 2009. Five of the speeches in our corpus are attributed to him. Adam Frankel also worked on the 2008 campaign, and has 9 speeches attributed in our training set. Ben Rhodes, who wrote for the campaign and now specializes in foreign policy speeches, has 10 attributions.

C. Obama's Speechwriters

In this paper, we will examine the President Barack Obama's national speeches and remarks to attempt to determine which speeches were written by which of his principal speechwriters. It is too early to say with certainty what Obama's place in history will be, but it is safe to say that he is a politician whose career was built on the strength of his oratory. Obama catapulted to the national stage with his address to the 2004 Democratic National Convention, before he had even won national office. His viability as a presidential candidate only four years later can only be explained by the strength of that speech, and the widespread national attention that it garnered. Although a gifted orator in his own right, even Obama has had plenty of help from others. Beginning with the start of his Senate career in 2009, Obama began to assemble a speechwriting team that would follow him to the White House and beyond.

Obama has four principal speechwriters with identifiable speeches: Jon Favreau, Cody Keenan, Ben Rhodes, and Adam Frankel. They are, as a group, remarkably young for presidential speechwriters. After five years of the Obama administration, most are now in their late 20s or early 30s. They joined the team at different times, and some have recently left the administration. Unfortunately, information about the specific speeches these writers worked on is very sparse. Out of hundreds of speeches delivered by Barack Obama over the course of his national political career, we were only able to attribute 37 to individual speechwriters through interviews available in the public domain.

Jon Favreau was Obama's lead speechwriter until early 2013. In 2005, Favreau began working for then-Senator Obama as his speechwriter. Some of President Obama's most defining speeches were written in conjunction with Jon

Favreau. After the Jeremiah Wright controversy engulfed the 2008 Obama presidential campaign, Jon Favreau worked on the key speech "A More Perfect Union," which began by contextualizing the remarks of that controversial preacher against the backdrop of race relations in the United States, and ended with a call to work together to address social problems[6]. The speech became popular very quickly on YouTube, with 1.2 million views in the first 24 hours after release. Favreau also worked on President Obama's "Nobel Peace Prize acceptance speech"[6]. Other important speeches that were identified as Favreau's work include both inaugural speeches[6,7], the "2008 Jefferson-Jackson Day Dinner address"[8], and, more recently, the President's "remarks at the prayer vigil for victims of the Sandy Hook Elementary school shooting"[6].

Cody Keenan joined the presidential speechwriting team in 2009, and succeeded Jon Favreau as the new Director of Speechwriting in 2013. Keenan wrote the President's "eulogy for Senator Ted Kennedy"[9], and the president's "remarks upon the signing of the 'Edward M. Kennedy Serve America Act"'[9]. Keenan also worked on the "2009 White House Correspondents' Association dinner remarks"[10], the President's "speech at the Tucson shooting memorial service"[10], and "the 2013 State of the Union Address"[10].

Ben Rhodes' official job title is deputy national security adviser for strategic communications. His speeches focus on foreign policy. Speeches that have been attributed in the press to Rhodes include the New Beginning speech made by Obama in June 2009 in Egypt as an attempt to improve America's image in the Middle East[11], a speech given to a large public audience in Israel in March 2013[12], a February 2009 speech on ending the Iraq War[11], a "speech to the Ghanaian Parliament in July 2009"[11], and a "speech delivered to the New Economic School in Moscow"[11].

Adam Frankel, like Jon Favreau, also started in the Kerry campaign. He was Favreau's first hire in 2007 after Obama announced his candidacy for the presidency, but left in 2011. Speeches that are known to have been written by him include the President's "eulogy for Senator Robert C. Byrd"[13], the President's "address to the Upper Big Branch coal mine disaster memorial service"[13], the President's "address at the National Peace Officers' memorial"[13], the President's "eulogy for civil rights activist Dr. Dorothy Height"[13], and the President's 2009 speech to the American Medical Association[13]. He also prepared remarks for the president at a National Prayer Breakfast after his first inauguration[8].

II. Related Work

Mosteller and Wallace pioneered the modern study of stylometry with their work on the Federalist Papers[2]. "Mosteller and Wallace employed numerical probabilities to express degrees of belief about propositions such as 'Hamilton wrote paper No. 52' and then used Bayes' theorem to adjust these probabilities for the evidence in hand. They attributed all 12 disputed papers to Madison, a conclusion broadly in agreement with historical scholarship" [1]. Mosteller and Wallace used a Naive Bayes classifier for the classification task, but many other algorithms have been used with success, for example: Naive Bayes, neural nets, linear discriminant analysis[4], and support vector machines [3,5]. Beginning in the 1980s, J. F. Burrows began to explore the use stylometric techniques to analyze Jane Austen's novels and other English authors of the Victorian era. Like Mosteller and Wallace, Burrows analyses used frequencies of function words such as by, the, from, etc. as discriminant features to classify authorial style. Burrows even used stylometric techniques to study character dialogue in some of Jane Austen's books to identify the stylometric fingerprint of different fictional characters. Burrows' primary contribution to the field was the pioneering use of Principal Component Analysis to reduce the high dimensionality inherent in using the frequencies of dozens of function words to categorize text[14]. Principal Component Analysis takes a correlation or covariance matrix of the dimensions being used in the analysis in order to find the highest eigenvalues, which correspond to the most significant variables. Usually, the marginal amount of variance explained by the dimensions or variables after the first few drops off dramatically.

The combination of feature selection of function words introduced by Mosteller and Wallace, and the generalization of that approach to multivariate methods using Principal Component Analysis for dimensional reduction by Burrows, set a standard that has been followed by many stylometric studies. For example, In the paper Who wrote the 15th Book of Oz, Jose Binongo uses exactly the approach described above to determine whether popular sequels to The Wizard of Oz by Frank Baum had been written by Baum himself or by Ruth Plumly Thompson, an associate of Baum's who is known to have written many of the later Oz books[15]. First, like Mosteller and Wallace, Binongo tallied the frequency of function words. Next, like Burrows' work, the high-dimensional 50 most frequently occurring words were reduced to 2 using Principal Component Analysis.

Similar methods have also been used to attribute presidential addresses. In the 2006 paper Who Wrote Ronald Reagan's Radio Addresses?, Airoldi, Anderson, Fienberg, and Skinner use function word frequency counts, Principal Component Analysis, and other methods to attempt to determine authorship of several hundred of Ronald Reagan's radio addresses that were delivered in the late 1970s, as Reagan prepared to run for the White House[3]. The Reagan study also tried many different methods of stylistic fingerprinting in addition to function word frequencies, including delta-squared measures, SMART list words, semantic features, information gain, and n-gram methods. In all cases, Principal Component Analysis was used for dimensionality reduction. The Reagan radio address paper used several methods for classification as well. Naive Bayes, LDA, Logistic Regression, Unit-Weight Models, SVM, k-Nearest Neighbors, CART, Random Forests, Majority Voting, and Maximum Likelihood approaches were all tried. The authors settled on using Naive Bayes to classify word features selected by the delta-squared measure.

The Reagan radio address study shares many of the problems inherent in our study; the line between a president's words and his aides is often unclear, the number of addresses known to be attributed to particular aides is small relative to the number of addresses of unknown origin, and there is no way to know for certain whether or not our predictions for authorship are accurate. The Reagan study has the advantage of being written over a decade after Ronald Reagan's presidency ended. The intervening years give time for aides

to report what they wrote, for historians and archivists to examine the documents produced by the administration, and for criticism, praise, and attribution of a President's speeches and addresses delivered for purely political reasons to recede behind the curtain of history and rational analysis.

While, to our knowledge, there have been no other studies of authorship attribution applied specifically to Obama's speeches, Savoy studied the use of classification techniques including support vector machines (SVM) and Nave Bayesian classifiers to identify speeches belonging delivered on the campaign trail as Candidate Obama, and those delivered while in office as President Obama[16]. While that study shares many similarities to ours, as it also uses individual word frequencies as feature vectors to a classification problem, it focuses on contextual words that reveal the subject matter of the speech. Our study, which attempts to solve a slightly different problem, restricts itself to function words that have little or no contextual value.

A relatively recent branch of stylometric research has interesting implications for our experiment. Adversarial stylometry attempts to measure the ability of authors to preserve their anonymity by consciously obfuscating their own writing style, often by emulating another authors style. Brennan, Afroz and Greenstadts 2012 study used a pool of volunteers who were able to reduce the accuracy of several popular stylometric classification techniques to the level of random chance[5]. Obama's speechwriters, although not participating in a study, do try to speak as one voice that of the President of the United States. Obama is known for having a distinctive rhetorical style, and it is the job of his speechwriters to try to emulate it.

III. Experimental Approach and Results

A. Pre-Processing

Our raw corpus of speeches consisted of the 37 major speeches whose authorship we could verify.

Four different pre-processing approaches were used on this collection of speeches. In each case, we followed standard text processing steps of removing numbers, removing punctuation, converting to lower case. At this point, in many studies, the corpus is stemmed. Stemming is a process by which words with the same root such as happy and happiness are converted, or stemmed, into their same common root. For English language documents, the Porter Stemming algorithm is the most commonly used algorithm for the task. However, we were concerned that stemming the corpus could remove some stylometric markers, if one author used certain verb forms or tenses more often than others. So, we performed all of our experiments on both a stemmed and an unstemmed corpus.

After these pre-processing steps were completed, we combined all of the stemmed words and documents of the corpus into one Document Term Matrix. At this point, there are many options for normalizing term and document frequencies. For many text mining applications, a TF-IDF (for Term Frequency, Inverse Document Frequency) matrix is used. This kind of matrix brings out the words most likely to capture the semantic essence of each document by finding the words that appear most often in individual documents (the term frequency) but not in all documents (inverse document frequency). For the purposes of our stylometric analysis, we were primarily interested in the words that appear most frequently in each speech, as these are most likely to be the function words. We did not use any document weighting whatsoever, since most document weighting schemes are designed specifically to reduce the weight of function words in the term-document matrix. We tried two term frequency weighting schemes. One was augmented term frequency, which allowed us to normalize word frequencies between large and small speeches. Augmented term frequency, originally introduced by the SMART information retrieval system is:

$$0.5 + \frac{0.5 * tf_{t,d}}{max(tf_{t,d})}$$

Where $tf_{t,d}$ is the raw term frequency of term t in document d.

The other term frequency weighting scheme we used was a simple normalization,

$$tf_{t,d}/T_d$$

where $tf_{t,d}$ is the raw term frequency of term t in document d and T is the total number of terms in document d.

If we had used raw word counts, of course, very long speeches would have much greater weight than shorter ones, and our final results might have simply clustered speeches by length rather than by the stylistic fingerprints we hoped to uncover.

To summarize, the four pre-processing methods used were: Stemmed / Augmented term weighting, Unstemmed / Augmented term weighting, Stemmed / Normalized term weighting, and Unstemmed / Normalized term weighting. These will be referred to as the S/A, U/A, S/N, and U/N sets, respectively. All experiments were performed on each of the four resulting data sets.

B. Feature Selection

For each of the preprocessed sets, individual analysis of variance (ANOVA) was conducted where either the augmented or normalized term frequency was the response variable and the speechwriter was the classification variable. Table 1 below summarizes the number of statistically significant $(p < 0.05)$ ANOVA results in other words, the number of function words whose between-speechwriter frequency distributions were least likely to differ from each other from chance and the 10 most significant discriminators (lowest p-value) for each of the 4 pre-processing groups. It is not surprising that unstemmed data sets should have more discriminating function word frequencies, as there are more words to choose from since the stemmer summarizes different versions of the same root as a single word.

Since the feature lists were restricted to function words only (using the well-known Snowball stemmers stop word list as a list of function words), it is hard to imagine how many of the words, such as "has", "got", "was", that were selected using the ANOVA tests outlined above could have any contextual meaning whatsoever. Some other words, such as "youngest" or "she" could possibly be found to have some contextual

Augmented Stemmed (28 total)		Augmented Unstemmed (35 total)		Normalized Stemmed (33 total)		Normalized Unstemmed (48 total)	
word	p	word	p	word	p	word	p
between	.0008	does	.0005	has	.0001	has	.0001
cannot	.0009	between	.0008	must	.0001	must	.0001
youngest	.0016	cannot	.0009	have	.0001	have	.0001
area	.0024	youngest	.0016	clear	.0001	clear	.0001
was	.0045	areas	.0042	will	.0002	will	.0002
has	.0058	was	.0045	end	.0007	areas	.0004
use	.0112	right	.0056	younger	.0008	find	.0006
end	.0119	has	.0058	got	.0011	parts	.0006
she	.0126	use	.0090	was	.0011	younger	.0008
just	.0129	she	.0126	just	.0013	end	.0009

TABLE I. LOWEST 10 P-VALUES FOR EACH SET

value, but it is more likely that such word would be common in several different categories of speeches.

C. classification

Four different off-the-shelf classification methods were explored: Nave Bayes, K-nearest neighbors on the projections of Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Feed-Forward Neural Networks. Each classification method was used with all possible combinations of relevant parameters (for example, k for K-nearest neighbors). For each classifier, we chose to use leave-one-out cross-validation was used to calculate classifier accuracy due to the very small size of the training set only 37 speeches. In leave-one-out cross-validation, for each speech, a classifier is built using the other 36 speeches, and the speechwriter for that speech is predicted using that classifier. The three non-linear classifiers, Nave Bayes, PCA+Knn, and Neural Nets all achieved accuracy ratings around 60-70%, while LDA achieved no better than random chance on some of the pre-processing sets. These results make intuitive sense; one would not expect a high-dimensional, messy data set of political speeches to be easily linearly separable. The best cross-validated for each classifier on each pre-processing set is shown on Table 2.

LOOCV Accuracy	Augm. Stem.	Augm. Unst.	Norm. Stem.	Norm. Unst.
Naive Bayes	57%	62%	70%	76%
PCA + Knn	70%	78%	68%	70%
LDA	57%	32%	24%	43%
Neural Nets	60%	70%	65%	68%

TABLE II. LOOCV ACCURACY FOR EACH PRE-PROCESSING SET, FOR EACH CLASSIFICATION ALGORITHM

Naive Bayes classifiers are commonly used for a wide range of classification problems, like e-mail spam detection. Nave Bayes classifiers assign category labels using Bayes theorem with the simplifying assumption that all observations are independent (hence the naivete in Nave Bayes). For each pre-processing set, we used the function words and their frequencies from our analysis of variance as feature vectors to train Nave Bayes classifiers. The leave-one-out cross-validation prediction confusion matrices are shown below. Each of these predictions was made by creating a classifier using all of the other speeches in the training set (37-1 = 36), and then using the data from the class label being predicted as a test set.

		Predicted Author			
		Favreau	Keenan	Rhodes	Frankel
Actual Author	Favreau	6	1	6	0
	Keenan	2	2	1	0
	Rhodes	1	0	9	0
	Frankel	3	2	0	4

Augmented, Stemmed Naive Bayes Cross-Validation Predictions (56.8% accuracy)

		Predicted Author			
		Favreau	Keenan	Rhodes	Frankel
Actual Author	Favreau	5	3	5	0
	Keenan	2	3	0	0
	Rhodes	1	0	9	0
	Frankel	1	2	0	6

Augmented, Unstemmed Naive Bayes Cross-Validation Predictions (62.2% accuracy)

		Predicted Author			
		Favreau	Keenan	Rhodes	Frankel
Actual Author	Favreau	11	0	0	2
	Keenan	2	0	1	2
	Rhodes	3	0	7	0
	Frankel	1	0	0	8

Normalized, Stemmed Naive Bayes Cross-Validation Predictions (70.3% accuracy)

		Predicted Author			
		Favreau	Keenan	Rhodes	Frankel
Actual Author	Favreau	11	0	0	2
	Keenan	3	0	0	2
	Rhodes	1	0	9	0
	Frankel	1	0	0	8

Normalized, Unstemmed Naive Bayes Cross-Validation Predictions (75.7% accuracy)

TABLE III. CONFUSION MATRICES FOR EACH PRE-PROCESSING SET

Since most text data, including stylometric data, is highly dimensional, it makes sense to try techniques to reduce the dimensionality of the data. Principal Component Analysis (PCA) is a method that summarizes high-dimensional data by finding the eigenvectors of covariance matrices of the original feature vectors. The first principal component summarizes the data in the direction of the highest variance, and each successive principal component summarizes the highest amount of variance possible subject to the constraint that it is orthogonal with preceding components. Figure 1 shows the projections of the first two principal components for each pre-processing set. Just from this simple visualization, there does appear to be some degree of authorial structure a conclusion borne out in our cross-validation prediction accuracy of about 70% for most of the pre-processing groups.

While some stylometric studies have used Linear Discriminant Analysis, or LDA, with success[5], in our experiment they produced the highest error rate of any of the classifiers that we tried. For the normalized, stemmed pre-processing set, LDA achieved the same performance as random guessing: 25%. These results confirm what our visualizations above suggest the data are not linearly separable, and the other 3 classifiers used, which are all capable of modeling complex decision boundaries, had much higher cross-validated prediction accuracy.

Authorial structure was also evident using neural nets. We used feed-forward neural nets with back-progagation using a single hidden layer. We tried different numbers of hidden neurons, from 1 to 36 for each pre-processing set. In each case, cross-validated prediction accuracy did not vary by more than 1 or 2 correct classifications for 4 or more hidden

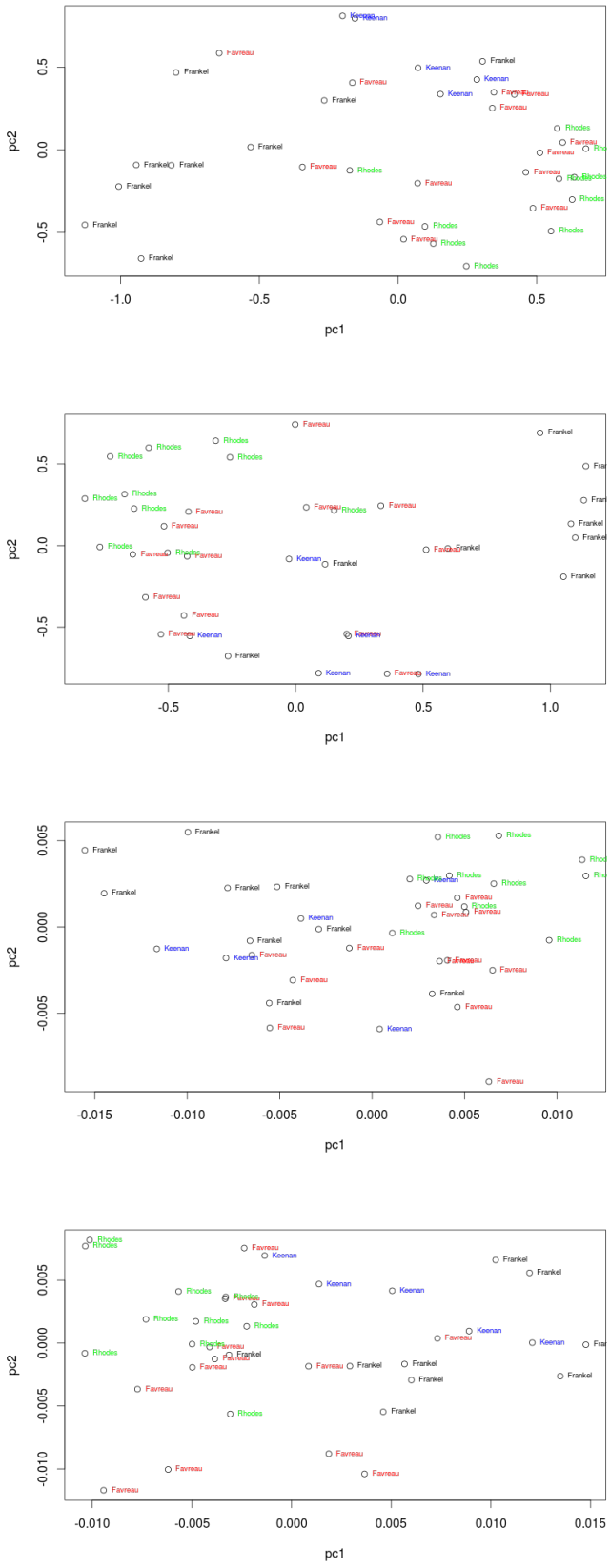


Fig. 1. Projections of first two principal components for each set. From top: augmented, stemmed; augmented, unstemmed; normalized, stemmed; normalized, unstemmed

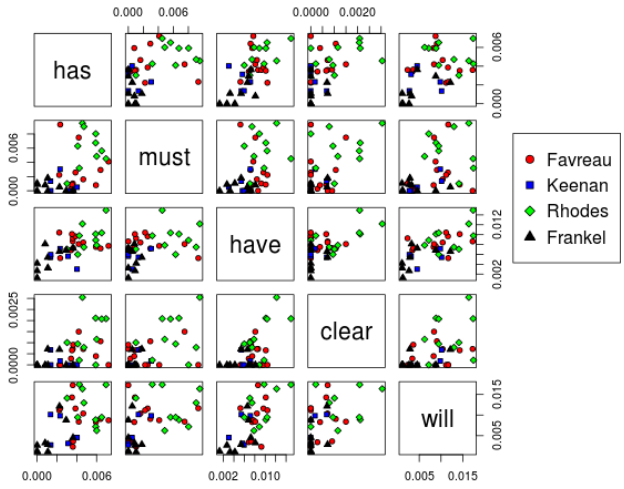


Fig. 2. Normalized, unstemmed scatterplot of top 5 most relevant features. Different colors correspond to different authors. Not lack of linear separability, though complex structure is apparent in the data

neurons. Neural nets have several advantages, all of which apply to our authorship study of Obama's speeches[17]: they can adaptively learn from the data itself, they can generalize on a limited training set, they are good at capturing non-linear interactions between input variables, and they are tolerant of individual data points that exhibit some unusual characteristics compared to other members of the set.

In the course of our experiments, we discovered that one author's work was consistently more difficult to identify than his peers: Cody Keenan (Table 4). Cody Keenan served as an assistant speechwriter during the period of time that the other three were active, so it is possible that the speeches attributed to him may have been more collaborative than the speeches written by the more senior speechwriters. Furthermore, we also have the fewest number of speeches - 5 - for Mr. Keenan out of any of the authors, so it is also possible that our sample size is simply too small to accurately evaluate the distinguishing characteristics of his writing style.

Removing Mr. Keenan's speeches from our corpus

LOOCV Accuracy	Augm. Stem.	Augm. Unst.	Norm. Stem.	Norm. Unst.
Favreau	46%	38%	85%	85%
Keenan	40%	60%	0%	0%
Rhodes	90%	90%	70%	90%
Frankel	44%	67%	89%	89%

LOOCV Accuracy	Augm. Stem.	Augm. Unst.	Norm. Stem.	Norm. Unst.
Favreau	62%	54%	85%	77%
Keenan	20%	20%	0%	20%
Rhodes	70%	80%	60%	60%
Frankel	50%	89%	67%	78%

TABLE IV. LOOCV ACCURACIES FOR INDIVIDUAL AUTHORS FOR NAIVE BAYES (TOP) AND NEURAL NETS (BOTTOM) CLASSIFIERS

improved accuracy significantly. The results for all pre-processing techniques and classification methods without

Keenan speeches are in Table 5. Intuitively, since Keenan was an assistant speechwriter for much of his tenure, we speculate that his speeches were edited by the other authors, confusing the issue of authorship. The fact that there are so few speeches attributed to him, and that he did not take an active role in the presidential campaign, suggests that his role in presidential speechwriting was quite limited compared to the other 3 authors. It is known that some speeches are more or less collaborative efforts[8], although principal authors are usually identified. In the case of Mr. Keenan's speeches, it would seem that his speeches were much more collaborative. To show how much clarity of results was improved by removing Keenan's speeches from our data set, we also present confusion matrices for each pre-processing set for Naive Bayes classification in Table 6. The difference is striking.

LOOCV Accuracy	Augm. Stem.	Augm. Unst.	Norm. Stem.	Norm. Unst.
Naive Bayes	72%	72%	84%	91%
PCA + Knn	91%	81%	84%	81%
LDA	70%	80%	60%	60%
Neural Nets	78%	84%	75%	75%

TABLE V. RESULTS WITHOUT KEENAN

		Predicted Author		
		Favreau	Rhodes	Frankel
Actual Author	Favreau	9	3	1
	Rhodes	1	9	0
	Frankel	4	0	5

Augmented, Stemmed Naive Bayes Cross-Validation Predictions (71.9% accuracy)

		Predicted Author		
		Favreau	Rhodes	Frankel
Actual Author	Favreau	7	5	1
	Rhodes	1	9	0
	Frankel	2	0	7

Augmented, Unstemmed Naive Bayes Cross-Validation Predictions (71.9% accuracy)

		Predicted Author		
		Favreau	Rhodes	Frankel
Actual Author	Favreau	12	0	1
	Rhodes	3	7	0
	Frankel	1	0	8

Normalized, Stemmed Naive Bayes Cross-Validation Predictions (84.4% accuracy)

		Predicted Author		
		Favreau	Rhodes	Frankel
Actual Author	Favreau	13	0	0
	Rhodes	2	8	0
	Frankel	1	0	8

Normalized, Unstemmed Naive Bayes Cross-Validation Predictions (90.6% accuracy)

TABLE VI. CONFUSION MATRICES AFTER EXCLUDING KEENAN SPEECHES

IV. CONCLUSION

Our study shows evidence that authors can still be identified when they consciously attempt to speak in a unified voice, although with somewhat diminished accuracy. Similar approaches to the ones presented here yielded superior accuracy in [2,3,15,17], but we did not experience the same degradation in accuracy as the adversarial stylometry experiment conducted in [3]. Although the individual speechwriters are not attempting to specifically defeat stylometric analyses, they are attempting to achieve a consistent rhetorical tone that is recognizably President Obama's. Together, their writings are the voice of the President of the United States, but individually, presidential speechwriters are expected to avoid excessive publicity. As Franklin Roosevelt put it, they are expected to have a passion for anonymity. These results suggest that stylometry can, in fact, be used to differentiate authors who are actively attempting to write in a similar style.

Our findings also suggest that avoiding the use of stemming algorithms during text pre-processing can increase the accuracy of stylometric techniques. With the exception of linear discriminant analysis, each of our classification techniques performed better on the unstemmed data sets than on their corresponding stemmed data sets. This finding provides further evidence that word forms can be a source of stylistic discrimination, and therefore should not be discarded.

V. REFERENCES

1. Holmes, D. and Kardos, J. (2003), "Who Was the Author? An Introduction to Stylometry". *Chance*, vol. 16, no. 2, pp. 5-8.

2. Mosteller, F. and Wallace, D. (1964), "Applied Bayesian and Classical Inference: The Case of the Federalist Papers," Reading MA: Addison-Wesley.

3. Airoldi, E., Anderson, A., Fienberg, S., and Skinner, K. (2006), "Who Wrote Ronald Reagan's Radio Addresses?" *Bayesian Analysis*, vol. 1, no. 2., pp. 289-320.

4. Baayen, R.H., Van Halteren, H., Nejit, A., and Tweedie, F. (2002) "An Experiment in Authorship Attribution." In *Proceedings of the 6th International Conference on the Statistical Analysis of Textual Data* (JADT 2002).

5. Brennan, M., Afroz, S., and Greenstadt, R. (2011), "Adversarial Stylometry: Circumventing Authorship Recognition to Preserve Privacy and Anonymity." *ACM Trans. Info. Syst.* Sec. 15, 3, Article 12, 22 pages.

6. Peterson, H. (2013, Feb. 28), "Outgoing speechwriter reveals how Obama casually asked him to spike a bin Laden joke on the night of the Pakistan raid as he reflects on his 8 years with the president." *The Daily Mail* [Online]. Available at: www.dailymail.co.uk/news/article-2286126/Jon-Favreau-reveals-Obama-asked-spike-Osama-bin-Laden-joke-Pakistan-raid.html

7. Stein, S. (2013, Jan. 24), "Obama's Inaugural Address 'One of the Hardest Speeches I've Written,' Jon Favreau Says." *Huffington Post* [Online]. Available at: www.huffingtonpost.com/2013/01/24/obamas-inaugural-address-jon-favreau_n_2545236.html

8. Berry, M., and Gottheimer, J. (2010), "Power in Words: the Stories Behind Barack Obama's Speeches." Boston, MA: Beacon Press.

9. Kohut, M. (2010, Jan. 11), "Alumnus Cody Keenan MPP 2008: White House Wordsmith." *Harvard Kennedy School Alumni Stories* [Online]. Available at:

www.hks.harvard.edu/news-events/news/alumni/alum-cody-keenan-jan10

10. Franke-Ruta, G. (2013, Feb. 12), "Who is Cody Keenan, Obama's SOTU Speechwriter?" *the Atlantic* [Online]. Available at: www.theatlantic.com/politics/archive/2013/02/who-is-cody-keenan-obamas-sotu-speechwriter/272938/

11. Brotzen, F. (2009, Aug. 6), "Rice alum helps craft Obama's international message." *Rice University News* [Online]. Available at: news.rice.edu/2009/08/06/rice-alum-helps-craft-obamas-international-message/

12. Landler, M. (2013, Mar. 15), "Worldly at 35, and Shaping Obama's Voice." *the New York Times* [Online]. Available at: www.nytimes.com/2013/03/16/world/middleeast/benjamin-rhodes-obamas-voice-helps-shape-policy.html

13. Nicholas, P. (2010, Sep. 3), "The brain behind Obama's speeches." *The Los Angeles Times* [Online]. Available at: articles.latimes.com/2010/sep/03/nation/la-na-speechwriter-20100903

14. Burrows, J.F. (1992), "Not Unless You Ask Nicely: The Interpretative Nexus Between Analysis and Information," *Literary and Linguistic Computing*, vol. 7, pp. 91-109.

15. Binongo, J. (2003), "Who Wrote the 15th Book of Oz? An Application of Multivariate Analysis to Authorship Attribution." *Chance*, vol. 16, no. 2, pp. 9-17.

16. Savoy, J., and Zubaryeva, O. (2011). "Classification based on Specific Vocabulary." In *2011 IEEE/WIC/ACM International Confences on Web Intelligence and Intelligent Agent Technology*.

17. Tweedie, F. J., Singh, S., and Holmes, D. I. (1996), "Neural Network Applications in Stylometry: the Federalist Papers." *Computers and the Humanities* vol 30. pp. 1-10.

Text categorization using topic model and ontology networks

Yinghao Huang, Xipeng Wang and Yi Lu Murphey, *Member, IEEE*

***Abstract*— Text categorization based on pre-defined document categories is one of the most crucial tasks in text mining applications in recent decades. Successful text categorization highly relies on the text representations generated from documents. In this paper, an innovative text categorization model, VSM_WN_TM, is presented. VSM_WN_TM is a special Vector Space Model (VSM) that incorporates word frequencies, ontology networks and latent semantic information. Unlike the traditional text representation using only Bag-of-words (BOW) features, it incorporates semantic and syntactic relationship among words such as synonymy, co-occurrence and context, with the purpose of providing more inclusive and accurate text representation. Support Vector Machine is used as document classifier, and the proposed system is evaluated on three publicly available datasets and one domain-specific dataset. Experiment result shows that our approach significantly improves text classification by outperforming approaches such as using only latent features and traditional VSM approaches.**

Keywords*— *Text Categorization, vector space model, ontology network, topic model, support vector machine.

I. INTRODUCTION

AS the exponentially increasing amount of digital text documents in recent decades, text categorization has become one of the most important tasks in text mining applications that have been drawing attention for quite a long time. Especially in the age of "Big Data", daily load of text document processing brings more and more necessity for accurate and efficient text categorization.

Automatic text categorization process, generally speaking, is the task of building a classifier that assign a pre-defined category to each unknown document. More formally, as described in [1], if D is a set of documents, $D = \{d_1, d_2, ..., d_N\}$, and C is a set of predefined categories, $C = \{c_1, c_2, ..., c_M\}$, the task is to approximate the classifier that maps each $d_i \in D$ to a $c_j \in C$, so that the estimated target mapping function $\hat{\phi} : D \to C$ coincide the real mapping function $\phi : D \to C$ as much as possible.

Document category is usually defined based on various application requirements, such as the topic a news article discusses, the importance of a vehicle diagnostic record that describes vehicle repair details [3], the fact stated in a medical diagnostic document that whether or not an injury condition sustains [2], etc. As a result, supervised machine learning algorithms are often applied to learn underlying patterns that connect documents to their assigned categories, so that the classification model can be trained and provides category prediction for unknown documents.

There are already plenty of research works that focus on developing and improving machine learning techniques adopted for building classification models, such as applying k-nearest-neighbor classifiers [8,12], Naïve Bayes classifiers [5,11], Self Organizing Maps [2,10], Neural Network classifiers [6,9,13] and Support Vector Machine classifiers [4,7]. By using these techniques, promising text classification performance could be obtained. However, whatever machine learning techniques are used to build these classifiers, the classification accuracy will be "bottlenecked" if the representation quality of the document is poor, i.e., the representation of a document does not reflect close relationship with its assigned category. For example, the typical and mostly used approach of representing a text document is the Vector Space Model (VSM) [14], in which each document is represented by a weighted vector that provide a more convenient way for computation and analysis. However, this approach does not consider the semantic relationship between words, such as synonyms, hyponyms, etc. The classification accuracy is especially deteriorated in the document set that different category has similar word occurrence [2,15]. As a result, a text categorization model that captures underlying semantic and syntactic information besides single word occurrence features is of great necessity.

Considering the above fact, researchers have been working on improving document representations for classification, by altering the VSM model. Some typical approach include applying different weighting scheme to each term [25], adding semantic information as additional features, by capturing word co-occurrence, such as [16], or building word ontology as external knowledge that provides synonym information, such as [17,18,19]. Note here, word ontology is defined as specification of a representational vocabulary for a shared domain of discourse that describes word relationships, according to [20]. However, this requires a lot of manual or semi-manual work to build word ontology from scratch, and this ontology is usually limited by domain-specific corpus. Therefore, to a large extent, current research work mainly focuses on learning ontologies from existing resources such as English lexicons [24], Furthermore, most of the state-of-art approaches use only nouns for ontology building, and a large extent of methods aims at constructing IS-A-related concept hierarchies. [21-23].

Yinghao Huang is with the Electrical and Computer Engineering Department, University of Michigan – Dearborn, Dearborn, MI 48128. (E-mail: yinghaoh85@gmail.com.)

Xipeng Wang is with the Electrical and Computer Engineering Department, University of Michigan – Dearborn, Dearborn, MI 48128. (E-mail: xipengw1990@gmail.com.)

Prof. Yi Lu Murphey is with the Electrical and Computer Engineering Department,University of Michigan - Dearborn, Dearborn, MI 48128. (E-mail: yilu@umich.edu.)

The above approaches, in fact, still only consider relationship between words instead of underlying semantic meanings. More advanced research about digging out latent semantic components could be referred to topic modeling, which is typically used for unsupervised text clustering, information retrieval and dimension reduction, such as Latent Semantic Indexing (LSI) [26], probabilistic latent semantic analysis (PLSA) [27], and Latent Dirchilet Allocation (LDA) [28]. These language models aim at estimating latent semantic components that are mapped from the word space. However, most of the work been done on applying topic model to text categorization is to purely use estimated latent features for classification, such as [15,29,30,31], which is claimed in [32] to be less accurate than using bag-of-word (BOW) features, especially when training data size is large.

In this paper, we present an inventive text categorization approach that focuses on augmenting VSM model with both ontology network and topic model. We first modify the original BOW representation by using an innovative global weight scheme, and then combine the weighted BOW matrix with a synset matrix based on WordNet ontology, and also with a latent topic matrix generated from PLSA model. Our system is evaluated using SVM classifier, on three publicly available datasets and a domain-specific dataset.

The remainder of the paper is organized as follows: Section 2 details our text categorization system, Section 3 presents the case study and discusses the empirical results, and Section 4 concludes the paper.

II. METHODOLOGY

In this section, we present the proposed text categorization system and algorithms in each system component, following the framework shown in Figure 1. Our system takes in the training document collection and generates a list of indexed terms. After that, each indexed terms are weighted, and the document corpus is modeled by traditional VSM as a weighted TDW matrix. PLSA model is applied to generate a "latent topic" level (LTD) matrix, and WordNet ontology is feed into the system to generate a new term-document matrix, and a "synset" level (SD) matrix. These matrixes are then combined together for final document representation, and used for SVM classifier training. These matrixes are then combined together for final document representation, and used for SVM classifier training. More details will be discussed in the following sub-sections.

A. Build Vector Space Model

As discussed in Section 1, text document is usually represented by VSM for the ease of computation and analysis. A vector space model should be built based on carefully selected terms and weighting schemes [35]. The vector space model generation consists of two stages: word indexing and term weighting scheme. First of all, the documents are represented as a set of keywords called indexed terms. After document indexing, a proper global weighting scheme is selected and applied to the vectors to adjust the influence each term may have on the model based on their appearing frequency.

1) Generating VSM model for a document

For a given set of training documents Tr, $Tr = \{D_1, D_2, ..., D_N\} = T_1 \cup T_2 ... \cup T_C$, where D_l is the l^{th} training document, C is the number of document categories, and T_c is the set of documents that belong to category c, $c = 1,2,...,C$, our vector space model is built through the following machine learning process:

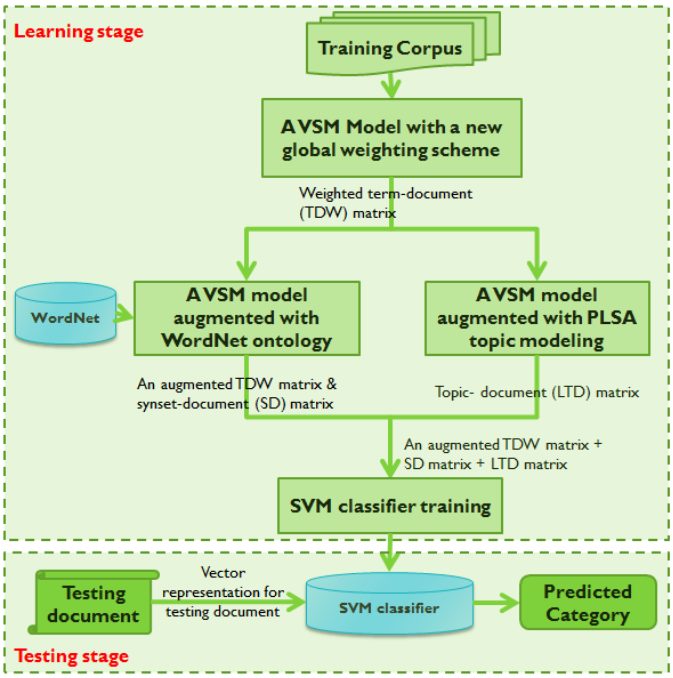


Figure 1. Proposed text categorization system framework

First of all, we generate an indexed term list from Tr, denoted as T_L, $T\_L = \{t_1, t_2, ..., t_K\}$, where t_i is the i^{th} indexed term, through a number of preprocessing tasks, including word tokenization, symbol and punctuation removing, automatic typo correction [3], stopping word removal and low-frequency term removal, etc. While generating the list of indexed term words, we need to keep only content bearing words, implying that the function words having both low and high frequency have to be removed [36]. As a result, we removed the high frequency stop word at the first stage, and then set up a term frequency threshold τ as a filter for low frequency words.

Secondly, for a document $D_l \in Tr$, its vector representation is defines as following:

$$W_{D_l} = [w_{1l}, w_{2l}, ..., w_{Kl}],$$

$$w_{il} = tf_{il} * g_{il}, i = 1,2,...,K.$$

Where tf_{il} is the occurrence frequency of term t_i within D_l, g_{il} is a global weight for term t_i, and K is the number of term features.

VSM models based on appropriate term weighting schemes is one of the most effective ways for information retrieval and

text categorization [35]. Global weighting schemes should be applied to each indexed term with the purpose of reducing or enhancing the effect they have on particular document. A number of term weighting schemes can be found in [36]. Examples of a well-known global weight schemes used in text mining is inverse document frequency (*idf*), which is defined as:

$idf_i = \log_2\left[\frac{N\_d}{df_i}\right]+1$, where tf_{ij} is the occurrence frequency of term t_i within document D_j; df_i is the document frequency, i.e., the total number of documents in the document collection that contain t_i, and N_d is the total number of documents in the training data set. When *idf* is used as the global weight function, we have $g_i = idf_i$.

2) An innovative global weighting scheme

Although there are plenty of global weight approaches available, most of them are designed for the entire dataset, i.e., training document corpus *Tr*. Based on our observation, important term words or their synonyms appear frequently in documents in a specific category, especially when the user defined category is determined by some specific keywords [2]. As a result, we developed the following category-entropy weighting function, denoted as *CE_W*:

1. For each term t_i in the term list *T_L*, calculate the proportion of the documents in *Tr* that contain t_i within *C* different categories.

   $p_{ij} = \frac{N\_c_{ij}}{N\_c_j}, j = 1,2,...C$,

   where N_c_{ij} is the number of documents within the j^{th} categories that contains t_i, and N_c_j is the total number of documents in the j^{th} category.

2. Normalize p_{ij}, so that $\alpha_{ij} = \frac{p_{ij}}{\sum_{j=1}^{C} p_{ij}}$.

3. Calculate the entropy with respect to t_i: $E_i = \sum_{j=1}^{C} -\alpha_{ij} \log \alpha_{ij}$.

   The entropy measure is a good indicator of how term t_i is distributed over different document categories. The higher the entropy, the less important item t_i is, since it is more evenly distributed among the document categories.

4. Calculate the global weight CE_W_i for t_i: $CE\_W_i = 1 - \frac{E_i}{\log C}$, where *C* is the total number of categories. This global weight function gives more weights to terms that have small entropy values.

We will show in Section 3 that the category-entropy based global weight function performs better than the inverse document frequency (*idf*) method.

At the end of VSM generation step, the output is a TDW matrix M_0, $M_0 = [W_{D_1}^T, W_{D_2}^T, ..., W_{D_N}^T]$.

B. A VSM augmented with ontology network

External or background knowledge has been found useful in improving text categorization, especially for short or ambiguous documents [37, 38]. It has a great advantage in helping extract semantic relationships, match important phrases, strengthen co-occurrences, etc. As discussed in Section 1, WordNet is one of the best known sources of external knowledge used for text categorization. It contains a network of semantically related words. Words are grouped into semantic groups (synsets) that provide word synonyms together with short explanations and general definitions. For the purposes of text categorization, it is mostly used to unify the vocabulary across the documents by modifying the document features with use of the related words [24].

In our proposed text categorization system, WordNet is used in two ways, derived from and modified based on basic approaches introduced in [21]: "add" and "repl" rules.

For each indexed term word t_i generated by VSM model, we could use POS tagging such as Stanford POS tagger [39] to identify its lexical category, and then find list of synonyms S_i for t_i in WordNet ontology. However, in a lot of applications, POS tagging may not be very reliable, e.g., text documents are noisy and lack of grammar structure and sentence boundary [3]. As a result, we generate the synset for t_i only considering one word class from "Noun", "Verb" or "Adjective", and choose the best one based on categorization performance, which will be discussed in section III.

1) Update term-document matrix (modified "repl" rule)

Under this rule, the term-document matrix M_0 generated in II-A is updated using WordNet, in a way that for a term t_i that has synset S_i, its weight in document D_l is updated using the following equation:

$w'_{il} = \max(\forall w_{rl} | t_r \in S_i, t_r \in T\_L)$.

The above equation ensures that similar terms share the same weighting value, so that they are considered as equally important. For example, if term t_x = "entire" appears in document *A* and t_y = "total" appears in document *B*, and suppose $S_x = S_y = \{t_x, t_y\}$, then we will have $w'_{xA} = w'_{yA} = w'_{xB} = w'_{yB}$. The output from this stage is an updated TDW matrix M_1 that has the same dimension as M_0.

2) Generate synset -document (SD) matrix ("add" rule)

Under this rule, a SD matrix M_c is generated using WordNet by introducing the "synset" level features, which represents the group of synonyms for each term. Mathematically, for a document $D_l \in Tr$, its "synset" vector representation Q_{D_l} is defines as following:

$$Q_{D_l} = [q_{1l}, q_{2l}, ..., q_{Vl}],$$

$$q_{il} = \sum_{r=1}^{R} (tf_{rl} * g_{rl}), i = 1,2,...,V; t_r \in S_i$$

, where q_{il} denotes the summation of term weighting values within each group of synonyms, S_i represents the i^{th} synonym group, and V represents the total number of synonym groups generated. Thus, we have:

$$M_c = [Q_{D_1}^T, Q_{D_2}^T, ..., Q_{D_N}^T].$$

C. A VSM augmented with PLSA topic modeling

1) Learning PLSA model from training documents

PLSA model is a well-known statistical language model for text clustering and information retrieval [27]. It represents a document with a convex combination over "latent topics", which are estimated by updating and fitting model parameters in order to maximize the joint probability of observed document-word pairs. It is a statistical variant of LSI [26] and based on a statistical generative model called Aspect Model [27]. The starting point of PLSA is the term-document frequency (TDF) matrix before applying global weight scheme, and it follows the bag-of-words assumption, in which each word appears independently, and the occurring order of each word is not considered. Figure 2 shows the graphical model representation of PLSA, based on Bayesian Networks [40].

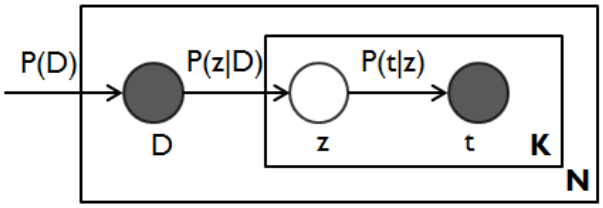


Figure 2. Graphical model representation of PLSA

In the above graphical model, the solid circles *D* and *t* represents a document and a term that are observed by people. PLSA model is a generative model that assumes there is a latent "topic" variable *z* between documents and terms. The two rectangles represents number of sample documents or words observed, and $P(D)$, $P(z \mid D)$, $P(t \mid z)$ represents the probabilities of observing a document *D*, a latent topic *z* occurring in *D*, and word *t* belonging to *z*, respectively.

The typical approach of PLSA modeling, is to estimate $P(z \mid D)$ and $P(t \mid z)$, for each document-topic and term-topic pair, by maximizing the following log–likelihood function:

$$L = \sum_{D,t} n(D,t) \log(P(D) \sum_{z} P(t \mid z) P(z \mid D)), \quad (1)$$

Where $n(D,t)$ denotes the term frequency of *t* appears in document *D*.

The generation process of PLSA model for training document corpus is proposed as following:

1. Select a document *D* from *Tr* based on $P(D)$.
2. Pick a topic *z* according to $P(z \mid D)$.
3. Given *z*, generate a word *t* based on $P(t \mid z)$.

The variables $P(z \mid D)$ and $P(t \mid z)$ are what we are interested in and want to estimate. Since $P(D)$ is not related to the parameter we want to estimate and we assume that it is constant among documents in *Tr*, we then have:

$$\arg\max(L) \propto \arg\max \sum_{D,t} n(D,t) \log(\sum_{z} P(t \mid z) P(z \mid D)) \quad (2)$$

This maximization likelihood estimation can be solved using Expectation Maximization (EM) algorithm [41]. Each iteration of EM algorithm consists of expectation step (E-step) and maximization step (M-step). In E-step, based on the current estimated $P(z \mid D)$ and $P(t \mid z)$, the posterior probability of $P(z \mid D,t)$ is computed for each document-word pair. In M-Step, $P(z \mid D)$ and $P(t \mid z)$ are updated by maximizing equation (2). Detailed steps of EM algorithm are discussed below:

Initialization: First determine maximum number of iterations *R*, and number of topics *G* to be generated. For each document-topic and topic-word pair, assign random values to $P_0(z \mid D)$ and $P_0(t \mid z)$.

E-step: At iteration *r*, for each observed topic, word and document, $z^{(p)}$, $t^{(b)}$, and $D^{(a)}$, compute:

$$P_r(z^{(p)} \mid D^{(a)}, t^{(b)}) = \frac{P_r(t^{(b)} \mid z^{(p)}) P_r(z^{(p)} \mid D^{(a)})}{\sum_{z} P_r(t^{(b)} \mid z) P_r(z \mid D^{(a)})} \quad (3)$$

where $P_r(z^{(p)} \mid D^{(a)})$ and $P_r(t^{(b)} \mid z^{(p)})$ are derived from iteration *r-1*.

M-step: At iteration *r*, for each document-topic and topic-word pair, compute $P_{r+1}(z^{(p)} \mid D^{(a)})$ and $P_{r+1}(t^{(b)} \mid z^{(p)})$ based on the following updating formulas:

$$P_{r+1}(t^{(b)} \mid z^{(p)}) = \frac{\sum_{D} n(D,t^{(b)}) P_r(z^{(p)} \mid D,t^{(b)})}{\sum_{D,t} n(D,t) P_r(z^{(p)} \mid D,t)} \quad (4)$$

$$P_{r+1}(z^{(p)} \mid D^{(a)}) = \frac{\sum_{t} n(D^{(a)},t) P_r(z^{(p)} \mid D^{(a)},t)}{\sum_{t} n(D^{(a)},t)} \quad (5)$$

More detailed derivation of (3) to (5) can be referred to [27], [42] and [43].

The output from this stage after EM learning is a latent topic-document (LTD) matrix M_{td}, in which each document has a vector representation H_{D_l} that is mapped from indexed term space to latent topic space, $H_{D_l} = [P_{1l}, P_{2l}, ..., P_{Gl}]$, and $P_{il} = P_R(z^{(i)} \mid D^{(l)})$, $i = 1,2,...G$, where R denotes the maximum number of iterations EM went through, and G denotes the number of topics generated. As a result, we have: $M_{td} = [H_{D_1}^T, H_{D_2}^T, ..., H_{D_N}^T]$.

2) Generate topic-document vector for unknown document

Although PLSA is originally designed for unsupervised learning, it can be easily extended to unknown documents. For a testing document D_u, we run through EM algorithm again, with all other parameters kept fixed, except $P(z \mid D_u)$. In E-step, based on the current estimated $P(z \mid D_u)$, the posterior probability of $P(z \mid D_u, t)$ is computed. In M-Step, only $P(z \mid D)$ is updated by equation (5). Therefore, a vector representation H_{D_u} is generated for D_u, with the same dimension as H_{D_l}.

D. Generate hybrid VSM model for classification

The matrixes generated in above sections are combined together as the final VSM representation of documents in *Tr*. The process of VSM matrix generation and augmentation is shown in the following Figure 3. The combined matrix is then used to learn and evaluate classification model, e.g., SVM, Neural Networks, Naïve Bayes classifier, etc.

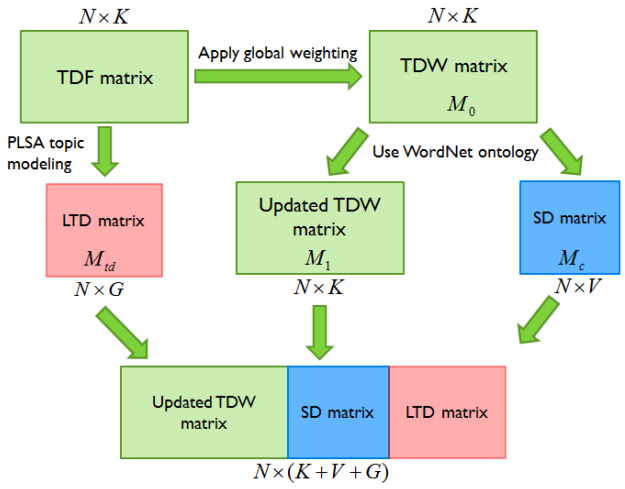


Figure 3. VSM matrix generation and augmentation

III. Empirical Case Study

In this section, we present experiments we conducted using our proposed text categorization framework, and classification results on several different datasets, in terms of accuracy.

A. Datasets

In this empirical study, we first use three publicly available and widely used datasets to evaluate our proposed system. These datasets include Reuters-21578 [45], Nist Topic Detection and Tracking corpus (TDT2) [44], and 20 newsgroups [46]. Reuters-21578 corpus contains 21578 documents in 135 categories. After removing documents with multiple category labels, it left 8293 documents in 65 categories. In TDT2, those documents appearing in two or more categories were removed, and only the largest 30 categories were kept, thus leaving 9,394 documents in total. 20 newsgroups dataset is a collection of 18846 newsgroup documents, partitioned (nearly) evenly across 20 different newsgroups.

To evaluate the system performance on domain-specific datasets that has customized category definition such as [2], we also used a dataset named VDR, that contains 600 vehicle diagnostic records, in which documents that contain descriptions that reveal systematic engineering or manufacturing failures are defined as of interests (Category-A), and all other documents belong to Category-B. The major challenge in this problem is that the documents of interests are not explicitly defined by either topics or general descriptions, as shown in the following examples:

Category-A document: "perform abs self roadtest found rear wheel speeds sensor connector corroded into sensor replace sensor and connector road tester ok clear code"

Category-B document: "road roadtest traction control lamp on eec roadtest code c1280 u415 om rcm contact hot line 103912699 check connection at rcm check mounting bolts ok clear code"

In all of these datasets discussed above, preprocessing tasks mentioned in Section II-A are conducted and stop words are removed. Note here, all words having occurrence frequency lower than $\tau = 5$ are removed, except VDR dataset. TDW matrix weighted by *CE_W* discussed in Section II-A is generated for each dataset. TDF matrix is also generated for PLSA model learning, and TDW matrix weighted by *idf* is generated for evaluation purpose.

B. Augment VSM with WordNet

WordNet ontology network is utilized in our system not only to update TDW matrix, but also to create new synset features. In our experiments, it generates 2628, 4384, 3063 and 696 synset features for Reuters, TDT2, 20newsgroups and VDR, respectively. In our experiments, we first look into the effect of text categorization using terms within different word class. The results are shown in Table I. It is obvious that the best word class is "Noun", which only generates 377, 1161, 621 and 21 "synset" features for Reuters, TDT2, 20newsgroups and VDR, respectively, and having a promising text categorization accuracy.

TABLE I. TEXT CATEGORIZATION PERFORMANCE USING WORDNET BASED ON DIFFERENT WORD CLASS

Categorization accuracy	Noun	Verb	Adjective	Mixed
Reuters	**92.74%**	92.34%	89.01%	92.26%
TDT2	**97.07%**	96.22%	96.15%	96.64%
20news	**87.68%**	86.48%	86.58%	86.73%
VDR	**84.53%**	82.32%	82.32%	83.97%
Number of synsets generated	Noun	Verb	Adjective	Mixed
Reuters	377	229	138	677
TDT2	1161	666	645	2201
20news	621	537	172	1165
VDR	21	28	11	58

C. Augment VSM with PLSA

With the purpose of being consistent, in PLSA learning, for all datasets, we define the maximum number of iterations $R = 500$ for training set, and $R = 200$ for testing set. Number of latent topics is defined as $G = 40$. These parameters can be tuned for optimized results, which will also be investigated in the next step of work. Also, the convergence goal is defined as ε = 1E-5. An example of log-likelihood function maximization on Reuters dataset is shown in Figure 4. It is obvious that the log-likelihood function converges very fast and become very stable after 500 iterations.

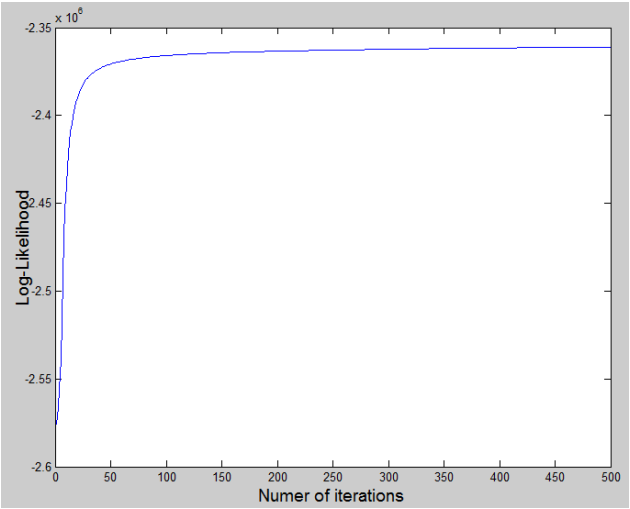


Figure 4. Example of log-likelihood maximization

D. SVM training

Considering that the focus of this work is not improving or comparing machine learning algorithms, we use SVM as our classification model throughout different experiments. SVM training is carried out with LIBSVM package, which is developed by Chih-Chung Chang and Chih-Jen Lin from National Taiwan University [47]. For each dataset, we did 3-fold cross validation, and in each fold, we choose 2/3 documents from each class as training set, and the remaining 1/3 documents as testing set. We apply the Gaussian Radial Basis kernel function (RBF) and tune the parameter gamma to 0.001, 0.001, 0.1 and 0.1, for Reuters, TDT2, 20newsgroups and VDR, respectively, based on the average testing accuracy of the 3 folds.

E. Experiment results & analysis

The classification results are presented in the following Table I. We evaluate several systems as our baseline, including TDW matrix weighted by *idf*, and using only PLSA generated LTD matrix. From the result, it is obvious that global weighting scheme *CE_W* outperform the *idf* weighting, and the SD matrix generated by WordNet improves categorization accuracy by combining with the *idf* weighted TDW matrix. Furthermore, the final proposed system, with *CE_W* weighted and WordNet updated TDW matrix, plus SD matrix generated by WordNet and LTD matrix generated by PLSA, significantly outperforms all other systems, indicating that adding both word relationships and latent semantic information could improve text representation.

TABLE II. TEXT CATEGORIZATION SYSTEM ACCURACY COMPARISON

	Reuters	TDT2	20news	VDR
TDW matrix weighted by *idf*	91.03%	89.37%	85.85%	80.95%
TDW matrix weighted by *CE_W*	92.10%	96.01%	87.25%	85.18%
TDW matrix weighted by *idf* + SD matrix	91.42%	95.09%	87.10%	83.07%
LTD matrix by PLSA	84.60%	90.24%	79.46%	82.32%
Updated TDW matrix weighted by *CE_W* + SD matrix + LTD matrix	**93.06%**	**98.78%**	**88.84%**	**87.84%**

IV. CONCLUSION

This paper proposes an innovative text categorization model, VSM_WN_TM, based on Vector Space Model (VSM), WordNet ontology, and PLSA topic modeling. Support Vector Machine is used as document classifier, and the proposed system is evaluated on publicly available datasets and domain-specific dataset. Experiment result shows that incorporating semantic and syntactic relationship among words such as synonymy, co-occurrence and context could greatly improve text representation, and our approach significantly outperforms conventional approaches such as using only BOW features or latent topic features.

REFERENCES

[1] B S Harish, D S Guru, S Manjunath, Representation and Classification of Text Documents: A Brief Review. IJCA Special Issue on *Recent Trends in Image Processing and Pattern Recognition*, RTIPPR, 2010.

[2] Yinghao Huang, Naeem Seliya, Yi Lu Murphey and Roy B. Friedenthal, Classifying Independent Medical Examination Reports

using SOM networks, *Proceeding of the 6th International conference on Data Mining*, Las Vegas, Nevada, USA, 2010, p58-64

[3] Yinghao Huang, Yi Lu Murphey and Yao Ge, Automotive diagnosis typo correction using domain knowledge and machine learning, *IEEE Symposium Series on Computational Intelligence 2013.*

[4] Zi-Qiang Wang, Xia Sun, De-Xian Zhang, Xin Li "An Optimal SVM-Based Text Classification Algorithm" Fifth International Conference on Machine Learning and Cybernetics, Dalian,pp. 13-16 , 2006.

[5] Jingnian Chen a,b,, Houkuan Huang a, Shengfeng Tian a, Youli Qua Feature selection for text classification with Naïve Bayes" Expert Systems with Applications 36, pp. 5432–5435, 2009.

[6] Bo Yu, Zong-ben Xu, Cheng-hua Li ,"Latent semantic analysis for text categorization using neural network", Knowledge-Based Systems 21- pp. 900–904, 2008.

[7] Tai-Yue Wang and Huei-Min Chiang "One-Against-One Fuzzy Support Vector Machine Classifier: An Approach to Text Categorization", Expert Systems with Applications, doi: 10.1016/j.eswa.2009.

[8] Bang, S. L., Yang, J. D., & Yang, H. J. , "Hierarchical document categorization with k-NN and concept-based thesauri. Information Processing and Management", pp. 397–406, 2006.

[9] Trappey, A. J. C., Hsu, F.-C., Trappey, C. V., & Lin, C.-I., " Development of a patent document classification and search platform using a back-propagation network", Expert Systems with Applications, pp. 755–765, 2006.

[10] Dino Isa,, V. P Kallimani Lam Hong lee, "Using Self Organizing Map for Clustering of Text Documents", ", Elsever, Expert System with Applications-2008.

[11] Sang-Bum Kim, Kyoung-Soo Han, Hae-Chang Rim, and Sung Hyon Myaeng, "Some Effective Techniques for Naive Bayes Text Classification", IEEE Transactions On Knowledge And Data Engineering, Vol. 18, No. 11, , Pp- 1457- 1466 ,November 2006.

[12] Bao Y. and Ishii N., "Combining Multiple kNN Classifiers for Text Categorization by Reducts", LNCS 2534, , pp. 340- 347, 2002.

[13] Cheng Hua Li , Soon Choel Park, "An efficient document classification model using an improved back propagation neural network and singular value decomposition" Expert Systems with Applications 36 ,pp- 3208–3215, 2009.

[14] Dik L. Lee, H.C., Kent Seamons, Document Ranking and the Vector-Space Model, IEEE Software, 1997. 14(2): p. 65-75.

[15] Wongkot Sriurai, IMPROVING TEXT CATEGORIZATION BY USING A TOPIC MODEL, Advanced Computing: An International Journal (ACIJ), Vol.2, No.6, November 2011

[16] Nagarajan, M., Sheth, A.P., Aguilera, M., Keeton, K., Merchant, A. and Uysal, M. Altering Document Term Vectors for Classification - Ontologies as Expectations of Cooccurrence LSDIS Technical Report, November, 2006.

[17] Burger, S. and Stieger, B., Ontology-based classification of unstructured information, Fifth International Conference on Digital Information Management (ICDIM), pp. 254-259, July 2010.

[18] Jun Fang, Lei Guo and Yue Niu, Documents classification by using ontology reasoning and similarity measure, 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), vol. 4, pp. 1535-1539, Aug. 2010.

[19] Janik, M., Kochut, K.J.: Wikipedia in Action: Ontological Knowledge in Text Categorization. 2nd International Conference on Semantic Computing (ICSC), Santa Clara, CA, USA (2008)

[20] Buitelaar, P., Cimiano, P., Magnini, B., Ontology learning from text: An overview. Ontology learning from text: Methods, evaluation and applications. Frontiers in Artificial Intelligence and Applications Series 123 (2005).

[21] A. Hotho, S. Staab, and G. Stumme. Ontologies improve text document clustering. In Proceedings of the International Conference on Data Mining — ICDM-2003. IEEE Press, 2003

[22] B. Shi, et al., Classification of Semantic Documents Based on WordNet, International Conference on E-Learning, E-Business, Enterprise Information Systems, and E-Government, vol. 0, pp. 173-176, 2009.

[23] Litvak, M., Last, M., & Kisilevich, S. (2007). Classification of Web documents using concept extraction from ontologies. Lecture Notes in Computer Science, 4476, pp. 287-292.

[24] WordNet: An Electronic Lexical Database. The MIT Press (1998)

[25] Alt.ncay, H., and Erenel, Z. 2010. Analytical evaluation of term weighting schemes for text categorization. In Journal of Pattern Recognition Letters, vol. 31 (11), pp. 1310 . 1323.

[26] Landauer, Thomas K., and Dumais, Susan T., Latent Semantic Analysis, Scholarpedia, 3(11):4356, 2008.

[27] Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd ACM SIGIR, pp. 50–57 (1999)

[28] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. J. Mach. Learn. Res., 3:993–1022, 2003.

[29] Shibin Zhou,Kan Li,Yushu Liu, Text Categorization Based on Topic Model, International Journal of Computational Intelligence Systems, Vol.2, No. 4 (December, 2009), 398-409

[30] Bıro, I. & Szabo, J. (2010) "Large scale link based latent Dirichlet allocation for web document classification"

[31] Bıro, I. & Szabo, J. (2009) "Latent Dirichlet Allocation for Automatic Document Categorization". In Proceedings of the 19th European Conference on Machine Learning and 12th Principles of Knowledge Discovery in Databases.

[32] Yue Lu, Qiaozhu Mei and ChengXiang Zhai, Investigating task performance of probabilistic topicmodels: an empirical study of PLSA and LDA, Inf Retrieval (2011) 14:178–203

[33] Pei Yang, Wei Gao, Qi Tan, Kam-Fai Wong, A link-bridged topic model for cross-domain document classification, Information Processing and Management 49 (2013) 1181‑ 1193, 2013 Elsevier.

[34] Xue, G. R., Dai, W. Y., Yang, Q., & Yu, Y. (2008). Topic-bridged PLSA for cross-domain text classification. In SIGIR-2008 (pp. 627‑C634).

[35] Salton, G., Buckley, C., Term weighting approaches in automatic text retrieval, *Information Processing and Management*, 24(5): 513-523, 1998.

[36] Liping Huang, Yi Lu Murphey: Text Mining with Application to Engineering Diagnostics. *IEA/AIE 2006*: 1309-1317

[37] Bloehdorn, S., Hotho, A.: Text Classification by Boosting Weak Learners based on Terms and Concepts. 4th IEEE International Conference on Data Mining (ICDM'04)(2004)

[38] Zelikovitz, S., Hirsh, H.: Improving Short Text Classification Using Unlabeled Background Knowledge. Seventeenth International Conference on Machine Learning (ICML), Stanford, CA (2000)

[39] Kristina Toutanova and Christopher D. Manning. 2000. Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger. In Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000), pp. 63-70.

[40] Borgelt, Christian; Kruse, Rudolf, Graphical Models: Methods for Data Analysis and Mining. Chichester, UK: Wiley. ISBN 0-470-84337-3, March 2002.

[41] Dempster, A.P.; Laird, N.M.; Rubin, D.B. (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm". Journal of the Royal Statistical Society, Series B 39 (1): 1–38. JSTOR 2984875. MR 0501537

[42] Gradshteyn, I. S. and Ryzhik, I. M. Tables of Integrals, Series, and Products, 6th ed. San Diego, CA: Academic Press, p. 1101-2000.

[43] Vapnyarskii, I.B. (2001), "Lagrange multipliers", in Hazewinkel, Michiel, Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4

[44] Deng Cai, Xiaofei He and Jiawei Han, "Document Clustering Using Locality Preserving Indexing", IEEE TKDE 2005.

[45] Reuters-21578, Distribution 1.0. Web document. URL: http://www.daviddlewis.com/resources/testcollections/reuters21578/

[46] Ken Lang, Newsweeder: learning to filter netnews. Proceedings of the Twelfth International Conference on Machine Learning, pp. 331-339, 1995.

[47] C.-C. Chang and C.-J. Lin. LIBSVM : a library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2:27:1--27:27, 2011.

Personalizing Query Refinement Based on Latent Tasks

Chao Xu, Mingzhu Zhu, Yanchi Liu, and Yi-fang Wu
Information Systems, New Jersey Institute of Technology, Newark, NJ, USA

Abstract - *Search tasks, comprising a series of search queries representing the similar information need, have been accepted as atomic units of modeling users' search intensions. However, most studies on user search tasks focus on applying user interests in personalizing search results, and few have examined the performance of applying search tasks to optimize the ranking list of candidate queries generated by query refinement. Moreover, few of the existing studies has examined the dynamic characteristics of user search task and user interests within a search task. This paper proposes a method for user profiling which addresses the two issues above. Our approach uses a four-descriptor model to represent long-term and short-term user interests extracted from search sessions within a search task. Then a graphical model is proposed to improve the rankings of the candidate queries for query refinement to assess the task dependency via exploiting a latent task space. Experimental results show that the proposed user-profiling method contributes to an increased precision of search results, and it produces more accurate refined queries, thereby potentially resulting in shorter search sessions.*

Keywords: Query Refinement, Search Task, Personalization

1 Introduction

Search engine users' information needs span a broad spectrum, such as booking a hotel, renting a house, and planning a journey, etc. As a common medium of recording users' search history, user logs are widely utilized to extracted users' search interests as good indicators of user's information needs and are mostly applied in the re-ranking and/or personalization of search results. However, few studies have examined the effectiveness of applying user's interests to assist them in obtaining queries that are more effective. Because user's information needs are various regarding their diverse backgrounds and search goals, it is very important for users to choose appropriate keywords to better describe their search intentions. Current studies, e.g. [1], have shown that users often provide short queries. Short queries are usually ambiguous and may not have enough context to accurately represent the user search intent. Many studies have been conducted aiming at helping users to build more effective queries. One of the famous methods is query refinement which is a process of generating a candidate query list on the basis of the original queries of a user. By generating more effective query, query refinement helps users to reformulate ill-formed queries and hence enhance the relevance of search results. Although numerous query refinement methods can help users to issue more effective queries than the originals, these methods fail to consider the diversity of user search intentions. For example, if two users having different interests such as coffee and programming language input the same query "java", then the current query refinement approaches provide both users the same candidate query list, such as "java script", "java games", "adobe", etc. However, this query list is useless to the user who wants to find information about coffee. Therefore, providing candidate query lists according to individual user interest is more beneficial than providing a generalized candidate query list. In this paper, we introduce an approach for identifying and applying user interests to personalize query refinement. The objective is to satisfy user's information need faster by providing more effective candidate queries of query refinement for each individual. These queries are generated according to not only user's original search query but also user's search interest. Therefore, new generated candidate query will result in more relevant search results and user's information needs will be satisfied faster.

Studies have shown that the vast majority of users are reluctant to provide any explicit feedback on search results and their interests [2]. Therefore, our proposed approach automatically learns user interests by using the past click history of users and applies the learned interests to analyze the user information needs. Meanwhile, studies prove that applying personalization is not always appropriate for aiding the user in accomplishing their information needs. This is due to the lack of examining and modeling the user's search contexts and activities besides the real time relevance feedback [3, 4]. In fact, it is crucial to restrict personalization to those queries that benefit from its application. Task-oriented user-search behavior analysis is a popular method for analyzing user search context and activities [12, 13]. However, few studies examine how to apply it into modeling user's dynamic search interests and none studies explore the latent task dependency for each candidate query. In this paper, we propose a personalization framework that scores candidate queries regarding their latent task dependency.

Moreover, most of the profiling techniques usually perform single-descriptor representation to model the general search interests of users. These techniques are effective in learning user interests, when user interests change at a constant rate. However, user's interests are not static but dynamically changing. The methods ignoring the dynamic characteristics

of such interests cannot adapt to the abrupt changes in user interests. Learning the change of user interests is closely related to learning long-term and short-term model [5] of the user search history. Long-term user interests represent the general preference of users. These interests are formed gradually over the long run and are stable after they converge. By contrast, short-term interests are unstable by nature. For example, user's interests on hot topics, i.e. political news, change from day to day. A single-descriptor representation cannot adapt to both types of interests simultaneously. Although a system could be designed to adapt to changes in short-term user interests by maintaining a fixed amount of recent feedbacks [6], it might ignore learned long-term user interest. These two problems indicate a need to develop a representation that can trade-off the shortcomings and benefits between long-term and short-term user interest models. In this paper, we use a four-descriptor model to represent and learn the long-term (positive and negative) and short-term (positive and negative) user interests for each task generated from past user search histories.

A number of studies argue that long-term and short-term user's interests can be calculated by time interval. For example, one study [6] denotes the final interests as sum of the weighted user interests for the past week, month, and the entire period. Although this method separates user interests from the time periods, it does not examine user interests in terms of different search contexts, such as search sessions or search tasks. In the proposed method, we model short-term and long-term user interests at the level of search session and search task, respectively. This approach has the following main advantages: 1) it distinguishes user interests in different periods, and 2) it keeps user interests distinguishable at the individual search activity levels.

This paper makes the following contributions:

- theorizes a four-descriptor model to learn and analyze long-term and short-term user interests;
- proposes a graphical model based on latent tasks for scoring candidate queries of query refinement;
- introduces a framework for personalization query refinement involving proposed user profiling model.

The rest of the paper is organized as follows. Section 2 summarizes studies that are similar to the current research. Section 3 presents user-profiling method, latent task based query scoring method for query refinement, and proposed framework of personalizing query refinement. Section 4 introduces the dataset, experimental design, evaluation methods, and performance comparison of the proposed system with the baseline systems. Finally, Section 5 summarizes the main conclusions and future works of this study.

2 Related Work

2.1 Query Refinement

Studies have shown that most queries are short and cannot express the actual user search intent. Query refinement, which is a process that provides users with a candidate query list based on their original queries, has attracted considerable attention on reducing the ambiguity of user queries.

Recently, a large number of approaches, e.g. [7, 8], on query refinement based on mutual information (MI) have been proposed. Some of these approaches [8] adopt the context information of user queries to analyze user search interests. However, existing studies have failed to consider the semantic relationships between terms in the generated query list [7]. A topic model-based method [6] has been proposed to extract the latent topics from user search histories to assess the query dependency of words in the query. Numerous approaches [12] have been adopted to improve the accuracy of query refinement by using session information from user search logs. For example, Bing et al. [7] utilize session information to construct a query–query Markov graph to generate candidate queries.

Although the above-mentioned studies have provided effective candidate query list based on original queries, none of them has examined how closely the generated candidate queries match user's preferences and interests.

2.2 Search Task

Search logs are viewed as rich resources for user search activities. A large number of studies have focused on methods for classifying queries into separate search goals to extract user search intentions [9]. User search contexts are found to contribute much information for analyzing user search interests [10] and for improving the performance of search results ranking. Prior research effort in examining user search contexts can be categorized into two classes: search session and search task.

A search session, as defined in [11], is a sequence of queries issued by a single user within a specific time limit. The related queries of the same session often refer to the same search goal or search activity. Based on this assumption, an algorithm has been proposed to group queries into search sessions by detecting the topic shifts among queries [12]. Another study [13] adopts topic models to extract session-level search goals. It is concluded that the method of examining user search activities through search sessions outperforms the traditional approaches that are based on relevance feedback.

Meanwhile, Georg et al. [14] indicate that a search task can be complex which spans a number of search sessions. To tackle this, they propose a method to generate task tour which comprising a set of related search tasks. Ming et al. [15] have proven the effectiveness of classifying queries and web pages into search tasks on improving the search performance. He et al. [16] introduced a two-step method for task identification. They first identify search session by setting a time interval as the session boundary, then a unsupervised method is conducted to extracted the multi-session search tasks. In this

paper, we adopt their method due to its effectiveness and simplicity to be implemented.

3 Methods

3.1 Constructing User Profiles

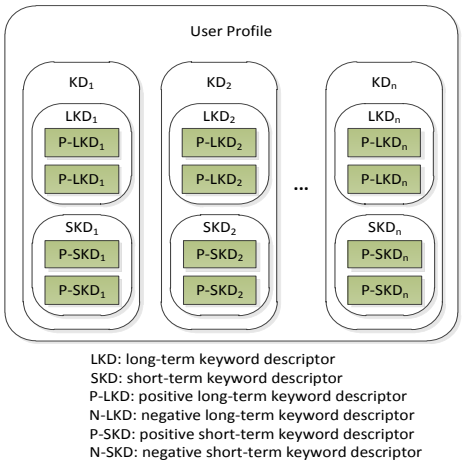


Fig-1. Task-based User Profile

Figure 1 shows the user profile proposed in this study. The proposed approach is a task-oriented method for user profiling, in which the user interests are learned within the context of a particular search task. In a search task, user interests are modeled by a keyword descriptor (KD) which is represented by a term-weight vector, as shown in Equation 1.

$$KD = <<t, w_1><t_2, w_2>\ldots<t_n, w_n>> \quad (1)$$

Each KD is represented by two descriptors, namely, long-term keyword descriptor (LKD) and short-term keyword descriptor (SKD). The long-term user interests are modeled by two descriptors, namely, positive long-term keyword descriptor (P-LKD) and negative long-term keyword descriptor (N-LKD). Similarly, short-term user interests are modeled by two descriptors, namely, positive short-term keyword descriptor (P-SKD) and negative short-term keyword descriptor (N-KSD). User interests can be represented by the following two-level descriptor:

$$KD = < LKD<P\text{-}LKD, N\text{-}LKD>, SKD<P\text{-}SKD, N\text{-}SKD>> \quad (2)$$

This separation aims to preserve the feature vectors of relevant and non-relevant documents, thus enabling the separate measurement of the similarities between subjects of interest and the positive and negative interests. The degree of a user's interest in a subject is computed by subtracting the user interest when negative descriptors are used from that when positive descriptors are used. The relevance feedback of all session data within a search task is used to model the long-term user interests, whereas the current user search session is used to model the short-term user interests.

A large number of algorithms can effectively obtain user relevance feedback. The Rocchio algorithm is one of the most famous algorithms and is widely used in information retrieval. Equation 3 represents the general form of the query refinement of the Rocchio algorithm during the relevance feedback process [17].

$$Q_{i+1} = Q_i + a\sum_{pos} D_j / n_{pos} - b\sum_{neg} D_j / n_{neg} \quad (3)$$

where a + b =1, Q_i indicates original user's interest in the query Q, Q_{i+1} indicates the updated user's interest in the query Q, D_i denotes the relevant document for Q_i, n_{pos} is the number of relevant documents, and n_{neg} is the number of non-relevant documents.

In this paper, we adopt Rocchio algorithm to learn the user relevance feedback in a four-descriptor model. For example, P-LKD and N-LKD are updated by the following equations:

$$P\text{-}LKD_{new} = P\text{-}LKD_{old} + D_{pos} - D_{neg} \quad (4)$$

$$N\text{-}LKD_{new} = N\text{-}LKD_{old} + D_{pos} - D_{neg} \quad (5)$$

where D_{pos} is the positive relevance feedback, and D_{neg} is the negative relevance feedback. The method of extracting the D_{pos} and D_{neg} for LKD and SKD is introduced in later subsections (3.3 and 3.4). The long-term user interest in task i is represented by $ILKD_i$, which is expressed as follows:

$$ILKD_i = \alpha\ P\text{-}LKD - (1-\alpha)N\text{-}LKD \quad (6)$$

Similarly the short-term user interest in task I is represented by $ISKD_i$, which is expressed as follows:

$$ISKD_i = \beta\ P\text{-}SKD - (1-\beta)\ N\text{-}SKD \quad (7)$$

where $\beta\ \varepsilon\ (0,1)$, β is the positive short-term interest weight, and 1–β is the weight of the negative short-term interest. The final interest in a task i is given by

$$IKD_i = \gamma\ ILKD_i + (1-\gamma)\ ISKD_i \quad (8)$$

where the range of $\gamma\ \varepsilon\ (0,1)$, γ is the long-term interest weight, and 1–γ is the short-term interest weight. Finally, as for any individual user, his/her interests in task i is represented as a word-weight pair vector, KD, as shown in equation 1. Note that the weights is normalized so that we have the constraint, $\sum_i w_i = 1$.

3.2 Latent Task-based Candidate Query Scoring

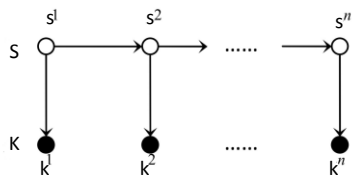


Figure 2: Latent task model for a candidate query

A candidate query for query refinement is a sequence of keyword denoted as q: k^1, k^2, …, k^n, where n represent the position of the keyword within the query after stopword removing. The latent task of k^1 is denoted as s^1, which is a task from the full task set S. We represent such a generative

process using a graphical model which is shown in figure 2. The latent search task is unobservable and represented by empty nodes. We compute the joint distribution of the term sequence denoted as $P(k^{1:n})$ for scoring the candidate query. Let $s^{1:n}$ be the task sequence, the candidate query score can be computed as

$$P(k^{1:n}) = \sum_{s^{1:n}} P(k^{1:n}, s^{1:n}) \quad (9)$$

According to the dependency structure as shown in figure 2, the marginal distribution of task sequence and keyword sequence can be shown in as

$$P(k^{1:n}, s^{1:n}) = \prod_{i=1}^{n} P(k^i|s^i)P(s^1)\prod_{i=2}^{n} P(s^i|s^{i-1}) \quad (10)$$

Where $P(k^i \mid s^i)$ denotes the probability that keyword k^i is generated by task s^i. Another part the model $p(s^i \mid s^{i-1})$ denotes the relationship between two search tasks. Such a relationship enables a means of governing the task context of neighboring keyword in a query sharing similar task content.

The parameter $P(k^i \mid s^i)$ can be easily obtained via equation 1 which is indicated in section 3.1. As for the second parameter, $p(s^i \mid s^{i-1})$, we calculate the pair-wise dependent probability that task s_i is followed by s_j. Recall our objective of query refinement is to provide more relevant candidate query in which the latent search task for each keyword should be consistent because user's search intention is unique for each search query. To achieve it, we investigate the semantic similarity between each pair of search task. That is, the probability is high if the two latent search tasks are similar, vice-versa.

$$P(s_j|s_i) = \frac{sim(s_j|s_i)}{\sum_{s_j \in S} sim(s_j|s_i)} \quad (11)$$

Where $sim(s_i \mid s_j)$ is a similarity measure between task s_i and s_j. In specific, we adopt the cosine similarity to calculate such a similarity.

$$sim(s_j|s_i) = \frac{\sum_{k_l} P(k_l|s_t)P(k_l|s_t)}{\sqrt{\sum_{k_l} P(k_l|s_t)^2}\sqrt{\sum_{k_l} P(k_l|s_t)^2}} \quad (12)$$

3.3 System Framework

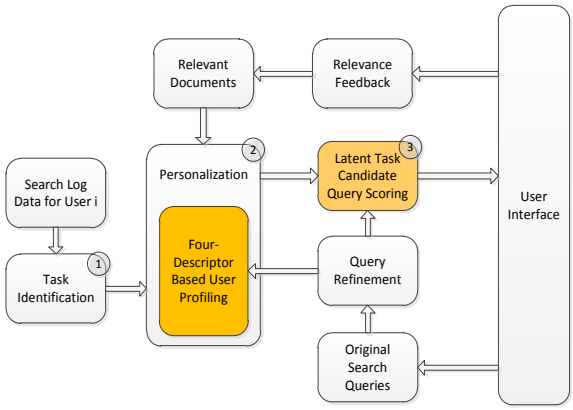


Fig-3. Task-based Personalization Framework

There are three main components (marked numerically as 1, 2, and 3 in figure 3) of the framework. First, search tasks are extracted for each individual user from the search-log training dataset by adopting the two-step method introduced by Hong et. al [10]. In specific, each session is viewed as a pseudo-document represented by a keyword vector. After the clustering algorithm is applied to the sessions, *K* clusters, which are represented by a keyword vector, are extracted. These tasks are then recorded into the user profile.

Second, the user relevance feedback is learned through the Rocchio algorithm and then applied to update the existing user profile. Combined with the search activity information, relevance feedbacks are added separately to long-term and short-term descriptors.

Third, once the user inputs a query, the top *N* candidate queries generated from the traditional query refinement method are selected, which are scored via the latent candidate query scoring module. The candidate queries are re-ranked according to their scores of latent task dependency. Finally, the original and new ranks are merged by using Borda's ranking fusion method [18].

4 Experiments

4.1 Data Sets and Preprocessing

The dataset we used was an AOL query log. The collection period began on 1 March 2006 and ended on 31 May 2006. This dataset contained 19,442,629 lines of click-through information, 657,426 unique user IDs, 4,802,520 unique queries, and 1,606,326 unique URLs. The AOL search engine recorded a dataset containing a large amount of noise, such as typographical errors and stop words. We conducted raw data preprocessing similar to that described in [7]. First, host navigation queries, such as "www.msn.com" and "www.bbc.com," were removed. Second, queries with non-alphabetical characters were removed as well. Third, stop words were removed from the queries. After duplication removal and data cleaning, we had 642,371 unique users, 4,224,165 unique queries, and 18,343,302 unique URLs in total.

Note that both session and task information can be obtained by our adopted two-step method introduced by He et al. [16]. Every two consecutive queries within the same session should share at least one term. The time interval within a session was less than 26 minutes. After session division, we split the dataset into training and test sets. The training set contained two-month-worth of search log data, whereas the test set contained one-month-worth of search log data. Pseudo-documents were constructed for each URL contained in the training and test sets. These pseudo-documents were used to represent the content of each clicked URL in the AOL dataset.

4.2 Evaluation Method

Performing manual evaluation of the output of query refinement is time consuming and labor intensive; therefore, we evaluated the output by utilizing the session information of query logs in the same manner as [7]. In a specific search session, when a user feels unsatisfied with the results of the current query, the user may refine the query and run a new search. When the user obtains satisfactory search results, the user may stop searching and start a new search activity. On the basis of the discussion by Downey et al. [19] on the importance of the terminal URL, we can conduct a reliable evaluation by using the terminal URL information. We defined two types of queries, which are mentioned in [7], as follows:

Definition 1 (satisfied query): In a user session, the query causing at least one URL to be clicked and is located at the end of the session is called a satisfied query.

Definition 2 (unsatisfied query): Any query located just ahead of the satisfied query within the same user session is called an unsatisfied query.

We adopted the metric P@K (precision at K) to evaluate the results. Here, *K* is the number of top queries given by the model. A maximum of 30 result queries were considered for each method. Considering that users more likely care about the top-ranked candidate queries, we also evaluated the performance of each system at the top *m* (*m* = 1, 2, 3, 4) of the candidate query list. For each session, we adopted the first unsatisfied query as the input query and used the corresponding satisfied query as the benchmark query of the refinement task. For each input query, the system generated a candidate query list for query refinement. If the benchmark query could be found in the top *m* queries of the list, then the input query of that list was denoted a successful query. Accuracy was defined as the total number of successful queries divided by the total number of input queries.

4.3 Parameter Selection

Before evaluating the performance of proposed framework, we need to learned the important parameters of our proposed four-descriptor model: interest impact weight of P-LKD, P-SKD, and LKD (α, β, γ). The effectiveness of the user-profiling method was measured by evaluating the performance of the method on a learning activity. A learning activity was used to simulate changes in a user interest among tasks. For simplicity, we used >> to represent the task transition within the learning activity. For example, if the user had an initial interest on the task of buying a laptop (labeled as T1, short for Task One) and then shifted this interest to the task of finding a Spanish restaurant (labeled as T2, short for Task Two), then the interest can be described as [T1] >> [!T1, T2], which represented two phases of interest learning. In this case, changing the interests consisted of learning a new interest and unlearning an old interest. An activity was designed to simulate the change of user interest from one task to another. Each phrase consisted of learning and unlearning the interests. This activity is described as follows:

Learning Activity 1: [T1] >> [!T1, T2] >> [!T2, T3] >> [!T3, T4] >> [!T4, T5].

The proposed user-profiling model can be measured by cycles of evaluations. Each cycle involves 1) learning relevance feedback from the clicked documents during the query sequence of a session and 2) measuring the accuracy by analyzing user interests at the end of each session. Each learning phrase, such as [T1] >> [!T1, T2], consisted of 10 cycles of interest learning and accuracy measurement.

At each cycle of evaluation, all the clicked documents were ranked according to their similarity values with the current user interests by using the KL-divergence algorithm. We examined the top 10 ranked documents and calculated the precision @ 10, which is defined as follows:

$$\text{precision @ 10} = \frac{\text{numbers of interested documents}}{10} \quad (13)$$

Note that the last query with at least one clicked document of each session was used as the satisfied query for evaluation, and the clicked URLs of the satisfied query were used as the user- interest documents.

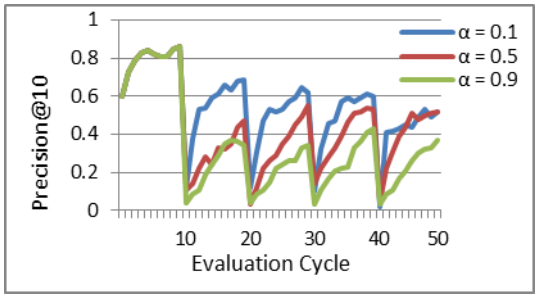


Fig-4. Performance of LKD in the Learning Activity

Figure 4 presents the performance of LKD in the test learning activity with various values of α (0.1, 0.5, and 0.9). By varying the interest impact weight α, we demonstrated that we received the highest average accuracy in the LKD model when α was 0.1. As is shown, the accuracy of matching the user interest increases steadily within each task (10 rounds of evaluation) which is caused by the accumulation of learned user interests. However, the LKD model suffers a sharp decrease of accuracy at each task transition. Even though the model can learn a user interest, the model is incapable of unlearning the old interests quickly when the user shifts to a new interest. This outcome also results in the decrease in accuracy from phase to phase. For example, in the first learning phase, the accuracy is stable at around 0.8, whereas the accuracy drops to 0.7 during the second phase and to 0.5 during the fifth phase. Thus, LKD does not match the user interest during task transitions.

The ability of the model to learn short-term user interests was examined within a session context. For a specific session, the relevant pseudo-documents of the clicked URLs were used

to simulate the short-term user interests. The initial short-term interest vector was also set to the zero vector and updated with all relevant pseudo-documents of the clicked URLs queries within the same session. The KL divergence was computed between each pair of the short-term user interest vector and each of the pseudo-documents in the corpus. The only difference between learning long- and short-term interests is that the short interest is learned within a session instead of being accumulated across sessions.

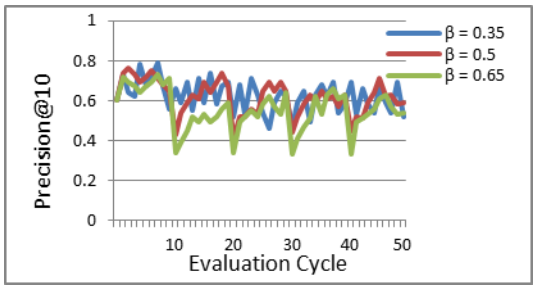


Fig-5. Performance of SKD in the Learning Activity

Figure 5 shows the performance of SKD on the test learning activity, with varying values of β (0.35, 0.5, and 0.65). Given that the SKD does not has a memory of former session interests, its accuracy in matching user interest fluctuates greatly compared with the performance of LKD in Figure 4. SKD does not learn the user interest as stably as LKD does. However, SKD exhibits stable accuracy during task transitions, particularly when β is small. This result indicates that SKD possesses better adaptability to interest changes even during task transitions. By varying the learning rate β, we found that the highest average accuracy of the SKD model is obtained when β is 0.35.

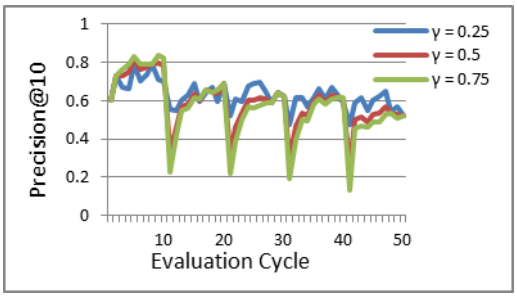


Fig-6. Performance of KD in the Learning Activity

The performance of KD was examined based on the above evaluations of LKD and SKD by setting the parameters of α as 0.1 and β as 0.35, thus maximizing the learning ability and prediction of user interests. The interest weight γ was used to control the effect of LKD and SKD in the KD. As shown in Figure 6, the system performance obtained is stable and has adaptive accuracy when the parameter γ is set to 0.25. This result indicates that SKD influences the performance of KD. KD is better at unlearning older interests than LKD is and is more reliable in matching user interests than SKD is. Thus, KD overcomes the weaknesses of both LKD and SKD.

4.4 Experimental Design

The experiment analyzed the contributions of our proposed user-profiling method to four baseline systems, namely, an MI model and a context-based mutual information (CMI) model [18], topic model based framework (LTI) [17], and a task-based method (MTP) [16]. In this study, we used these two baseline methods to generate the original candidate query list of query refinement. We then applied the proposed model to re-rank these two candidate lists. Specifically, we obtained two personalized models, namely, personalized mutual information (P-MI) and personalized context-based mutual information (P-CMI) models, after using MI and CMI, respectively, as query refinement modules of the proposed query refinement framework. We compared the effectiveness of the two pairs of systems. First, we compared the performance between MI and P-MI and then compared the performance between CMI and P-CMI. LTI is a framework of utilizing latent topic consistency within a query to re-rank candidate query list, which doesn't consider task information in modeling user's search interests. MTP is a framework of matching search task for personalization, which consider the task information but doesn't examine user's search interests in a search task. Note that we adopted query scoring and applied noise filtering, respectively, in each method.

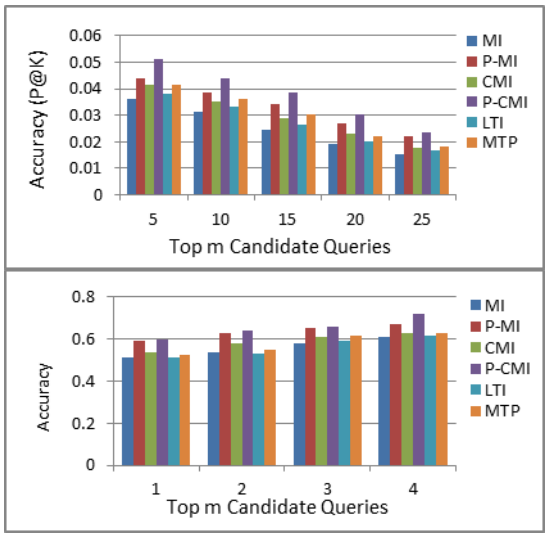


Fig-7. Comparison of Scoring Performance among MI, P-MI, CMI, P-CMI, LTI, and MTP

We generated user profiles by randomly selecting 400 users with more than 50 sessions in our AOL training set. A total of 100 user sessions were used in the parameter determination experiment, whereas the sessions of the other 300 users were used to compare the effectiveness of the two pairs of systems mentioned above. In the second experiment, the first 25 sessions of each user were used to set up the initial task-based interests of users, and the next 25 sessions were used to learn the long- and short-term user interests and evaluate the effectiveness of the system. Note that for each session, the last

query with at least one clicked document was used as the satisfied query for evaluation.

Figure 7 shows the performance of two traditional query refinement methods (i.e. MI and CMI), proposed personalized methods (i.e. P-MI and P-CMI), and two baseline personalized methods (i.e. LTI and MTP). As is shown, proposed P-MI and P-CMI perform much better than the baselines. Specifically, P-MI and P-CMI perform much better than the LTI baseline. For example, the accuracy values (at position 1) of P-MI and P-CMI were 0.59 and 0.60, whereas that of LTI was 0.50. The P@5 value of P-MI and P-CMI were 0.043 and 0.051, whereas that of LTI was 0.038. Similarly, the MTP outperforms other three baseline method including MI, CMI, and LTI. For example, The P@15 value of MTP was 0.030, whereas those of MI and CMI were 0.023 and 0.029. However, it doesn't perform as well as proposed P-MI and P-CMI. Particularly, the improved accuracy of the top four candidate queries is not evident because it doesn't consider user's dynamic interest change within a search task.

5 Conclusions and Future Works

In this paper, we study the problem of personalizing query refinement on the basis of user queries and past click history. In specific, we introduce a four-descriptor model to learn and update the dynamic long-term and short-term user interests. And we propose a graphical model to scoring the candidate queries regarding individual's search interests by modeling the dependency among keywords of candidate queries and their latent tasks. Finally, we illustrated a framework involving proposed user profiling and candidate query scoring modules. Our experimental results show that the latent task based user modeling method could increase the accuracy of query refinement better than the baseline systems could.

6 References

[1] J. Guo, G. Xu, H. Li, and X. Cheng, "A unified and discriminative model for query refinement," Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, Singapore, Singapore, 2008.

[2] F. Qiu and J. Cho, "Automatic identification of user interest for personalized search," Proceedings of the 15th international conference on World Wide Web, Edinburgh, Scotland, 2006.

[3] J. Luxenburger, S. Elbassuoni, and G. Weikum, "Task-aware search personalization," Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, Singapore, Singapore, 2008.

[4] J.-w. Ahn, P. Brusilovsky, J. Grady, D. He, and S. Y. Syn, "Open user profiles for adaptive news systems: help or harm?," Proceedings of the 16th international conference on World Wide Web, Banff, Alberta, Canada, 2007.

[5] D. H. Widyantoro, T. R. Ioerger, and J. Yen, "Learning user interest dynamics with a three-descriptor representation," Journal of the American Society for Information Science and Technology, vol. 52, pp. 212-225, 2001.

[6] A. Ahmed, Y. Low, M. Aly, V. Josifovski, and A. J. Smola, "Scalable distributed inference of dynamic user interests for behavioral targeting," Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, San Diego, California, USA, 2011.

[7] L. Bing, W. Lam, and T.-L. Wong, "Using query log and social tagging to refine queries based on latent topics," Proceedings of the 20th ACM international conference on Information and knowledge management, Glasgow, Scotland, UK, 2011.

[8] X. Wang and C. Zhai, "Mining term association patterns from search logs for effective query reformulation," Proceedings of the 17th ACM conference on Information and knowledge management, Napa Valley, California, USA, 2008.

[9] Y.-S. Chang, K.-Y. He, S. Yu, and W.-H. Lu, "Identifying User Goals from Web Search Results," Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence, 2006.

[10] R. W. White, P. Bailey, and L. Chen, "Predicting user interests from contextual information," Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, Boston, MA, USA, 2009.

[11] A. Kotov, P. N. Bennett, R. W. White, S. T. Dumais, and J. Teevan, "Modeling and analysis of cross-session search tasks," Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, Beijing, China, 2011.

[12] D. He, A. Göker, and D. J. Harper, "Combining evidence for automatic Web session identification," Information Processing & Management, vol. 38, pp. 727-742, 2002.

[13] A. Hassan, R. Jones, and K. L. Klinkner, "Beyond DCG: user behavior as a predictor of a successful search," Proceedings of the third ACM international conference on Web search and data mining, New York, New York, USA, 2010.

[14] A. Hassan and R. W. White, "Task tours: helping users tackle complex search tasks," Proceedings of the 21st ACM international conference on Information and knowledge management, Maui, Hawaii, USA, 2012.

[15] M. Ji, J. Yan, S. Gu, J. Han, X. He, W. V. Zhang, et al., "Learning search tasks in queries and web pages via graph regularization," Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval, Beijing, China, 2011.

[16] J. He, M. Bron, and A. P. d. Vries, "Characterizing stages of a multi-session complex search task through direct and indirect query modifications," Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, Dublin, Ireland, 2013.

[17] D. H. Widyantoro, T. R. Ioerger, and J. Yen, "An adaptive algorithm for learning changes in user interests," Proceedings of the eighth international conference on Information and knowledge management, Kansas City, Missouri, United States, 1999.

[18] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, "Rank aggregation methods for the Web," Proceedings of the 10th international conference on World Wide Web, Hong Kong, Hong Kong, 2001.

[19] D. Downey, S. Dumais, D. Liebling, and E. Horvitz, "Understanding the relationship between searchers' queries and information goals," Proceedings of the 17th ACM conference on Information and knowledge management, Napa Valley, California, USA, 2008.

Positive Unlabeled Learning to Discover Relevant Documents Using Topic Models for Feature Selection

Mingzhu Zhu, Chao Xu, and Yi-Fang Brook Wu
Department of Information Systems, New Jersey Institute of Technology, Newark, New Jersey, USA

Abstract – *Search by Multiple Examples (SbME) is a new search paradigm proposed recently to overcome the shortcomings of the keyword-based search. It allows users to provide multiple positive documents to express their information needs. Traditionally, documents are treated as vectors, of which the features are keywords in the collections. Such a term-vector based document representation brings high dimensionality problems when the collection is large. In this research, we propose a framework of using PU learning for SbME using latent topics identified by a topic model for feature dimension reduction. Specifically, we use Latent Dirichlet Allocation (LDA) to reduce the feature dimension of document vectors to a lower dimension of topic vectors. Then the procedure of discovering relevant documents using a PU learning method is conducted in the topic space. Using Mean Average Precision (MAP) and precision at 10 (P@10) measures, experiments on two benchmark datasets indicate that the topic based method has comparable retrieval performances with the term based method in terms of MAP; more importantly, the former outperforms the latter significantly in term of P@10, when the size of positive examples is small. Given that the LDA method has a significantly smaller dimension than the term based method, it is more practical in a SbME setting, where computational efficiency is crucial in providing real time update of search results per the user's query documents.*

Keywords: Information Retrieval, Positive Unlabeled Learning, Text Mining, Search by Multiple Examples

1 Introduction

Keyword-based search is one of the most widely used search paradigms, but it does not usually fulfill users' information needs, because it is not easy for users to compose the right queries using a simple string of keywords. The main reason for the shortcomings of the keyword-based search is that it is often difficult for users to express their information needs as a simple string of keywords. To overcome this problem, a new search paradigm named Search by Multiple Examples (SbME) [1][2][3][4] has been proposed to enable users to express their information needs using multiple relevant examples (denoted as query examples).

Most of the previous studies on SBME adopt the transductive Positive Unlabeled learning (PU learning) techniques by treating the query examples as positive training data P and the documents in the entire data collection as unlabeled data U. There are two stages involved in these methods: 1) document preprocessing, 2) using PU Learning algorithms to rank the unlabeled data. In the first stage, documents are usually transformed into term vectors after feature selection and feature weight determination. In the second stage, PU Learning algorithms are applied on the prepared data (i.e. term vectors) for learning classifiers and making prediction on the unlabeled data. Thus, the key for a given PU learning algorithm to achieve good performance is to select a set of good features and use appropriate feature weighting methods.

From a machine learning perspective, feature selection is one of the basic problems of documents representation [5], which aims at extracting a small subset of features from the problem domain to retain the fundamental information of the documents while getting rid of the redundant, irrelevant or even ambiguous features. We believe feature selection is of vital importance in the SbME scenario using transductive PU learning. As the main goal of transductive learning is not about learning a model for generalization; instead, it is about learning a model for each dataset of interest. As a result, the learning and prediction process must be very efficient. So it is crucial to identify a small number of features to represent the documents, as a high number of features will bring the curse of dimensionality problem.

In a comparative study of feature selection methods in statistical learning of text classification, Yang and Pedersen [5] evaluate document frequency (DF), information gain (IG), mutual information (MI), $\chi 2$ (CHI) and term strength (TS); and find IG and CHI to be the most effective term-based feature selection methods. In this research, we choose CHI as the baseline feature selection method because of its simplicity and effectiveness.

As the number of terms selected from CHI is still large, we try another method to reduce the dimension size further. As a new method to represent a document as a topic distribution, topic model (e.g. LDA) has received substantial attention from the machine learning and text mining community. In this research, we explore the possibility of using topic model for feature selection as a means to achieve dimension reduction while maintaining comparable search effectiveness.

Latent Dirichlet Allocation (LDA) is one of the most popular topic models that allow documents to have a mixture of topics [6]. It allows sets of documents to be explained by latent topics, which can explain why some terms which are related

to a special topic are similar. Using LDA, we can obtain the topic distribution of a document with the probability that the document belonging to each of the latent topics. Then a document can be represented as a topic vector by using each of the LDA discovered topics as a feature and the probability as the corresponding feature weight. The resulted topic vectors can be used as the input to a PU Learning system.

To accomplish the research goal, we propose a framework of using PU learning for SbME using topic models to perform feature dimension reduction by transforming the document representation from a term vector into a topic vector. We aim to explore whether the latent topics discovered by LDA are effective in calculating the similarity between two documents in a topic level, and whether such topic based similarity calculation can improve the performance of PU learning algorithms. Specifically, we conducted experiments to compare the performance of the PU Learning based SbME system between two feature selection methods: 1) using LDA to represent documents as topic vectors, and 2) using CHI method for feature selection to represent documents as tf-idf based term vectors.

We conducted a set of experiments using two benchmark datasets: Reuters-21578 dataset and WebKB dataset, and adopted Mean Average Precision (MAP) and precision at 10 (P@10) as the evaluation measures.

The remainder of this paper is organized as follows. We begin by describing the related work in PU learning, feature selection and applications of topic models. Then we present the research methods. We then illustrate the experimental framework and analyze the results from the experiments. The article ends with the conclusion and discussion.

2 Related Work

In this section, we review the prior studies most related to our work. Most of them come from two broad areas: positive unlabeled learning and feature selection.

2.1 Positive Unlabeled Learning

Using unlabeled data to improve the performance of supervised learning algorithms has attracted many interests from researchers in machine learning and data mining community [7] [8] Using labeled and unlabeled data for classification is also called semi supervised learning, where positive, negative examples and unlabeled data are needed. Positive Unlabeled Learning (PU learning) is a special case of semi supervised learning. It works on only a positive set P and an unlabeled set U. It aims to identify the hidden positive documents or negative examples from the unlabeled data.

Many techniques about PU learning have been proposed [8][9][10][11]. Liu et al. [9] adopt an EM algorithm and naïve Bayesian classification method to separate positive and negative examples. Nigam et al. [10] use a small set of labeled instances and a large set of unlabeled instances to build a classifier. They show that the classifier based on labeled and unlabeled documents has better performance than those based on a small set of labeled documents alone. Yu et al. [11] propose a mapping-convergence algorithm for PU learning. Liu et al. [13] proposed a technique called Roc-SVM, where- the reliable negative documents are extracted using Rocchio algorithm [14].

Most PU learning algorithms belong to a two-stage strategy category. In the first stage, reliable negatives examples (RN) are identified, and in the second stage, the RN is used to develop classifiers iteratively. Another direction of using PU learning is identifying positive examples from unlabeled data directly.

Some PU learning methods such as bias-SVM [13] are proved to be more effective than others. However, they are not suitable for a SbME system for two reasons. First, they are inductive oriented, which means they are highly dependent on the i.i.d assumption, which means the training data and the test data are independent and identically distributed. However, the documents in an IR system usually come from different sources, so the i.i.d assumption might not be valid. Thus, the transductive learning paradigm is preferable to the inductive one. Second, even though some of them can be used in the transductive scenario, they require parameter selection, which is usually time consuming making them not suitable in the SbME system where efficiency is of vital importance. We believe a simple and efficient PU Learning algorithm is more preferable to the more effective but low efficient PU learning algorithms for SbME. That is why we chose Rocchio classifier as the PU Learning algorithm in this study.

2.2 Feature selection

Many studies on using automatically extracted features and dimension reduction techniques for learning have been proposed [15][16]. Some of them are based on the traditional Document Frequency (DF) and Term Strength (TS) method. Others use part-of-speech (POS) and co-occurrence statistics to calculate the importance of features. The DF and TS based methodologies are simple and easy to implement, but such methods ignore the relationships between terms in the documents thus are weak in dealing with terms which have meanings in different context. Although the POS based methods try to distinguish the meaning of certain terms by considering their context information, they still fail to adopt the semantic relationship between the words in documents to calculate their similarity. While the co-occurrence based methods can retain the semantic information between certain terms in some extent, such semantic relationships may not be captured completely, as the length of the text window for calculating the co-occurrence frequency of terms is fixed and the context information of the terms are usually not utilized.

Other approaches using semantic similarities of terms for document representation have also been proposed. Hofmann

[17] proposes a method using pLSA models to systematically derive semantic representation using Fisher kernels [18], which are combined with a standard vector space representation using a heuristic weighting scheme. Cai et al. [19] propose a three-stage approach of using topic model for feature reduction for text categorization. In the first stage, pSLA [20] is used to derive semantic concepts for document representation over these concepts. In the second step, weak classifiers are constructed using both the term features and concept features. In the third stage, AdaBoost [21] is used to combine the two weak classifiers to integrate the term-based and concept-based information.

Our research is different from the previous studies in that we derive topics for document representation in a PU learning based SbME framework, which is much different from the traditional text categorization system. In addition, we explore the possibility of using latent topics discovered by a topic model to improve the Rochhio algorithm for SbME.

3 Research Framework

In this selection, we describe the proposed framework of applying PU learning for SbME using latent topics identified by a topic model. The framework, illustrated in Figure 1, consists of the following modules and steps.

1) Data collection which is used for training a topic model
2) Training module produces a topic model
3) Users' query examples form the positive data
4) Other search results form the unlabeled data
5) Converting P and U into topic vectors
6) Running a PU Learning algorithm on the topic vectors of P and U
7) Ranking the instances in U using the PU learning algorithm.

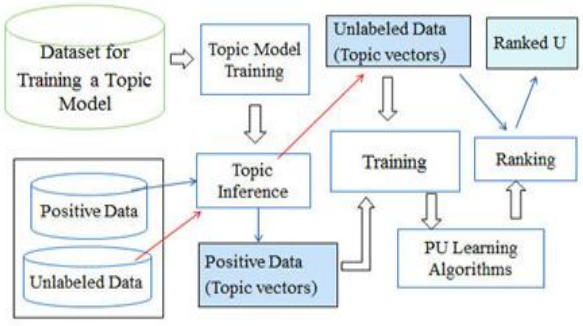


Fig. 1. The Framework of PU Learning based SBME system using topic model for feature selection.

The proposed framework begins in the top left corner where a database stores the document collection. The training module produces a topic model to represent the document collection. The query examples or the positive feedback documents from the user form the positive data P and the rest of the search results form the unlabeled data U. All the documents in P and U are transferred into topic vectors using the trained topic model. Then a standard PU learning algorithm is applied on the topic vectors such that the documents in U will be re-ranked.

Since the topic vectors are ultimately used for PU learning, the dataset for training a topic model should be carefully selected such that the topic inference module can make predictions that reflect the true nature of the documents in P and U. The criteria to select the topic model training data include: 1) it must be large enough to contain as many topics as possible; 2) it should be able to represent the documents in the collection of search system. The selection of the training data is usually done by experts based on the collection in the system. For a search system that contains a large set of documents that are from different domains, the training data should be collected from as many domains as possible. Another strategy is to use a dataset that contains almost all of the domains. For example, for a normal search system, a subset of Wikipedia can be used for the topic training. If no such a collection is available, an alternative is to use all of the documents in the retrieval system to train the topic model. In this research, we take this strategy. We adopted all of the documents from the retrieval system for the topic model training.

In the module of topic model training, a pSLA or LDA model can be adopted. We chose LDA in this research, as it has a more complete document generation assumption, and it has been shown as more effective than pSLA. We will briefly describe LDA in the next section.

Once an LDA model is trained, it is used for topic inference for each document in the retrieval system. Actually, topic model training and inference is taking place offline. Therefore, it is unknown whether a document to be processed belongs to P or U. We use P and U in Figure 1 to illustrate how the proposed framework works.

Another important part of the framework is the PU learning based document ranking subsystem. Important questions include: 1) which PU learning algorithms should be selected? 2) How can we use P and U to discover the positive examples in U? More details about selecting PU Learning algorithms will be described later.

Once a PU learning algorithm is selected, the process of training and ranking the unlabeled data is similar to the traditional methods, where the documents are represented as term vectors.

3.1 Latent Dichilet Allocation

Using statistical topic models to perform text analysis has received much attention in recent years, especially in information retrieval and text mining fields [22][23][24][25]. Griffiths et al [23] and Blei et al. [22] adopt topic models to extract scientific research topics. We adopt a topic model to

extract topics from documents and convert each document into a topic vector. The main difference between this research and the previous ones is that we use topic model to represent documents as topic vectors for PU learning based SbME, which has not been studied before.

One advantage of using topic model for feature selection is that they reduce the dimensionality of feature space significantly. This is important as high dimensionality causes several problems for text mining algorithms [26]. In this research, we chose Latent Dirichlet Allocation (LDA) [6] as the specific topic model. The basic assumption behind LDA is that documents are associated with latent topics, and the corpus is modeled as a Dirichlet distribution of the topics, where each topic is characterized by a distribution over words. Based on this assumption, each document is represented as a probability distribution over some topics, and each topic is represented as a probability distribution over a number of words.

In the preprocessing stage, we used the LDA model to get the topic distributions of each document in the data collection. The results can be used to represent a document as a topic vector, where the topics are the attributes and the probability is the weight for the corresponding feature.

3.2 Positive Unlabeled Learning

After topic training and inference, all the documents in the IR system are represented as topic vectors. In the document ranking stage, it is important to choose appropriate PU learning algorithms. Many PU learning algorithms are available, such as bias-SVM [13], E-spy [8], Rocchio classifier [14]. In this research, we chose Roochio algorithm as the PU learning method because it is the most widely used method in IR and Information Filtering Systems for its high effectiveness and simplicity.

The classic Vector Space Model is the foundation of this research. Each document is represented as a K-dimensional vector where each dimension corresponds to a term when CHI is used for feature selection, or a topic when LDA is used for feature selection. Let $\vec{q}$ denote the query vector, the ranking of documents in U is usually determined by the cosine similarity between a document vector $\vec{d}$ in U and $\vec{q}$. Since we have positive examples P and unlabeled examples U, the simplest method is using the centroid of P as the query vector $\vec{q}$. For instance, the centroid can be calculated using the mean of all vectors in P using the following formula:

$\vec{q} = \frac{1}{|P|}\sum_{i=1}^{|P|} X_i$, where $X_i \in P$.

As the centroid method does not take into account the unlabeled data when building the query vector, its performance is usually poor. Rocchio algorithm computes the query vector as the centroid of P by taking into account both of the documents in P and U using the following formula:

$$\vec{q} = \frac{1}{|P|}\sum_{i=1}^{|P|} X_i - \frac{1}{|U|}\sum_{j=1}^{|U|} X_j,$$

where $X_i \in P$ *and* $X_j \in U$.

Once $\vec{q}$ is obtained, the cosine similarity between $\vec{q}$ and each document vector in U will be calculated. Then U will be ranked based on the similarity value.

3.3 CHI for Feature Selection

To distinguish the positive examples from the negative examples in the unlabeled set, it is important to do feature selection to identify those features of negative examples and positive examples. Feature selection is a process that a subset of the terms in the training set is selected and used as features in text classification [27]. Feature selection is based on such an algorithm that a utility measure for each of the terms to a class is computed and the K terms that have the highest values of the measure will be selected. Other terms that have lower values will not be used in the classification.

Yang et al. [5] show that CHI is one of the most effective feature selection methods in text categorization. In this research, we selected CHI [28] as the baseline feature selection method for getting a set of features for the term based PU learning algorithm. As a popular utility measure for feature selection, the CHI method is applied to test the independence of two random variables in statistics. In feature selection, the two random variables represent the occurrence of the term and the occurrence of the class. The utility measure is calculated by using the following formula:

$$\mathrm{X}^2(t,c) = \frac{N \times (AD - CB)^2}{(A+C)(B+D)(A+B)(C+D)}$$

where A is the frequency of *t* and *c* co-occur, B is the frequency of *t* occurs without *c*, C is the frequency of *c* occurs without *t*, D is the frequency of neither *t* nor *c* occurs, and N is the total number of documents.

In this research, we first conducted a set of experiments using the top K features identified by CHI method. We hypothesize that, with different K being used, the performance of the methods will vary accordingly. When K is too small, the features may underrepresent the documents, thus the performance of the algorithms degrades. When K is too large, not only too much noisy information is included, but also consequently the high dimensionality of the document vectors. Both are detrimental to the performance. We attempt to select the best K for conducting the following experiments.

4 Experiments and Results

4.1 Evaluation Methods

To measure the performance of the ranking system, we adopt the widely used MAP measure [29] to evaluate the performance of the approach. A good ranking means all the relevant (positive) results are in the top ranked positions.

In addition to MAP, we also use precision at k as the evaluation method. In this research, we chose k=10 for comparing the topic based method with the term based method.

4.2 Data Collections

Two benchmark datasets are used for the empirical evaluation. The first one is the Reuters-21578 dataset, which is a commonly used benchmark news article collection in text classification. There are 135 potential topic categories, of which only the most frequent 10 are used as the source of user' query examples, while all the documents are kept to from the unlabeled dataset.

The second dataset is the WebKB collection [12], which contains web pages gathered from university computer science departments. The pages are grouped into seven categories. In this study, we only use the four classes: course, faculty, project, and student, which contain most frequent instances.

4.3 Experimental Design

The experiments were conducted on two benchmark datasets. For each topic of a dataset, the documents belong to it form the positive pool, the remaining documents form the negative pool. For each of the generated datasets, we randomly selected |P| documents from the positive pool as the query examples or positive feedback documents to simulate user's information needs. Then the unlabeled dataset was constructed by randomly sampling |PU| positive examples and |NU| negative examples with the constraints that there is no overlap between PU and P.

A topic with a specific number of |P|, |PU| and |NU| (i.e. |P|=1, |PU|=60, |NU|=1000) forms a unit of the experiment, which results in an AP and a P@10 value. Each unit of the experiments is carried out 10 times, and the average AP and P@10 value is calculated for each topic for a specific number of |P|, |PU| and |NU|. The MAP and P@10 for a specific |P|, |PU| and |NU| is the mean AP and P@10 over the 10 topics.

To reflect the real situation of information retrieval, we keep |PU| much smaller than |NU| and let |P| change from 1 to 30. In both of the experiments of feature selection using CHI and the topic determination for LDA based method, we keep the unlabeled data unchanged, and make |P| change from 1 to 30.

After the best number of features (K) and the number of topics (N) are identified, we conducted additional experiments to see whether the LDA based method outperforms the CHI based method using the best K and N.

4.4 Experimental results

To compare the performance of the topic based Rocchio classifier (TopicRoc) with the term based one (TermRoc), we first conducted experiments to select a set of features using CHI for the term based Rocchio classifier and determine the optimum number topic numbers for the topic based method on the two benchmark datasets. For the term based method, 4000 features are selected for conducting experimental experiments on the Reuters dataset, while 3000 features are selected for conducting experiments on the WebKB dataset. For topic training and topic inference, N is set as 50 for the Reuters dataset and 30 for the WebKB dataset. Then we conducted experiments with α=|PU|/|NU|=20% to simulate the real situations, where the proportion of positive examples in the unlabeled set is small.

For a given unlabeled dataset with a specific α, the size of the query examples changes from 1 to 30. Both MAP and P@10 are recorded. The experimental results are shown in Figure 2 to Figure 5, where TopicRoc (N=50) denotes the topic based method using 50 as the topic size.

From Figure 2 and Figure 3, we can see that the two methods have comparable performance in terms of MAP. When |P| is smaller than 10, the two methods have almost the same performance. From Figure 4 and Figure 5, where the labels on the X axis denote the numbers of positive feedback documents, we can see that the topic based method performs as well as or even better than the term based method in terms of P@10. For instance, from Figure 4, we can see that the topic based method performs better than the term based method when |P|=1, 20 and 30. And from Figure 5, we can observe that the topic based method outperforms the term based method in terms of p@10 when |P|=1, 3, 5 and 30. Since users usually pay more attention on the top ranked results, the P@10 is more useful in evaluating the performance of a ranking system when the size of the potential relevant documents is large. The experimental results indicate the effectiveness of adopting topic model for feature dimension reduction in the PU learning based SbME framework.

We also conducted experiments when α is smaller (i.e., α=10%). The similar observations are also obtained. We also notice that with the increase of α, which indicates more positive examples are included in the unlabeled set, the performance of both methods tends to increase.

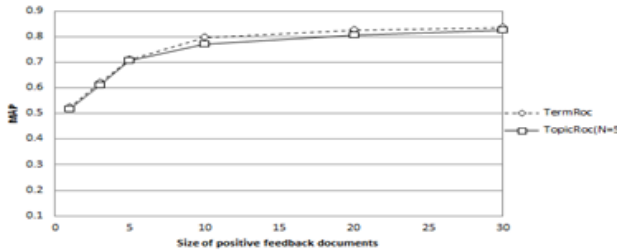


Fig. 2. MAP under different numbers of positive feedback documents on Reuters dataset. (α=20%).

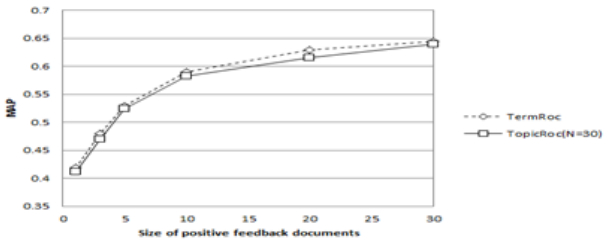


Fig. 3. MAP under different numbers of positive feedback documents on WebKB dataset. (α=20%).

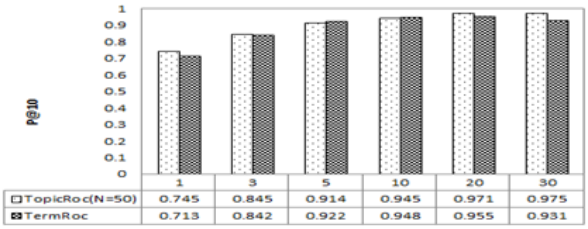


Fig. 4. P@10 under different numbers of positive feedback documents on Reuters dataset (α=20%).

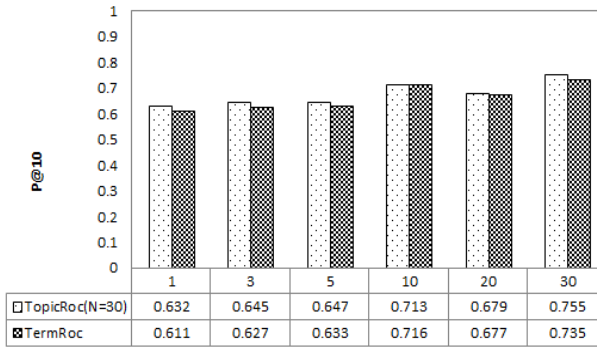


Fig. 5. P@10 under different numbers of positive feedback documents on Reuters dataset (α=20%).

So in practice, reducing the proportion of negative examples in the unlabeled set may lead to the improvement of the system performance.

Since much less features are used in the topic based method than in the term based method, the TopicRoc should be more efficient than the TermRoc. For instance, for the Reuters dataset, only 50 features are used for the topic based method, but 4000 features are used for the term based method. We conducted experiments to show the difference of the efficiency between the two methods on the two datasets. We set |PU|=60 and |NU|=1000 for generating different datasets for evaluation. For the topic based method, all the documents are transferred into topic vectors beforehand. And the algorithms were implemented in a uniform language (Java). Table I shows the results of the experiments, which were conducted on a DELL E6430 Laptop with 2.7GHZ Intel Core i5 CPU, 4GB main memory and windows 7 operation system. We can see that the topic based method took about as half the time as the term based method did. In practice, the number of features used in a real IR system is much larger than the number of features used here, which suggests that the topic based method is more efficient than the term based method in reality.

TABLE I
COMPUTATION EFFICIENCY COMPARISONS BETWEEN THE TOPIC BASED METHOD AND THE TERM BASED METHOD, WHERE R DENOTES THE REUTERS DATASET AND W DENOTES THE WEBKB DATASET; THE TIME UNIT IS MILLISECOND (MS).

	|P|=5		**|P|=20**		**|P|=30**	
	R	**W**	**R**	**W**	**R**	**W**
TopicRoc (N=50)	28.3	25.2	29.6	27.5	29.9	27.9
TermRoc	74.6	65.3	79.1	67.4	84.5	72.8

5 Analysis and Conclusions

The experiments on two benchmark datasets indicate that the proposed method using topic model for feature dimension reduction performs as well as or even better than the term based method using p@10 for evaluation. Using MAP measure, the proposed method has a comparable performance with the term based method. Such results indicate the effectiveness of using topic models for document representation in the PU learning based SbME framework.

One advantage of the proposed method is that a small number of features (e.g. K=50 for Reuters dataset) are used for representing the document vector. However, the term

based method requires much more terms (e.g. N=4000 for Reuters dataset) as features. From the perspective of computation efficiency, the proposed method is superior to the term based method. This is especially true in real IR systems, where the size of terms is much larger. The topic based method can be far more efficient than the term based method.

One concern of the proposed method is that it needs topic model training and topic inference of transferring all the documents in the IR system into topic vectors, which requires lots of computing resources. However, the topic model training and inference can take place offline, so it has no effect on the online part of the SbME system.

It should be noted that the dataset for training the topic model is of particular importance of discovering appropriate latent topics for the documents in the IR system. If the dataset is not representative to the documents in the IR system, the derived topic vectors may be misleading. As a result, the topic based PU learning algorithms will have poor results. In

this research, we simply use all the documents from the IR system, and it turns out that this method works well.

Using topic model for documents analysis has attracted lots of interests before. These approaches are different in that how they use the topics from topic models. Our work is an initial of using topic model for document representation in a PU learning based SbME framework. Our research indicates the potential of using topic models to represent documents for doing other tasks such as clustering analysis.

Our contributions of this research include:

1) We propose a framework of PU learning based SbME system using latent topics from a topic model for feature dimension reduction.

2) We conducted experiments to compare the topic based method with the term based baseline method and show that the topic based method has comparable performance with the term based method in term of MAP, and performs as well as or even better than the term based method in terms of p@10.

3) Given the big difference of the number of features used in the topic based method and the term based method, our research shows that the topic based method outperforms the term based method significantly in terms of computation efficiency.

In the future work, we will deploy the system into a digital library and conduct user studies to evaluate its performance further.

6 References

[1] K. El-Arini and C Guestrin. Beyond keyword search: discovering relevant scientific literature. *Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining*, 439-447, 2011.

[2] D. Zhang and W.S. Lee. Query-By-Multiple-Examples using Support Vector Machines. Journal of Digital Information Management. 7(4): 202-210, 2009.

[3] K. Duh and K. Kirchhoff. Learning to rank with partially-labeled data. *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, 251-258, 2008.

[4] M. Zhu, C. Xu, and Y. B. Wu. IFME: information filtering by multiple examples with under-sampling in a digital library environment. *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, 2013.

[5] Y. Yang, and J. O. Pedersen. *A comparative study on feature selection in text categorization*. Paper presented at the Proceedings of 14th International Conference on Machine Learning, 1997.

[6] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022, 2003.

[7] Y. Altun, D. McAllester, M. Belkin. Maximum margin semi-supervised learning for structured variables. In Proceedings of Neural Information Proccessing Systems, 2005.

[8] B. Liu, W. S. Lee, P. S. Yu, X. Li. Partially supervised classification of text documents. In ICML'02: Proceedings of the Nineteenth International Conference on Machine Learning, pages 387–394, 2002.

[9] X. Li, B. Liu. Learning to classify texts using positive and unlabeled data. In IJCAI, pages 587-594, 2003.

[10] K. Nigam, A. McCallum, S. Thrun, T. Mitchell. Learning to classify text from labeled and unlabeled documents. In Proceedings of the Fifteenth National Conference on Artificial Intelligence, pages 792-799, 1998.

[11] H. Yu, J. Han, K. C. Chang. PEBL: Web page classification without negative examples. IEEE-Transactions on Knowledge and Data Engineering. 16(1): 70-81, 2004.

[12] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. *Learning to extract symbolic knowledge from the World Wide Web*. Proceedings of AAAI-98, 1998.

[13] B. Liu, Y. Dai, X. Li, W. S. Lee, and P. S. Yu. *Building Text Classifiers Using Positive and Unlabeled Examples*. Proceedings of the Third IEEE International Conference on Data Mining., 2003.

[14] J. Rocchio. Relevance feedback in information retrieval. In G. Salton (ed.). The SMART Retrieval System: Experiments in Automatic Document Processing, 1971.

[15] Y. Yang. Noise reduction in a statistical approach to text categorization. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 256-263, 1995.

[16] Y. Zhang, B. Feng. A Co-occurrence based Hierarchical Method for Clustering Web Search Results. In Proceedings of International Conference on Web Intelligence, 407-410, 2008.

[17] T. Hofmann. Learning the similarity of documents. In MIT Press, editor, Advances in Neural Information Processing Systems, volume 12, 2000.

[18] T. S. Jaakkola, D. Haussler. Exploiting generative models in discriminative classifiers. In In Advances in Neural Information Processing Systems, volume 11, pages 487-493, 1999.

[19] L. Cai and T. Hofmann. Text Categorization by Boosting Automatically Extracted Concepts. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, Canada, 2003.

[20] T. Hofmann. Probabilistic latent semantic indexing. In Proceedings of the 22nd ACM-SIGIR International Conference on Research and Development in Information Retrieval, pages 50-57, 1999.

[21] R. E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. Machine Learning, 39(2/3):135-168, 2000.

[22] D. Blei, and J. Lafferty. Correlated topic models. Advances in Neural Information Processing System. 2005.

[23] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In Proceedings of KDD'04, 306-315, 2004.

[24] T. L. Griffiths, M. Steyvers. Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America, 101 (Suppl 1):5228–5235, 2004.

[25] Xin Wei, W. Bruce Croft. LDA-based document models for ad-hoc retrieval. In proceedings of SIGIR'06,178-185, 2006.

[26] H. Kriegel, P. Kröger, A. Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering, ACM Transactions on Knowledge Discovery from Data. 3 (1): 1–58, 2009.

[27] G. Forman. An extensive empirical study of feature selection metrics for text classification. Journal of Machine Learning Research, pages 1289–1306, 2003.

[28] B. Liu. Web Data Mining: Exploring Hyperlinks, Contents and Usage Data. Springer, 2006.

[29] E. Agichtein, E. Brill, S. Dumais. Improving web search ranking by incorporating user behavior information. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pages 19-26, 2006.

A Framework for Flexible Educational Data Mining

Kyle DeFreitas and Margaret Bernard,
Department of Computing and Information Technology
University of the West Indies, St Augustine Trinidad and Tobago

Abstract — Educational Data Mining (EDM) focuses on tools and techniques for discovering previously unknown patterns in the data generated by the educational process. Despite the increased awareness of data mining, the adoption among educators has been slow because the data mining tools are complex and the use of the tools requires a detail understanding of the parameters and requirements of the algorithms used. In this paper, we present a framework for developing an educational data mining environment that is flexible in that it caters for educators who have little technical skills as well as for more advanced users with some data mining expertise.

Keywords: Educational Data Mining, Learning Management Systems

1 Introduction

THE increased use of technology in education has brought about a fundamental change in the way educational institutions operate. The increased usage of electronic based systems has amplified the amount of data available for making better decisions, and improvements in the data mining algorithms make analysis of this volume of data easier and more accessible. In recent years, there has been tremendous interest and research in the field of Educational Data Mining [1].

The use of Learning Management Systems (LMS) are increasingly popular within e-learning environments [2]. The offerings vary from open source solutions such as the Modular Objective Oriented Development Learning Environment (Moodle) [3], commercial solutions such Blackboard [4] and software as a service systems such as Edmodo [5]. However despite their popularity and the universal desire for useful information, LMS are not designed to facilitate analysis using data mining techniques [6]. This compounded with the unique challenges of applying data mining techniques within the educational context has seen a low adoption of data mining among stakeholders [1],[7]. This paper proposes a framework that allows developers to build a data mining environment that is flexible to the level of knowledge and skill of the educator.

This paper is structured as follows. Section 2 briefly explains the challenge of adoption of data mining within educational environments and describes two approaches for user interface tools catering to technical and non-technical educators. In Section 3, the proposed framework is presented and clarified. Section 4 discusses an example of an implementation of the proposed framework while section 5 concludes the paper and provides suggestions for future works.

2 Flexibility in Educational Data Mining

Data mining is the extraction of relevant and useful information from a relatively large dataset [8]. Educational Data Mining (EDM) is the application of different techniques and algorithms with the focus on discovering unknown patterns in data generated by the educational process [9]. Learning Management Systems log all interaction of users (learners and educators) therefore every click and action performed by users is stored in the LMS database. This vast quantity of data becomes a 'gold mine' for extracting interesting and useful information, beyond the simple statistical reports that are provided by popular LMS. The process is not straightforward as EDM techniques must consider the unique pedagogical and semantic characteristics of the data that is extracted and stored [10].

Educators in particular are often required to have expert knowledge of Data Mining to set up parameters and configurations that can be used by the Data Mining process. They need to select the tables that they are interested in, apply the right filters and configurations for modeling the Data Mining tasks, understand enough of the DM techniques to be able to make a choice amongst methods as to which method is the most suitable in the context, and even how to interpret the results.

A number of researchers have tackled educational data mining from different directions. In [6], the authors propose a data model to structure and export usage data stored by the LMS. All EDM techniques must do some preprocessing of the raw log data from the LMS so this is an important step in simplifying the whole process. In [11], the authors apply a classification techniques to predict student performance. EPRules [12] provides a visual means of identifying associative rules within an Adaptive Hypermedia Architecture Course. TADA-ED [13] combines visualization and data mining to analyze web logs from we based courses. MultiStar [14] utilizes data warehousing and data mining resources for supporting distance learning courses within universities. GISMO [15] visualizes students usage data

extracted from Moodle which allows educators to assess where issues may exists and what topics may be potentially problematic. Moodle Mining Tool [16] allows researchers to apply data mining analysis such as clustering and associative rule mining with the usage data contained in the logs of the LMS and eLAT [17] which allows users to choose from a set of indicators related to the hypothesis they would like to test.

Each of the tools previously highlighted makes assumptions about the user's ability and competence with analysis. Therefore we considered the question, can the process at the user end by simplified, discoverable and explanatory?

This paper describes an EDM system, called FLexEDM that was developed as a plug-in to Moodle LMS and, based on that system, an EDM framework that provides flexibility for educators with different levels of technical expertise is proposed. The Educator is provided with options that allows the utilization of Data Mining techniques through either the use of predefined questions or through a guided Data Mining process.

Fig 1. User interaction flow for non-technical user

In FlexEDM the non-technical educators are presented with pre-defined questions on student performance such as:

1. Is there a relationship between the grade on a given assignment and the final grade that students achieve in a course?
2. Is there a relationship between assignment performance and Resource usage?
3. If a student does well on a particular question on a quiz, are they more likely to pass the course?
4. Can students be classified based on assignment performance?

These questions are developed by educators knowledgeable in Data Mining. The system presents these questions to the non-technical educator who can simply select a question of interest for the given course. The system then selects the appropriate Data Mining technique, performs the configure procedures using the stored settings and presents the results to the user. However the selection of the techniques and the configuration details are hidden from the user. The steps involved in performing this type of analysis is highlighted in figure 1.

This approach provides a place for the educator to start as often educators are not sure what questions to ask as they are unsure what the system can provide. Frequently re-used questions developed by other educators are added to the list so it becomes a growing repository of questions that gives educator a 'pot of gold' for analysis of student performance.

Alternatively, for educators who want to perform more exploratory analysis and who have some knowledge of Data Mining, a guided Data Mining Discovery option is presented to them. This gives the educator the ability to explore different configurations to determine any value that may exist within their data. The steps involved in performing this type of analysis is highlighted in figure 2. Educators can first select from different DM techniques supported by the Data Mining engine, including various Classification, Clustering and Association techniques. They can select what dimensions/activities to include in the analysis; this includes assignment, forum, questions, quiz, wiki, attendance and any other resource that are supported by the LMS. Educators are also guided through the process of modeling the Data Mining problem and applying filters such as discretize to convert numeric attributes in the dataset to nominal attributes. This provides a wealth of exploration for educators and the accompanying descriptions on their choices allows educators to learn while performing data mining tasks.

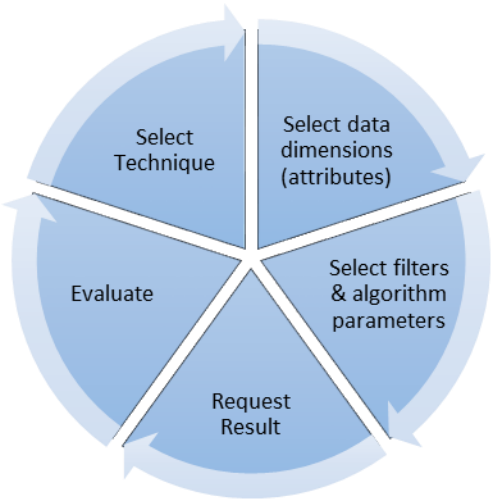


Fig 2. User interaction flow for expert user

3 Framework for Flexible EDM

The literature shows a growing number of EDM systems being developed by various researchers [1]. While many of them are external to the LMS there is an increased focus on developing tools that are integrated and provide useful insights within the learning environments. Utilizing the experience from FlexEDM, we present a framework for the development of an EDM environment. The framework was developed as a representation of the set of tasks that are performed among the various educational data mining systems. Its contribution is to reduce the level of effort among developers allowing them to focus on the specific section of interest rather than requiring the redevelopment of the entire application architecture for every research question within educational data mining.

The flexibility of the framework is based on its ability to

cater for multiple types of users; here we focus on educators who are expert users as well as educators who are non-technical users. The system facilitates this by decoupling the interaction of the user from the other components of the system. These interfaces can then be modified and improved to increase the user-friendliness of the interface for the specific types of users.

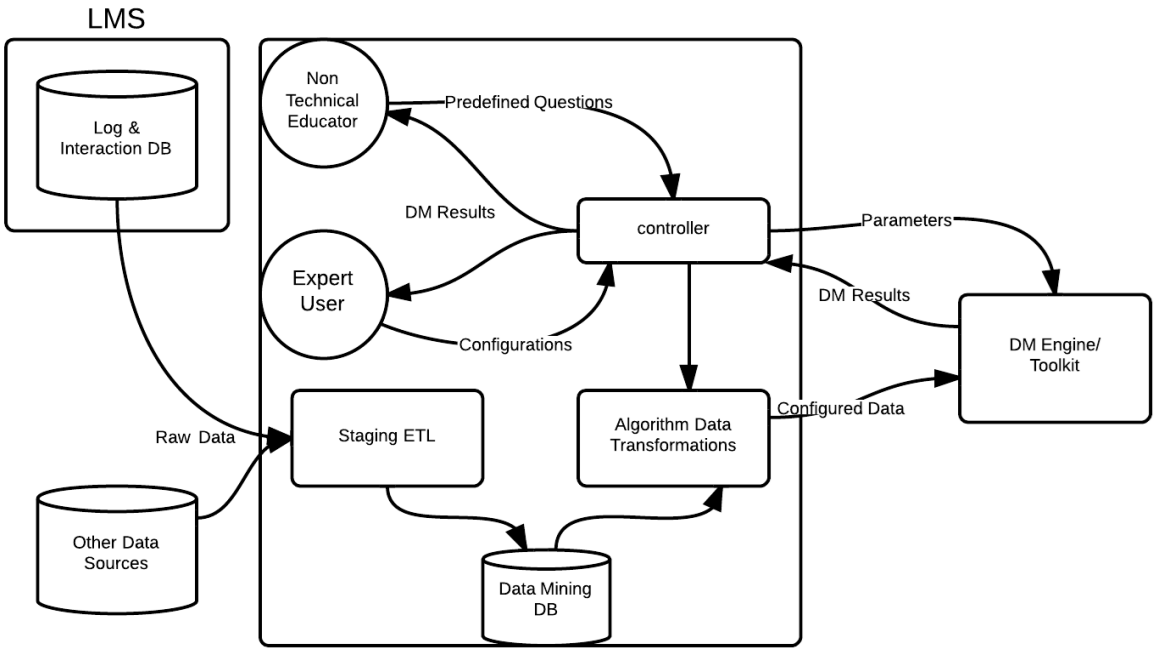


Fig 3. Framework for development of a Flexible Education Data Mining Application

Each component of the framework as illustrated in Figure 3, is pluggable and interact with each other via a standardized interface. This allows each component to be swapped and changed without adversely affecting the functionality of the other components. The following describes each of the components by highlighting the functionality of each.

3.1 Non-Technical Educator

The non-technical educator user interface component provides the elements for educators to attempt to extract meaningful trends from within the data. This component consists of pre-configured parameters and algorithms which are displayed to the user via questions. These questions provide the user with the ability of performing the analysis without knowledge of the underlying algorithms, data transformation or parameters for configurations. The non-technical educator component will return the results of the algorithm as well as additional information that indicates the performance of the algorithm with the data supplied. The additional data will help educators determine the extent to which they can rely on the accuracy of the result produced by the analysis. The choices of techniques were based on work covered by [18] and [19].

3.2 Expert User

The expert user component provides granular control of the data mining algorithm. The interface provides the user with the data mining task (association rule mining, classifications and clustering) and the options and filters that correspond to each of the respective tasks. The data mining results as well as the performance data is returned which allows the user to determine how to continuous modify the parameters of the algorithms in order to discover new insights.

3.3 Controller

The Controller acts as the intermediary between the various components within the framework. It coordinates and controls the passing for data and results within the system to facilitate the data mining analysis. The controller will receive the commands from the user-based components and invoke the appropriate transformation and send the data to the data mining engine based on the technique that the user selects. The controller will receive the results and pass the results to the user interface elements which will translate the results into the appropriate representation for the user.

3.4 Staging ETL

The staging ETL contains the necessarily logic and procedures for extracting the information from the LMS or e-learning system that is being analyzed and storing that preprocessed data in the Data Mining DB. This component will handle some of the preprocessing by transforming the data that may be distributed across multiple tables within the LMS into more logical groupings that make the final extraction for analysis much easier. This component in

addition to the Data Mining Database is based on an extension of a proposed model that attempts to make the data mining analysis of educational data more consistent across the various LMS platforms [6].

3.5 Algorithm Data Transformations

The Algorithm Data Transformation is responsible for converting the data into the required form that it needs for processing. Different algorithms have different requirements such as whether data can accept text, nominal or only numerical data. This component will handle the transformation in a way that is easily extendable. This allows future developers to add new algorithms and their corresponding transformations without changing the overall structure of the application.

4 Implementation

This section highlights how the component of the proposed framework is represented in the application developed.

FlexEDM uses the WEKA toolkit [20] as the data mining engine. Two techniques from the classification, clustering and association rule mining categories of analysis were selected. They were selected based on common usage [18] [19] and availability within the WEKA toolkit. The C4.5 decision tree and the Simple Bayes Network performed classification analysis, the Apriori and the FP-Growth algorithms were selected for association analysis, while the Simple K Means and the Simple Expectation Maximization (EM) algorithms were used for clustering analysis. LMS are varied in the environment in which they are developed, however the majority run over the web. Therefore a REST-based wrapper was developed to allow external entities to access the analytical capabilities of the toolkit. The system upon receiving the data for analysis will perform the following steps using the Java API for WEKA:

1. Set the class index of the data,
2. Apply the selected filters for the desired data transformation before analysis,
3. Develop the model based on the technique specified using 10-fold cross validation to reduce data over-fitting,
4. Evaluate the results acquired from the model and
5. Send the results including the report of evaluator to the requesting controller from the framework.

Though the data format and the required attributes or fields for each of the analysis types and the specific technique is different the process or sequence of activities for generating the model and performing the analysis is constant.

The Staging ETL was developed for the Moodle LMS and extracted the information from the multiple set of Moodle tables and store the results in the Data Mining Database.

For the non-technical educators the questions discussed in section 2 were presented for selection. The system performed an association rule mining using the Apriori algorithm with the usage logs for question 1, 2 and 3 while it performed the clustering analysis using the simple k-means algorithm for question 4. For the selection of the associative rule mining, the system used statistics to determine the correlation between the dimensions in the question and presented a list of generated rules with their respective confidences. For each question an explanation of the analysis was given to help the user better understand what decisions can be made from the question asked and how best to interpret the results received.

For example the question "*Is there a relationship between the grade on a given assignment and the final grade that students achieve in a course*" with use the Apriori algorithm to determine if any relationships exists. The algorithm is configured with a support threshold of 60% and a confidence threshold of 85%. Performing the rule analysis on the raw grades due to their continuous nature will not produce useful results, therefore all of the grades, including the final grade were discretized and the analysis performed on the nominal groups created from this process. The rules created and their respective support and confidence were then displayed to the user for evaluation.

For the expert user the system provides the user with a series of step by step instructions to allow the selection and configuration of the analysis desired. The first step was the selection of the techniques to be used. The user was presented with the choice between performing classification, clustering or association rule mining.

During the second step the user selected the dimensions for analysis (forums, quizzes and courses) and the filters and parameters to be passed along with the data to the data mining engine.

In using the guided data mining discovery, one educator selected Decision tree Classifier (C4.5 algorithm) to classify students on the assignment dimension splitting the users on the basis of grades on a scale of 0 – 5; another used an association rule mining technique (Apriori algorithm) to produce a list of associations to find possible relationships that may exist between dimensions.

5 Conclusion

The paper proposes a framework that allows developers and researchers to build data mining applications that are flexible to the various user types providing layers of abstractions that speak to a common interface of the application. It further demonstrates the application of this framework utilizing tools that are commonly used for data mining analysis within a LMS that is popular with both institutions and researchers alike. It builds and extends previous initiatives to decouple the data mining process from specific e-learning tools and attempts to provide more intuitive user interfaces that facilitate self-discovery of rules

and trends.

The future work of this framework is the inclusion of feedback. The system should progressively increase the options available to the non-technical users based on the feedback and successes of the expert users. Further work can be extended within the guided steps for the expert user by giving insightful suggestions based on the structure and quality of the data available. Further work can also include the exploration and measurement of different interfaces to determine which encourages the utilization of data mining analysis adoption among non-technical users. Finally, we intend to extend the framework to other classes of users such as students and administrators.

6 References

[1] C. Romero, and S. Ventura, "Educational data mining: a review of the state of the art," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on,* vol. 40, no. 6, pp. 601-618, 2010.

[2] C. Romero, P. G. Espejo, A. Zafra, J. R. Romero, and S. Ventura, "Web usage mining for predicting final marks of students that use Moodle courses," *Computer Applications in Engineering Education,* vol. 21, no. 1, pp. 135-146, 2013.

[3] "Moodle," 12 Apirl 2014, 2014; https://moodle.org/.

[4] "Blackboard," 12 April 2014, 2014; http://www.blackboard.com.

[5] "Edmodo," 12 April 2014, 2014; https://www.edmodo.com/.

[6] A. Krüger, A. Merceron, and B. Wolf, "A Data Model to Ease Analysis and Mining of Educational Data." pp. 131-140.

[7] T. C.-K. Huang, C.-C. Liu, and D.-C. Chang, "An empirical investigation of factors influencing the adoption of data mining tools," *International Journal of Information Management,* vol. 32, no. 3, pp. 257-270, 2012.

[8] J. Han, M. Kamber, and J. Pei, *Data mining: concepts and techniques*: Morgan kaufmann, 2006.

[9] R. Baker, "Data mining for education," *International encyclopedia of education,* vol. 7, pp. 112-118, 2010.

[10] F. Castro, A. Vellido, À. Nebot, and F. Mugica, "Applying data mining techniques to e-learning problems," *Evolution of teaching and learning paradigms in intelligent environment,* pp. 183-221: Springer, 2007.

[11] A. Zafra, C. Romero, and S. Ventura, "Multiple instance learning for classifying students in learning management systems," *Expert Systems with Applications,* vol. 38, no. 12, pp. 15020-15031, 2011.

[12] C. Romero, S. Ventura, P. De Bra, and C. De Castro, "Discovering prediction rules in AHA! courses," *User Modeling 2003,* pp. 25-34: Springer, 2003.

[13] A. Merceron, and K. Yacef, "Tada-ed for educational data mining," *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning,* vol. 7, no. 1, pp. 267-287, 2005.

[14] D. R. Silva, "MTP: Using Data Warehouse And Data Mining Resources For Ongoing Assessment Of Distance Learning."

[15] R. Mazza, and C. Milani, "Gismo: a graphical interactive student monitoring tool for course management systems." pp. 18-19.

[16] O. R. Zaïane, and J. Luo, "Towards evaluating learners' behaviour in a web-based distance learning environment." pp. 357-360.

[17] A. L. Dyckhoff, D. Zielke, M. A. Chatti, and U. Schroeder, "eLAT: An Exploratory Learning Analytics Tool for Reflection and Iterative Improvement of Technology Enhanced Learning." pp. 355-356.

[18] V. Kumar, and A. Chadha, "An Empirical Study of the Applications of Data Mining Techniques in Higher Education," *International Journal of Advanced Computer Science and Applications,* vol. 2, no. 3, pp. 80-84, 2011.

[19] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, and S. Y. Philip, "Top 10 algorithms in data mining," *Knowledge and Information Systems,* vol. 14, no. 1, pp. 1-37, 2008.

[20] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: an update," *SIGKDD Explorations,* vol. 11, no. 1, pp. 10-18, 2009.

SESSION

LATE BREAKING PAPERS AND POSITION PAPERS: DATA MINING

Chair(s)

Dr. Robert Stahlbock

Author Attribution of Thomas Paine Work

Smiljana Petrovic, Gary Berton, Robert Schiaffino, Lubomir Ivanov

***Abstract*—Thomas Paine is one of the most significant historical figures who, through his political, philosophical, and socio-economic writings influenced – and continues to influence – the course of history. While Paine's major works are widely known, there are period writings of unknown or disputed authorship which may be attributed to Paine.**

The main goal of this project is to develop a methodology for automated authorship attribution, and apply it to documents of disputed origin. The results generated by the software are cross-referenced with facts about Paine's life and work by experts in Paine studies and 18th-19th century literature and history.

The authorship attribution is performed using machine learning software, trained through the use of works of undisputed authorship to recognize unique features of Paine's style compared to other authors of the period. Once trained, the software is applied to documents of questioned authorship, yielding probabilistic results, further verified by human experts. The results have been both surprising and inspiring: For some disputed documents the software strongly points to Paine as the author. In other cases Paine's previously presumed authorship has been refuted. These results will help better understand Paine's impact on literature, philosophy, history, and politics.

Keywords: Authorship Attribution, Thomas Paine, Machine Learning, Interdisciplinary

I. Introduction

THOMAS Paine is an inherently controversial historical figure whose story has been shrouded in disinformation. Sentenced to obscurity after his death by a power structure that feared him, Paine is the most important historical actor to have been marginalized by academia. Barely mentioned by leading historians for over 200 years when writing about the American and French Revolutions, scholarship on Paine was left largely unexplored. Academic interest in Paine began in earnest in the 1960's, but drew initially upon faulty biographies written to discredit Paine or upon over-enthusiastic biographies by Paine supporters. Most early Paine studies were, therefore, inaccurate due to large factual gaps and biased personal opinions. Even good studies of Paine ([1-4]) have suffered from a lack of an exhaustive factual reservoir to draw from. The biographies of Thomas Paine have been hampered by a lack of knowledge of Paine's early life and writings, and that vacuum has been filled with speculations. The difficulty in studying Paine's life and works is compounded by the fact that Paine wrote anonymously until 1791 when Rights of Man appeared. Thus, there exist a number of writings that may have been created by Paine but have never been attributed to him. There are also works which may have been erroneously attributed to Paine or whose authorship is disputed. While Paine's enormous contribution can stand on his major works alone (Common Sense, American Crisis, Rights of Man, Age of Reason, Agrarian Justice), to fully appreciate Paine's significance and impact on literature, philosophy, history, and politics, a clarification of his authorships is essential.

In 2011, the Institute for Thomas Paine Studies - a collaboration between Iona College, New Rochelle, NY and the Thomas Paine National Historical Association - began a multi-directional text analysis project whose main goal is to develop a solid scientific methodology for authorship attribution, and use it to verify the authorship of a number of documents that may have been written by Thomas Paine. The developed methodology is based on rigorous scientific principles, and takes advantage of modern computer technologies and techniques.

While the results generated by the authorship attribution software may be indicative of a strong possibility that a particular paper may or may not be attributed to Thomas Paine, they can never be absolutely conclusive. Thus, once a trend is uncovered by the authorship attribution software, a further verification and cross-reference of the facts is carried out by experts in Thomas Paine studies, American history, and 18th-19th century literature. Among the issues considered are the conformity of the work to the ideological content of the author's other writings and a match with the historical circumstances and personal idiosyncrasies of the author.

This paper focuses primarily on the automatic authorship attribution aspects of the work while highlighting the interaction between the software-based and the human-expert based aspects of the project.

II. Author Attribution

Authorship attribution is the task of identifying the author of an anonymous text or a text whose authorship is in doubt [5]. While many text mining applications analyze the content of a document as an important indicator for classification, authorship attribution usually focuses on the

DMIN'14: LATE BREAKING PAPER

Smiljana Petrovic is with Iona College, 715 North Av. New Rochelle, NY, 10801 (email: spetrovic@iona.edu, phone: 914-740-4620)
Gary Berton is the Coordinator of the Institute for Thomas Paine Studies at Iona College, New Rochelle, NY, USA, and the historian for the Thomas Paine National Historical Association, 715 North Av. New Rochelle, NY, 10801 (email: gberton@iona.edu, phone: 914-633-2648)
Robert Schiaffino is with Iona College, 715 North Av. New Rochelle, NY, 10801 (email: rschiaffino@iona.edu, phone: 914-633-2338)
*Lubomir Ivanov (**Contact Author**) is with Iona College, 715 North Av. New Rochelle, NY, 10801 (email: livanov@iona.edu, phone: 914-633-2342)*

style of the document rather than its contents; typically, all candidate authors write about similar topics and use similar, topic-specific words and phrases. However, stylistic features are often used unconsciously and consistently and, if correctly identified, may correctly reveal identity of the author.

We approach authorship attribution as a classification task: here, to classify a document means to assign it to the class of documents written by the same author. One way to perform classification is through supervised machine learning: Special algorithms use documents of known authorship (training examples) to train the system to recognize each author's writing style. Once training has been completed, the created model can be used to attempt to identify the creator of a document of disputed authorship.

A. Lexical Features

Among the most common "off-the-shelf" lexical features are function words, N-grams of characters and words, and sentence lengths [6].

Function words are the most common words (articles, prepositions, pronouns, etc.) in the English language. Since function words are topic-independent, they are usually excluded from the feature set of a topic-based text classification. However, since function words are often used in a subconscious manner, they well reflect the author's style and are among the best features for authorship attribution. In this work, we used function words as defined by Mosteller-Wallace in their Federalist papers study [7].

Word N-grams and character N-grams are also standard features used in text analysis. Word 2-grams consider sequences of 2 words from a given text. For example, word-2-grams of the text "Author Attribution of Paine and his Contemporaries" are "Author Attribution", "Attribution of", "of Paine", etc. Similarly, character-2-grams consider sequence of 2 characters from a given sequence of characters. For example, character-2-grams associated with the text "Author Attribution" are "au", "ut", "th", etc. N-grams features are simple, language independent and often very effective in text mining applications.

Our approach is to extract the fifty most frequent words from each document. Their union is a pool from which the fifty most frequent words are used to create the vector of features. The normalized vectors of frequencies of those words represent our training examples. For example, the vector of the most frequent functional words in one of our experiments was (to, but, for, no, by, every, has, been, who, of, were, are, more, his, would, any, on, had, be, such, so, or, and, shall, not, that, than, will, this, can, have, one, from, was, if, all, is, with, may, it, a, at, as, the, in, should, which, an, their, our). The normalized vector of frequencies of those words in Paine's "Forester Letters" was (0.07546, 0.01189, …, 0.01276). That vector is labeled as "Paine" and considered one training instance. Vectors of all documents of known authorship represent training data for one experiment.

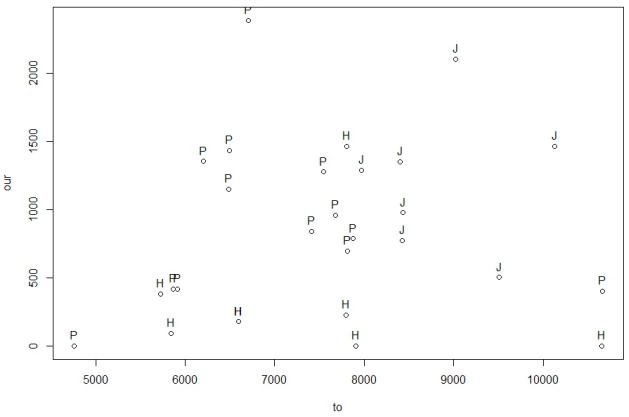


Figure1: Scatter plot of relative frequencies of two functional words, "to" and "our", in the 28 documents written by Paine (P), Jefferson (J) and Hopkinson (H)

Consider, for example a set of twenty eight documents written by Hopkinson, Jefferson and Paine in the two-dimensional space of relative frequencies of the words "to" and "our" (Figure 1). We can see that Jefferson relatively often uses "to", while Hamilton tends to use the word "our" less frequently that the other two authors. Ideally, all documents of the same author would have similar relative frequencies and would be clustered together: they would have small inter-cluster distance and large intra-cluster distance. Our example, however, illustrates that documents are often not clearly separated, and that documents by the same author may have very different feature values.

To attribute a document of an unknown or disputed authorship, a vector of its relative frequencies is created and submitted to the software. The machine learning algorithm attempts to extrapolate from its stored knowledge, and attribute the document to the author with the closest existing vector. In our example from Figure1 an unattributed document represented by vector (7000, 1200) would be probably attributed to Paine and one with vector (9000, 1000) to Jefferson by most methods. However, the classification of a document with vector (8200, 500) would be uncertain and probably different from method to method. Vector proximity is based on specific criteria characteristic of each leaning method. Thus, different learning methods may produce different attributions.

B. Learning methods

Linear Support Vector Machines (LSVM) method seeks a hyperplane in the n-dimensional input space which best separates points corresponding to different candidate authors. The best separator is the hyperplane that maximizes the distance to the closest training data points of different authors. To attribute a disputed document, we evaluate on which side of the hyperplane the point corresponding to that document lays.

Table 1: Features used in our analysis and their descriptions

Style marker	*Description*
MW Function Words	Considers function words as defined by Mosteller-Wallace in their Federalist papers study.
Word *n*-grams	Considers sequence of *n* items from a given sequence of words (we used the values 2, 3 and 4 for n)
Character *n*-grams	Considers sequence of *n* characters from a given sequence of characters (we used the values 2, 3 and 4 for n)
Part of Speech	Marking up a word in a text as corresponding to a particular part of speech; identification of words as nouns, verbs, adjectives, adverbs etc. Uses the Maxent Tagger developed by the Stanford NLP Group [9]
Vowel-initial Words	Considers words beginning with vowels
Sentence Length	The number of words in a sentence
Special Words	Special words selected from the documents that are frequent or have an atypical spelling (e.g. hath, juster, willful)
Prepositions	Most common prepositions
Suffixes	The last 3 letter of every word
First Word in Sentence	First Word in Sentence

Centroid Nearest-Neighbor approaches represent each author by its centroid vector - a vector whose coordinates are averages of coordinates of all training instances. An unknown document is associated with the author with the nearest centroid. Distance can be measured using different metrics. In our work, we used Euclidian distance (L2 metric) and cosine distance (normalized scalar product distance).

In order to check the accuracy of the model, the given documents are usually divided into training and testing sets. The training set is used to build the model, and then the model is tested on the remaining documents. In our work we adopted a "leave-one-out" validation: *n-1* of the available *n* documents are used for training, and validation is carried out using the remaining one document. The procedure is repeated n times, so every document is at some point used for validation. The percentage of correctly classified documents constitutes the "leave-one-out" accuracy of the method.

To further improve performance, we used a weighted sum of supports of different methods for different authors. Each method independently makes a choice (supports one author). We associate with each method a weight proportional to its leave-one-out accuracy. The weighted sum method selects the author with the largest weighted sum of all supports the author received from different methods. In our experiments, the weighted sum usually outperforms any individual method.

Table 2: Learning methods used in this study

Learning Method	*Description*
Support Vector Machine with Linear Kernel	Generates a linear separator to divide the feature space into regions, each corresponding to a specific author
Centroid with Histogram Distance	Nearest-neighbor approach using Euclidian Distance (L2 metric)
Centroid with Cosine Distance	Nearest-neighbor approach using normalized dot product distance

III. Experimental Design and Results

There are two major components that determine accuracy of learning - the set of lexical features considered and the choice of a machine learning (classification) algorithm. We consider sixteen lexical features (Table 1). Some of them are common, "off the shelf" lexical features, widely used in authorship attribution literature, such as function words, N-grams of characters and words, and sentence lengths. We also developed some domain-specific features based on our general knowledge of the documents: set of special words (e.g. "hath", "juster", "willful") and prepositions. For training we use a corpus of sixty-nine documents of ten authors [see Appendix A]. Results are obtained using the JGAAP (the Java Graphical Authorship Attribution Program), open source software [8] and programs written by authors of this paper. We extracted sixteen standard features from the documents and paired each with each of three learning approaches (Table 2). We considered fifty most common values for each feature.

The next table provides precision and recall data for each author. The *precision* of an author is the fraction of documents attributed to him that are indeed his work. The *recall* of an author is the fraction of his documents that were attributed to him. If a classification method attributes all training documents to one author, the recall will be 100%, as the software correctly classified all documents of that author, but precision will be low, as many documents were incorrectly attributed to that author. In the first experiment, we consider sixty-nine documents of ten candidate authors (Table 3). The leave-one-out validation correctly classified 90% documents. Note that if authors were assigned by random guessing, expected accuracy would be 10%. Seven authors (Adams, Benezet, Jefferson, Paine, Price, Priestley and Rush) had all their documents correctly identified (100% recall). Five authors (Adams, Benezet, Hopkinson, Price and Priestley) had 100% precision, indicating that they were associated only with their own documents. Four of them (Adams, Benezet, Price and Priestley) were associated with all of their documents, and with no other documents.

Table 3: Recall and precision of each author when learning was on sixty-nine documents and ten candidate authors. Accuracy of leave-one-out cross validation is 90%.

	Recall	***Precision***
Adams	100	100
Benezet	100	100
Franklin	89	80
Hopkinson	50	100
Jefferson	100	88
Paine	100	80
Price	100	100
Priestley	100	100
Rush	100	86
Witherspoon	71	83

In the next two sections we present experimental results using the selected learning methods and the weighted sum method when different candidate authors were considered for each of two unattributed/disputed-attribution documents. We start with 10 authors in the first experiment, and then select authors based on supports they received in previous experiments. We removed from consideration authors that received less than 10% support and repeated experiment with the narrowed set of candidate authors. The comprehensive list of all documents appears in Appendix A. For the four selected methods we report leave-one-out accuracy and their attribution choice. We choose generally well performing lexical features combined with linear support vector machine based learning. We report the leave-one-out accuracy and choice made by the weighted sum approach. We also provide break out of percentages of supports that each candidate author received from the weighted sum approach. For support we consider only methods that were correct on at least half of the documents in the leave-one-out validation. Ideally, all methods would choose the same author, giving him the 100% support.

A. The Dream Interrupted

Experimental Results

The performance of four experiments with different candidate authors and distribution of support to each author by weighted sum is shown in Table 4. All experiments selected Hokinson as the author of "The Dream Interrupted". The most successful four methods were character 3-grams, functional words, character 4-grams and words 2-grams in combination with LSVM, with accuracies of 87%, 77%, 74% and 72% respectively. Hopkinson was also voted as a probable author by the weighted sum method with support of 43%. In the second experiment, we eliminated from consideration authors that received less than 10% support, and repeated the experiment with only Adams, Hopkinson, Paine and Price. We, then, ran another experiment with Hopkinson, Paine and Price, and finally considered Hopkinson and Paine as the only candidates (Table 5). In all experiments, Hopkinson was selected as the author. Individual methods voted for Hopkinson vs. Paine with 65% support.

Table 4 Accuracy and choice made by four lexical features and weighted sum in experiments with different candidate authors on The Dream Interrupted

Method	***Accuracy***	***Choice***
All 10 authors (69 documents)		
Function words	77%	Hopkinson
Word 2-grams	71%	Adams
Character 3-grams	87%	Hopkinson
Character 4-grams	74%	Hopkinson
Weighted sum	**90%**	**Hopkinson**
Adams, Hopkinson, Paine, Price (32 documents)		
Function words	72%	Hopkinson
Word 2-grams	84%	Adams
Character 3-grams	84%	Hopkinson
Character 4-grams	81%	Hopkinson
Weighted sum	**91%**	**Hopkinson**
Hopkinson, Paine , Price (23 documents)		
Function words	83%	Hopkinson
Word 2-grams	87%	Hopkinson
Character 3-grams	83%	Hopkinson
Character 4-grams	78%	Hopkinson
Weighted sum	**87%**	**Hopkinson**
Hopkinson, Paine (16 documents)		
Function words	75%	Hopkinson
Word 2-grams	75%	Hopkinson
Character 3-grams	81%	Hopkinson
Character 4-grams	75%	Hopkinson
Weighted sum	**81%**	**Hopkinson**

Notice that Hopkinson has very low recall in all of our experiments. We believe that it is due to his experimental writing style that is hard to capture. As the number of training documents decreases, the misclassified Hopkinson's documents account for higher percentage of misclassifications, hence the overall accuracy decreases. However, the percentage of support to Hopkinson increases. More importantly, in all experiments, Hopkinson has 100% precession: every document attributed to him was truly his work. This fact additionally supports the attribution of "The Dream Interrupted" to Hopkinson.

Table 5: Support, recall and precision of each author in experiments with different candidate authors on The Dream Interrupted

	Support	*Recall*	*Precision*
All 10 authors (69 documents)			
Adams	16	100	100
Benezet	2	100	100
Franklin	3	89	80
Hopkinson	**43**	**50**	100
Jefferson	3	100	88
Paine	22	100	80
Price	10	100	100
Priestley	0	100	100
Rush	0	100	86
Witherspoon	0	71	83
Adams, Hopkinson, Paine, Price (32 documents)			
Adams	13	100	100
Hopkinson	**42**	62	100
Paine	25	100	73
Price	20	100	100
Hopkinson, Paine, Price (23 documents)			
Hopkinson	**54**	62	100
Paine	28	100	73
Price	18	100	100
Hopkinson, Paine (16 documents)			
Hopkinson	**65**	62	100
Paine	35	100	73

Human Expert Cross-Verification

The article "The Dream Interrupted" appeared in the Pennsylvania Magazine in May, 1775. Philip Foner described the article as "an interesting example of Paine's ability to use different literary techniques to bring home a vital political message" [2]. While it is true that Paine used different styles to get his message across, this is not one of them.

Our Author Authentication method suggests that this article was written by Francis Hopkinson. Hopkinson was a frequent contributor to the Pennsylvania Magazine, writing under the pen names of B, or the Old Bachelor (also used by others), but mostly unsigned. Hopkinson resided in Bordentown, NJ, where Paine eventually purchased property and spent a great deal of time. The friendship between the two men is well documented, and Paine's ties to Bordentown probably stem from this association at the Pennsylvania Magazine. Hopkinson was a strong advocate of independence, and the politics of "The Dream Interrupted" fit his views as well as Paine's.

The context of "The Dream Interrupted" should have also raised some questions regarding Paine's authorship. The fact that it took place in Bucks County (signed "Bucks County"), which is across the river from Bordentown in Pennsylvania, though not definitive, would not fit in with Paine's movements in the first five months in America. The reference to a "fatiguing journey from Virginia" in the first sentence of the article should also have raised questions, since there is no indication that Paine could have made time to venture such a trip. Bucks County would be the last leg of a trip from Virginia to Bordentown. It would not be part of an itinerary from Virginia to Philadelphia where Paine resided at that time. Paine's physical involvement in Bordentown did not begin until 1778.

Hopkinson also used the device of dreams in other Pennsylvania Magazine articles, such as "The Revery" and "The Extraordinary Dream". Hopkinson's more classical style fits this employment of dreams, as does his sweeping grandiose metaphors. In this period of his life, Hopkinson had set aside his musical compositions to focus on experimental writing styles, hence his frequent contributions to the magazine.

B. The Magazine in America

Experimental Results

Our experiments (Tables 6 and 7) strongly point to Paine as the author of "The Magazine in America". Among the ten candidates, only Paine and Price received a support higher than 10%, with Paine's support of 77% and Price's of 18%. In an experiment with only those two candidates, Paine received overwhelming support of 93%. Among highly weighted methods, the part of speech and character-2-grams supported Price, all others favor Paine. In all other experiments in which Paine was included, the weighted sum algorithm highly supported him.

Human Expert Cross-Verification

Scholars have questioned the authorship of this article which appeared in the Pennsylvania Magazine in January 1775 because it was written soon after Paine arrived in America with an acute, debilitating illness. This was the premiere issue. However, our analysis supports Paine's authorship. Paine was present in Philadelphia, and looking for employment, and his subsequent involvement in the magazine all favor his authorship. The article demonstrates Paine's hands-on role at the magazine, his early selection of pieces to publish, and the editorial position he assumed. This reinforces those scholars who give weight to articles appearing in the magazine as reflective of Paine's political and philosophical leanings.

Table 6: Accuracy and choice made by four lexical features and weighted sum in experiments with different candidate authors on The Magazine in America

Method	*Accuracy*	*Choice*
All 10 authors (69 documents)		
Function words	77%	Paine
Word 2-grams	71%	Paine
Character 3-grams	87%	Paine
Character 4-grams	74%	Paine
Weighted sum	**90%**	**Paine**
Paine, Price (15 documents)		
Function words	93%	Paine
Word 2-grams	93%	Paine
Character 3-grams	93%	Paine
Character 4-grams	93%	Paine
Weighted sum	**100%**	**Paine**

Table 7: Support, recall and precision of each author in experiments with different candidate authors on The Magazine in America

	Support	*Recall*	*Precision*
All 10 authors (69 documents)			
Adams	0	100	100
Benezet	0	100	100
Franklin	0	89	80
Hopkinson	0	50	100
Jefferson	0	100	88
Paine	**77**	100	80
Price	18	100	100
Priestley	0	100	100
Rush	5	100	86
Witherspoon	0	71	83
Paine, Price (15 documents)			
Paine	**93**	100	100
Price	7	100	100

IV. Future Work

The team is currently working to increase the size of our training corpus by adding additional documents written by the authors that we have been studying, and by identifying other authors from this period and including their works. Under our current classification methods, an unknown document will be always attributed to one of the candidate authors whose documents were used during the training. The possibility that a document is authored by someone not among the selected set of authors is not currently supported in the software model. Hence, ensuring the inclusion of the names and representative documents of additional relevant candidate authors is crucial.

Developing new features relevant for authorship attribution is another focus of our work. While we have already considered several well-known lexical features and learning methods, we intend to enhance our current methodology by including new methods for text analysis (e.g. artificial neural networks) and new lexical features such as the use of alliterations.

As the size of the corpus expands and the number of features grows, the computational complexity is expected to increase drastically. Therefore, we are investigating the use of advanced computing techniques so that some of the computations can be carried out in parallel rather than sequentially.

Additional challenges we face in our future work include the reliability of training documents and the influence of incorrectly attributed training documents, which could have serious impact on accuracy of the model and its predictions.

Finally, our future work will consider relationships among authors. We would like to be able to consider not only direct collaborations on some documents, but also evidence of influences of one author on another, in ideas as well as in style. An additional challenge is detecting the influence that editors or even typesetters may have had on final versions of the published documents. We would also like to attempt detecting common characteristics of authors that intentionally experimented with different styles (e.g. Hopkinson).

These challenges necessitate a close collaboration with historians, political scientists, and 18th/19th century literary scholars. The results of our analysis should be considered directions for further study by historians. Only through a close collaboration can the true nature of the life and work of Thomas Paine and his global impact on posterity be truly revealed.

Appendix A

Table 8: A corpus of work of ten authors we used for training. Some of large publications were broken into multiple documents.

Author	*Work*
Adams	Defense
	John Adams Thoughts
Benezet	Guinea (Sans Quotes)
	Some Observations On The Situation (Sans Quotes)
	Caution To Great Britain And Her Colonies (Sans Quotes)
Franklin	Correspondence and essays from 1768 to 1776
Hopkinson	Improved Plan Of Education (Sans Quotes)
	Consolation For The Old Bachelor (Sans Quotes)
	The Ambiguity Of The English Language
	A Revery (Sans Quotes)
	A Prophecy
	A Pretty Story (Sans Quotes)
	Extraordinary Dream
	Translation Of The Letter
Jefferson	Correspondence 76-77

	Correspondence 78-79
	Drafts of documents & Correspondence 78
	On The Instructions Given To The First Delegation Of Virginia To Congress
	Correspondence Post 1780
	Correspondence Pre 1776
	Summary View
Paine	Common Sense
	Crisis Papers
	Forester Letters
	Miscellaneous Articles In 75 And 76
	Of Monarchy And Hereditary Succession
Price	Observations Civil Liberty (Sans Quotes)
	Review Of The Principal Questions In Morals
	Britains Happiness
	Discourse Of Love Country
	Evidence Of Future
	Fast Sermon
	Observations on the Importance of the American Revolution (Sans Quotes)
Priestley	An Essay On The First Principles Of Government (Sans Quotes)
Rush	A Plan For Establishing Public Schools In Pennsylvania
	An Account Of The Life And Death Of Edward Drinker
	An Address To The Ministers Of The Gospel Of Every Denomination In The United States
	Paradise Of The Negro
	Thoughts Upon Female Education
	Thoughts Upon The Amusements And Punishments Which Are Proper For Schools
Witherspoon	Aristides
	On Conducting The American Controversy
	On The Affairs Of The United States
	On The Convention With General Burgoyne
	On The Proposed Market In General Washington
	Reflections
	Thoughts On American Liberty

REFERENCES

[1] Claeys, Gregory, *Thomas Paine: Social and Political Thought. Boston: Unwin Hyman*, 1989.

[2] Foner, Philip, *The Complete Writings of Thomas Paine,* 2 vols. New York: Citadel Press, 1969.

[3] Williamson, Audrey, *Thomas Paine*, New York: St. Martin's Press, 1973.

[4] Fruchtman, Jr., Jack. Fruchtman, Jr., Jack, *Thomas Paine: Apostle of Freedom, Four Walls Eight Windows*, New York, 1994

[5] Love, H., *Attributing Authorship: An Introduction*, Cambridge, Cambridge University Press, 2002.

[6] Stamatatos, Efstathios, *A survey of modern authorship attribution methods,* Journal of the American Society for Information Science and Technology 60, no. 3 (2009): 538–556.

[7] Mosteller, Frederick, and David L. Wallace, *Inference and disputed authorship: The Federalist*, Reading, MA: Addison-Wesley, 1964.

[8] Juola, Patrick, *Authorship Attribution*, Foundations and Trends in Information Retrieval 1, no. 3 (2006): 233-334.

[9] Toutanova, Kristina, Dan Klein, and Christopher Manning. *Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network*. HLT-NAACL, 2003: 252-259.

Identifying Outliers in Human Movement Trajectories Clustered By Hausdorff Distance

Shibin Parameswaran
Space and Naval Warfare Systems Center Pacific
53560 Hull St., San Diego, CA 92152, USA
Email: shibin.parameswaran@navy.mil

Jeffrey Ellen
Space and Naval Warfare Systems Center Pacific
53560 Hull St., San Diego, CA 92152, USA
Email: jeffrey.ellen@navy.mil

***Abstract*—In this paper, we perform outlier detection on human trajectory data via clustering methods using Hausdorff distance. Hausdorff distance proves suitable to measure the similarity between trajectories of different lengths. The trajectories are then clustered based on their pair-wise Hausdorff distances using k-medoids and different versions of hierarchical clustering algorithms. The clustering results obtained from these algorithms are analyzed for their effectiveness based on our evaluation criteria which is in line with human perception of outliers and clusters. We provide evidence that maximum hierarchical clustering provides the best performance. Additionally, our approach is completely unsupervised and does not require domain-specific knowledge.**

***Keywords*—human trajectory analysis, hausdorff distance, outlier detection, k-medoids, hierarchical clustering**

I. Introduction

Our goal is to automatically flag outliers or out-of-ordinary movement patterns displayed by a particular individual within a set of human movement trajectories. We accomplish this through the use of clustering algorithms based on Hausdorff distance to identify individual clusters and outliers in an unsupervised manner.

A. Background and Previous Approaches

With the proliferation of smart-phones and ubiquitous computing gadgets, there is an increased availability of historical (or real-time) location data for individuals using these technologies. Data can come from the portable devices themselves, or externally from the increasing amounts of sensor networks. These trajectories of human movement are different from trajectories studied in previous decades, such as RADAR/SONAR reflectances, AIS transponders, etc. of vessels and vehicles for two reasons. First, the volume of trajectories generated, and second, the physical properties of the quantity being tracked. Vessels have different properties, such as momentum or restricted freedom of movement. For example, maritime motion is generally restricted to shipping lanes, which greatly influences choice of analytical methods when attempting to find outliers [1].

Ge. et al [2], identify outlier trajectories by setting a threshold where the top 'n' instances are defined to be the outlier. Our approach differs in that we allow for a varying number of outliers. Another approach to outlier identification is to build a comprehensive model quantifying 'normal' behavior, and then identify activity falling arbitrarily far from the model's prediction [3]. This approach is less general than our approach, because it requires assumptions about the patterns of movement. Their approach is better suited to smoother, more uniform trajectories, while our approach clusters the data, and then arbitrarily defines a number of clusters which contain the outlier behavior allowing for analysis of non-smooth data input.

Hausdorff distance has previously been used to identify outliers in human movement trajectories by Laxhammar and Falkman. However, they examine a specific security application consisting of only 239 trajectories, all of uniform length (5 data points) [4]. So in their case, no clustering step is required, they simply rank trajectories according to Hausdorff distance. The trajectories in our data set have hundreds or thousands of points, and are of variable length. Zhou et al. [5] and Chen et al. [6] used a similar approach to ours to cluster cyclone tracks which have fewer data points and are much more regular in their trajectories than the human movement data.

Contact Author: Jeffrey Ellen (jeffrey.ellen@navy.mil)

In this paper, we present and compare different human trajectory clustering methods using Hausdorff distance metric. All the methods presented here can be used to compare and cluster trajectories that have different lengths. In particular, we show that hierarchical clustering can be used with Hausdorff distance to identify clusters that are highly unbalanced (large variation in cluster sizes) in nature and can be effectively used for outlier detection applications.

II. Evaluation Dataset

GeoLife Trajectories [7]–[9] is a GPS trajectory dataset published by Microsoft Research. The GPS data is collected and recorded by 182 users over a three year period of time and contains more than 17,000 trajectories covering approximately 750,000 miles. The GPS trajectories contain latitude, longitude and altitude information. The trajectories of this dataset have a variety of sampling rates, but over 90% of the trajectories have a reporting frequency of 5 seconds or less. Additionaly, this dataset captures the complexity of human movement patterns and includes a wide range of user activities such as daily routines (commuting from home to work and vice versa), entertainment (shopping, dining) and sport activities (hiking, cycling). Here, we show representative examples of two users from this dataset in order to demonstrate the applicability of our method.

The GeoLife dataset provides timing information with each data point, which we did not use. This means our technique is generalizable to other trajectory datasets that provide no temporal information.

A. Evaluation Criteria

With no prior knowledge of a person's movement patterns, it is impossible to definitively identify specific trajectories as anomalous. GeoLife trajectories comprises of data collected from regular people over the course of their daily life, but does not provide detailed information about the activities or circumstances that created the behaviors that resulted in the recorded trajectory. Therefore, we do not have ground-truth labels available that clearly identify out of the ordinary behaviours by these users. This prevents us from quantifying outlier detection performance. Another option would be to add synthetic outliers into the dataset. We did not opt for this method because that does not help with the testing and validation of our method on real-world data. Simply inserting random data is insufficient; randomized data cannot be guaranteed to be an outlier in the sense of the true distribution of the unobserved variables being modeled. Inserting synthetic data systematically would only help in determining whether the heuristic we used to generate the outliers was identifyable.

Instead, we use the definition provided by Hawkins in [10], outliers are those observations that deviate *drastically* from others indicating that they were generated by a different underlying mechanism. According to this definition, any movement by a user that digresses from the majority of their other trajectories could be considered anomalous. To this end, we want our methodology to identify patterns that do not fit with the rest of the data and flag them as outliers. For experimental evaluation, the resulting clusters are then visually inspected and verified as acceptable clusters comprising exclusively of outliers and normal patterns. We consider the cases where multiple clusters contain outliers to be less optimal, but still superior to a cluster containing false positives. In other words, if clusters are homogenous, it would be relatively easy to label all cluster members as outliers. However, if clusters are heterogeneous, it decreases usefulness of the clusters.

III. Experimental Approach

In order to group data points into clusters, one has to define a metric such that the value of that metric between two points accurately represents the degree of dissimilarity (or similarity) between them. The common distance metrics used for this purpose are the Euclidean distance, cosine distance, Manhattan distance etc. However, these distance measures require data samples to have the same dimensionality. Hence, these distance measures cannot directly be used on human trajectory data because they almost always have a varying number of GPS co-ordinates. Therefore, trajectories will usually be vectors of different lengths. One potential technique is to compare the distances between the spatial points (GPS coordinates) individually. In this approach, the meaning of a trajectory is lost and the resulting clusters fail to provide any trajectory level information. Another alternative is to find the centroid of each trajectory, and then use Euclidean distance to measure the distance between the two centroids, but this also loses a large amount of information. For example, a 100 point circular trajectory and a 2 point linear trajectory acting as a diameter to the circle would have the same centroid, the center of the circle, but their shape is completely different. A metric which reported zero distance between them would clearly

be flawed. Similarly, clustering algorithms such as k-means cannot be used for trajectory data because the notion of *mean* trajectory is not defined for a set of sample trajectories that are of non-uniform lengths.

In order to address these issues, we used Hausdorff distance for comparing trajectories where we consider each trajectories to be an unordered set of spatial co-ordinates. Likewise, we experiment with clustering algorithms such as k-medoids and variants of hierarchical clustering algorithms that do not require calculation of means or centroids of pairs of trajectories. Below, we provide a brief description of Hausdorff distance and clustering algorithms considered in our experiments.

A. Hausdorff distance

The Hausdorff distance is used to measure the dissimilarity of two sets of points in a metric space. It is defined as the maximum distance of the first set to the nearest point of the second set. That is, given two sets of points $A = \{a_1, a_2, ..., a_n\}$ and $B = \{b_1, b_2, ..., b_m\}$, the *directed* Hausdorff distance from A to B is given by

$$h(A, B) = \max_{a \in A} \left\{ \min_{b \in B} \{d(a, b)\} \right\} \quad (1)$$

where $d(a, b)$ is the distance between points a and b under any distance metric of our choice. For our application, we have chosen to use $haversine$ formula to calculate the distances between two datapoints using their reported GPS coordinates (please refer to section III-A1 for more details).

The distance function $h(A, B)$ is not symmetric and therefore, $h(A, B)$ is generally not equal to $h(B, A)$. There are multiple ways of combining the directed Hausdorff distances to obtain an undirected distance measure [11]. In many instances, undirected Hausdorff distance is obtained by taking the maximum of the two directed measures.

$$H(A, B) = H(B, A) = \max \{h(A, B), h(B, A)\} \quad (2)$$

In the rest of this paper, we refer to the distance metric given by equation 2 as the Hausdorff distance for simplicity.

1) Haversine Formula for Computing Distances: Since we will be comparing GPS trajectories, we use the *haversine* formula to calculate the distances between the individual points of sets A and B, denoted by $d(a, b)$ in equation 1. Using the haversine formula, the *approximate* distance between two points, a and b, on the surface of the Earth (assuming Earth is a perfect sphere) can be calculated as follows:

$$d(a, b) = 2R \arcsin(\sqrt{\gamma}) \quad (3)$$

where,

$$\gamma = \sin^2\left(\frac{\phi_b - \phi_a}{2}\right) + \cos(\phi_a)\cos(\phi_b)\sin^2\left(\frac{\lambda_b - \lambda_a}{2}\right)$$

Here ϕ_a and ϕ_b are latitudes, λ_a and λ_b are longitudes of points a and b respectively and $R = 6371$km is the approximate radius of the Earth modeled as a sphere.

B. k-Medoids Clustering

The k-medoids algorithm is a form of partitioning algorithm that works by partitioning a given dataset into groups. It is closely related to the k-means algorithm. The goal of both these algorithms is to minimize the distance between the datapoints of a cluster to the point designated as the cluster center. It is in the definition of center where the k-means and the k-medoids algorithms differ. In contrast to the k-means algorithm, where the center of a cluster is defined as the average of all the datapoints in the cluster, k-medoids algorithm defines the cluster center as the datapoint (from the given dataset) that has the minimum dissimilarity to the other points in that cluster. That is, a medoid of a cluster is the data sample in the cluster that has the smallest average distance to all the other samples in that cluster.

Our selection of k varied depending on the number of trajectories being evaluated. In most cases, we used $k < 10$.

C. Hierarchical Clustering

A hierarchical clustering approach is a method of building hierarchy of clusters. There are two general classes of hierarchical clustering namely agglomerative and divisive. For this study, we focused on agglomerative clustering methods. In agglomerative clustering, each data point starts out as its own cluster. Next, the *closest* pair of clusters are merged based on a distance measure given by the user. This is repeated until all the points are merged into one root cluster. The criterion used to determine the distance between two clusters as a function of the pairwise distances between datapoints in these clusters is called the linkage criterion. In this work, we have experimented with the following three criteria.

- Single linkage clustering: The distance between two clusters is defined as the shortest distance

among the pairwise distances of the members of the two clusters.

- Maximum linkage clustering: The distance between two clusters is defined as the largest distance among the pairwise distances of the members of the two clusters.
- Average linkage clustering: The distance between two clusters is defined as the mean over all the pairwise distances of the members of the two clusters.

These clustering methods also require the user to choose a parameter to determine the number of resulting clusters, which we set equal to k from the k-medoids approach, primarily $k < 10$.

IV. Results

In this section, we illustrate our outlier detection results obtained from the above-mentioned clustering algorithms used with the Hausdorff distance metric.

Figures 1 and 2 show sample results for trajectory clustering obtained by our methods with $k = 2$ for two of the users in the GeoLife dataset. We refer to these users as person A and person B for ease of reference.

Person A has 54 trajectories and is shown in figure 1a. Please note that there is a clear outlier trajectory (in the upper right corner) completely disconnected from the other trajectories. The k-medoid approach forms two clusters of size 35 and 19 (figures 1b-1c) neither of which explicitly identify the outlier. By contrast, the maximum linkage hierarchical clustering forms clusters of size 53 and 1 (figures 1d-1e), with the outlier in its own cluster, as we had intended.

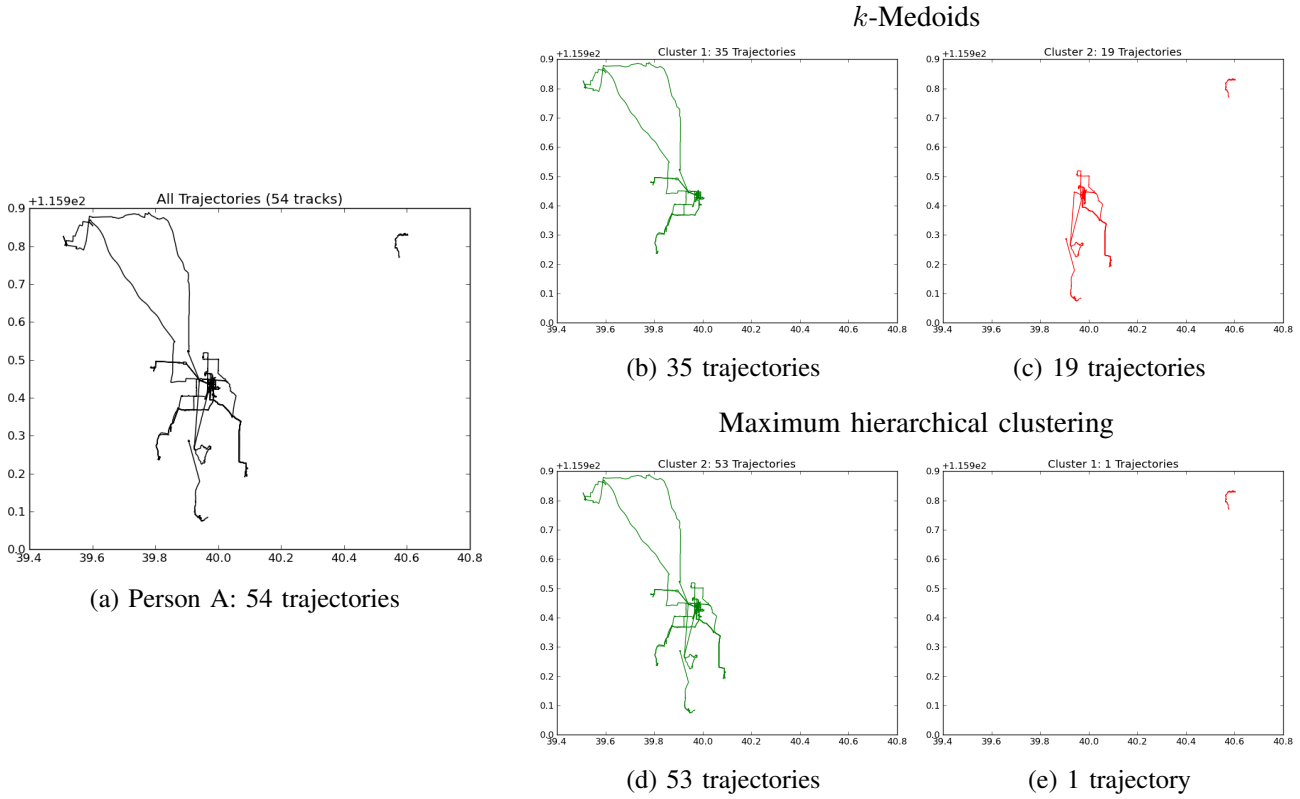
A less trivial example is shown in figure 2 with trajectory data from person B, consisting of 71 trajectories. Again, k-medoids provides a seemingly arbitrary clustering of the data into clusters of size 27 and 44. However, heirarchical clustering forms clusters of size 70 and 1. The single trajectory cluster identified by this method is an outlier when compared to the other samples from this user. Specifically, despite the fact that it originated closer to the dense traffic area directly in the middle, this particular trajectory is different than some of the more peripheral left-right trajectories. On further analysis, it was found that all the other trajectories were urban and along streets, the one we identified as an outlier was an off-road hike, following a trail up into hilly terrain.

In figure 3 we present the clusters identified within person B's data by the different clustering algorithms. This representative result also demostrates the nuanced differences between the different hierarchical methods and the inability of k-medoids to identify homogeneous clusters of trajectories even as k is increased to 5. While the k-medoids results show cluster sizes of 16, 8, 12, 12, and 23 that seem to represent a cluster for each direction and a few dense clusters in the middle, there is no obvious way to select an outlier cluster in an unsupervised manner. For instance, the few trajectories a human may designate as outliers, including the aforementioned hike, lie in the clusters of size 16, 12, and 12 which contain other trajectories that represent more regular movement patterns. By contrast, the hierarchical clustering methods regardless of linkage criterion used, return a cluster of 65+ trajectories that can be labeled as *normal*, and then four other clusters of no more than two trajectories. Additionaly, the trajectories in these small clusters show characteristics that represent some digression from the normal cluster that contain the majority of the trajectories. Therefore, hierarchical clustering methods provide results that are more in line with human perception of outliers and clusters which is the basis of our evaluation criteria.

In all of the cases we tested, the k-medoids clusters seemed to be as heterogeneous as the original data set. Concentration of outliers into specific clusters was infrequent. We experimented with using this approach as an initial partitioning stage, and then ranking trajectories within each cluster by their distance to their respective medoids. This requires setting a second threshold which, given the success of the hierarchical clustering, seems unnecessary. However, this could be used in future while analyzing large amounts of data to identify outlier clusters among multiple users.

A. Limitations and Extensions

Our current approach does not leverage timestamps provided with the data. While this could lead to loss of some information, such as identifying trajectories with an anomalous speed, this simplification reduces complexity and increases generality. The analysis, however, could be expanded upon to take temporal information into consideration. Our approach would work equally well on concatenated trajectories, or partial trajectories. For example, if hour-wise granularity was desired, then the duration of each trajectory could be limited to an hour. Such paritioning has previously been explored by Lee,

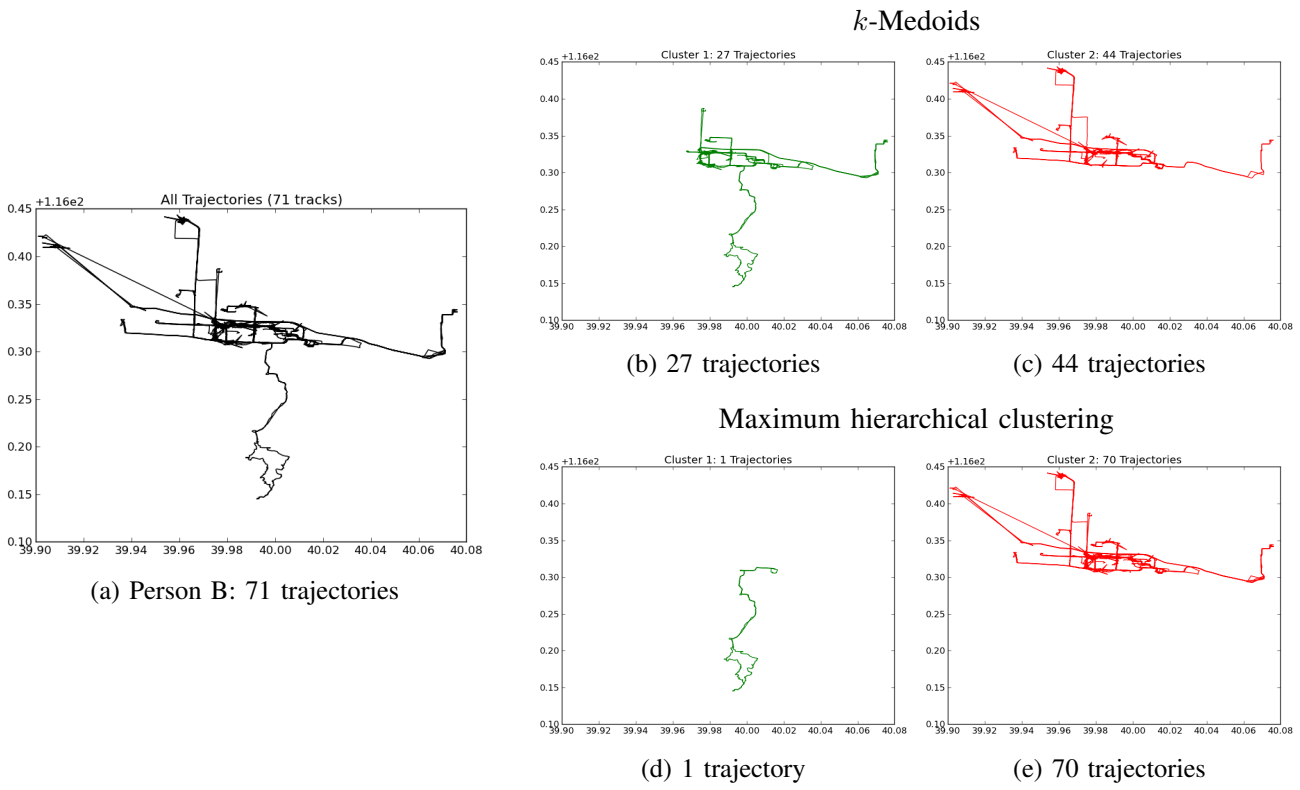


(a) Person A: 54 trajectories

(b) 35 trajectories (c) 19 trajectories

(d) 53 trajectories (e) 1 trajectory

Fig. 1: Hierarchical clustering results picks out outliers automatically in the presence of an obvious outlier trajectory

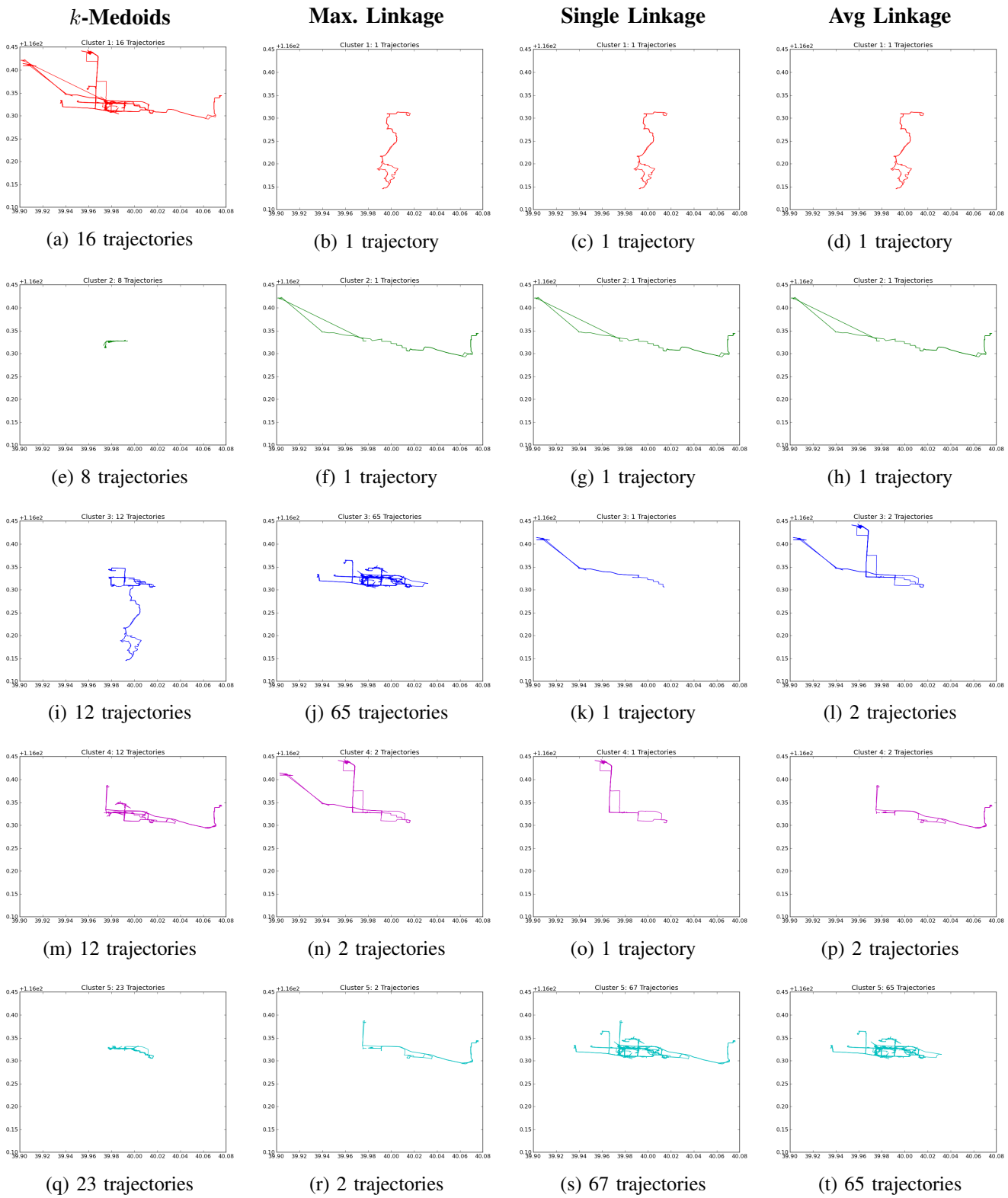


(a) Person B: 71 trajectories

(b) 27 trajectories (c) 44 trajectories

(d) 1 trajectory (e) 70 trajectories

Fig. 2: Hierarchical clustering results are more intuitive than k-medoids clusters



(a) 16 trajectories (b) 1 trajectory (c) 1 trajectory (d) 1 trajectory

(e) 8 trajectories (f) 1 trajectory (g) 1 trajectory (h) 1 trajectory

(i) 12 trajectories (j) 65 trajectories (k) 1 trajectory (l) 2 trajectories

(m) 12 trajectories (n) 2 trajectories (o) 1 trajectory (p) 2 trajectories

(q) 23 trajectories (r) 2 trajectories (s) 67 trajectories (t) 65 trajectories

Fig. 3: Trajectory clustering results on Person B (k = 5)

Han, and Li [12] among others, and our approach does not preclude this.

V. Conclusion

The combination of appropriate distance metric and clustering algorithm helps identify outlier trajectories in our selected dataset. The advantages of our approach are that it is completely unsupervised, generalizes because it does not require domain-specific expert knowledge [3] or labeling [4] as required in other efforts, and should work across data sets of different sizes and complexity. Like all outlier detection work, we are required to set a threshold, which for our approach is the selection of k. In contrast to other approaches that return a fixed amount of outliers [2], our threshold approximately determine the number of outliers returned, but it is not a hard threshold, and more or less outliers may be identified by the algorithm.

Our selection of Hausdorff distance as a metric is motivated by the ambiguity in comparing two variable length trajectories, and intuitive perspective regarding what type of behavior constitutes anomalous. Our metric selection is validated by the strong results we achieved when using it in combination with hierarchical clustering.

Acknowledgment

The authors would like to thank Dr. Dave Rees and the rest of the SSC Pacific Science and Technology Advisory Board for their support.

References

[1] B. Ristic, B. La Scala, M. Morelande, and N. Gordon, "Statistical analysis of motion patterns in ais data: Anomaly detection and motion prediction," in *Information Fusion, 2008 11th International Conference on*. IEEE, 2008, pp. 1–7.

[2] Y. Ge, H. Xiong, Z.-h. Zhou, H. Ozdemir, J. Yu, and K. C. Lee, "Top-eye: Top-k evolving trajectory outlier detection," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. New York, NY, USA: ACM, 2010, pp. 1733–1736. [Online]. Available: http://doi.acm.org/10.1145/1871437.1871716

[3] B. T. Morris and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 8, pp. 1114–1127, 2008.

[4] R. Laxhammar and G. Falkman, "Sequential conformal anomaly detection in trajectories based on hausdorff distance," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*. IEEE, 2011, pp. 1–8.

[5] G. Zhou, B. Lin, and X. Ma, "A spatial clustering algorithm for line objects based on extended hausdorff distance," in *Geoinformatics (GEOINFORMATICS), 2013 21st International Conference on*. IEEE, 2013, pp. 1–4.

[6] J. Chen, R. Wang, L. Liu, and J. Song, "Clustering of trajectories based on hausdorff distance," in *Electronics, Communications and Control (ICECC), 2011 International Conference on*, Sept 2011, pp. 1940–1944.

[7] Y. Zheng, Q. Li, Y. Chen, X. Xie, and W.-Y. Ma, "Understanding mobility based on gps data," in *Proceedings of the 10th International Conference on Ubiquitous Computing*, ser. UbiComp '08. New York, NY, USA: ACM, 2008, pp. 312–321.

[8] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, "Mining interesting locations and travel sequences from gps trajectories," in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW '09. New York, NY, USA: ACM, 2009, pp. 791–800.

[9] Y. Zheng, X. Xie, and W.-Y. Ma, "Geolife: A collaborative social networking service among user, location and trajectory." *IEEE Data Eng. Bull.*, vol. 33, no. 2, pp. 32–39, 2010.

[10] D. M. Hawkins, *Identification of outliers*. Chapman and Hall London, 1980, vol. 11.

[11] M.-P. Dubuisson and A. Jain, "A modified hausdorff distance for object matching," in *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision amp; Image Processing., Proceedings of the 12th IAPR International Conference on*, vol. 1, Oct 1994, pp. 566–568 vol.1.

[12] J.-G. Lee, J. Han, and X. Li, "Trajectory outlier detection: A partition-and-detect framework," in *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. IEEE,

Automatic Solar Cavity Detection Using Haar Cascade Classifier

Aijuan Dong, Athena Johnson, George Dimitoglou, and Joshua Shelley
Department of Computer Science, Hood College, Frederick, MD, USA

Abstract - *Solar physicists have recently been focusing on the examination of solar cavities, circular structures appearing on the sun's upper atmosphere. The high volume of continuously received solar data from observing spacecraft make it difficult to find and identify solar cavities. In this study, we developed a computer model for automated detection of solar cavities using Haar cascade classifiers. Our preliminary experimental results using data from the Solar Dynamics Observatory (SDO) show great promise towards the reliable automated detection of solar cavities.*

1 Introduction

Solar storms, such as solar flares and coronal mass ejections (CMEs), can have serious impacts on critical infrastructure, especially the electric grid and satellites used for communications, global positioning, intelligence gathering, and weather forecasting [1][2]. As we become more and more reliant on technology, and thus, more susceptible to the impact of solar activities, it is important to study, understand, and better predict massive solar eruptions.

Solar storms vary in size and shapes. Coronal mass ejections (CMEs) are spectacular solar eruptions and the primary drivers of "space weather," a term that refers to the variable conditions on the sun and in space. To understand the roots of CMEs, scientists are now zeroing in on mysterious cavities in the Sun's outer atmosphere, or corona [3,4,5,6]. It is believed now that coronal cavities serve as launch pads for billion-ton clouds of solar plasma from CMEs.

Solar cavities have been studied in a variety of ways. In [3], three-dimensional (3D) tomographic model was constructed via extreme ultraviolet (EUV) images and used to study temperatures and densities of solar corona. The study found that cavities have lower densities and broader local differential emission measures. To quantify density and temperature, Gibson et al. established a three-dimensional morphology model of a cavity and used it in studying cavity density and visibility variation [5]. In another study, Reeve et al. used data from the X-ray telescope to examine the thermal emission properties of a cavity and found evidence for elevated temperatures in the cavity center [4]. Durak and Nasraoui [6] presented a methodology for the automated detection of coronal loop regions from solar images using .principal contour extraction and contour classification. The system reported 85% detection accuracy via 10-fold cross validation. While of significant impact, these studies, conducted mainly by solar physicists, focus on manually investigating the physical properties of coronal cavities, such as shape, density, and temperature. Due to the large volume of solar images from space- and ground-based observatories, such as SOHO, TRACE STERO, and SDO, this manual approach is impractical and subjective

In this study, we propose a computer model for the automated detection of solar cavities. The major contributions of the work are: 1) the training of Haar cascade classifiers for automatic solar cavity detection, and 2) the reduction of false positive detections by region grouping and averaging.

The rest of the paper is organized as follows. In Section 2, related background is presented. The details of the proposed method are discussed in Section 3. Experiment and result discussion are described in Section 4 and the paper concludes in Sections 5.

2 Background

The Haar cascade classifier was first used in real-time face detection [7]. There are four key concepts involved: simple rectangular Haar-like features, integral image for rapid feature calculation, boosted machine-learning methods, and a cascaded classifier to combine many features efficiently [8].

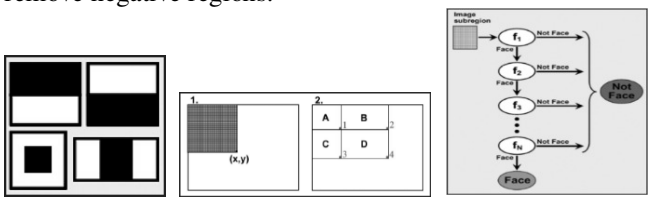
Haar-like features (Figure 1a) share an intuitive similarity with Haar wavelets. The rectangle combinations are picked to match object recognition tasks. The presence of a haar-like feature is determined by subtracting the average of dark-region pixel value from the average of light-region pixel value. If the difference is above a threshold, then the feature is present.

The presence or absence of Haar-like features can be computed rapidly using an intermediate representation of an image, i.e. the integral image (Figure 1b). In an integral image, the pixel (x, y) contains the sum of all pixel values above x and to the left of y. With integral images, a basic feature can be calculated in approximately constant time.

Boosting is a machine learning meta-algorithm based on the idea that several weak classifiers can create a single strong classifier. A weak classifier is a classifier that performs slightly better than random guessing. Weak classifiers, therefore, would not be usable on their own.

However, if we have a lot of them and each pushes the result to the right direction a little, then the combination of them, with each assigning a different weight during learning, will be a strong classifier.

A cascaded classifier (Figure 1c) combines a series of boosted classifiers as a filter chain. The filter at each stage is trained to classify image regions that passed all previous stages. The order of filters is based on their importance were more important filters are placed first to quickly remove negative regions.



a b c

Fig. 1. Haar Cascade Classifier

3 The proposed method

Automated solar cavity detection is a challenging task. First, solar cavities vary in numerous ways, including, but not limited to, size, orientation, density, arc height, and shape (Figure 2). These variations make it very challenging to build a robust computer model that works for every possible configuration. Second, other solar events, with similar structural characteristics such as prominences and flares may also be appearing in the images [6]. Finally, due to the size and continuous growth of solar images, automated detection should be fast, preferably in real time.

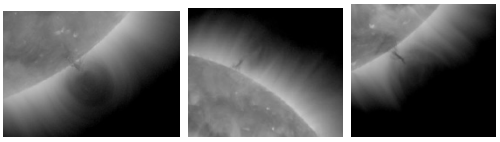


Fig. 2. Sample images of solar cavity

To address these issues, we proposed an automated cavity detection model based on cascade classifier in OpenCV library (http://opencv.org/) (Figure 3). The core of this model is based on Haar-like features. These features can take many forms and orientations, and can easily be scaled by changing the pixel group size under examination, making it possible to detect a variety of cavities. In the following sub-sections, we discuss the model in detail.

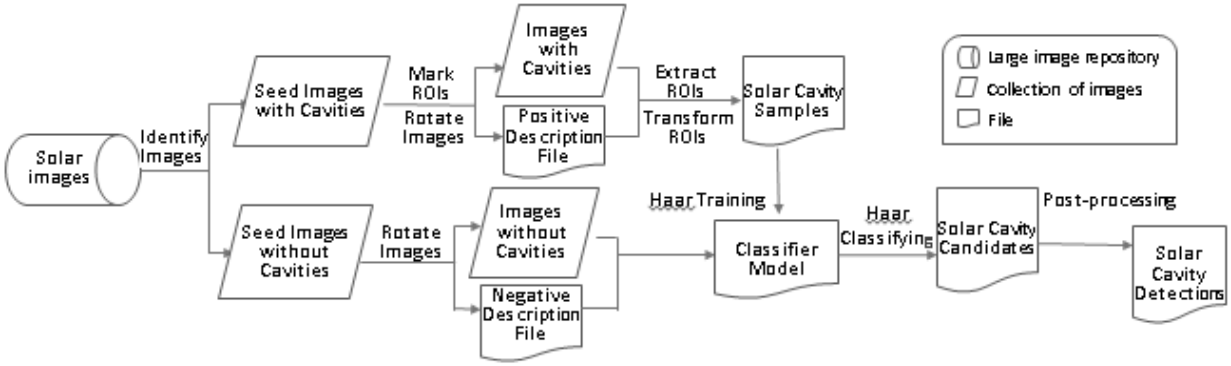


Fig. 3. The proposed solar cavity detection model

3.1 Identify images

The process begins with data preparation, that is, identifying images with cavities and images without solar cavities. Since Haar training requires thousands of images, manually selecting all the images and then marking regions of interests (ROIs) on each one is time-consuming and impractical. In our model, we proposed to pick a small number of representative seed images (i.e. Seed Images with Cavities and Seed Images without Cavities in Figure 3) and then produce a larger number of images required for training via image rotation as described below.

3.2 Mark Regions of Interests (ROIs) and rotate images

For seed images with cavities, the next step is to manually mark ROIs on those images and then rotate them one degree each time to produce Image with Cavities collection (Figure 3).

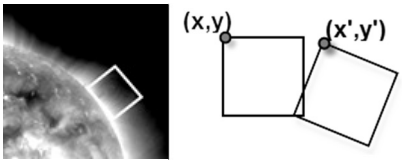


Fig. 4. Image Rotation

Since ROIs (white rectangle on Figure 4) on seed images are manually marked and the coordinates are known, the locations of ROIs on rotated images can be derived using Equation (1). In Equation (1), x and y represent the coordinates on seed images; while x' and y' represent the coordinates of the same point on the image produced by rotating the image θ degrees.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \quad (1)$$

The equation assumes images are based on left-handed Cartesian coordinate system (Figure 5) and image rotation is always clockwise.

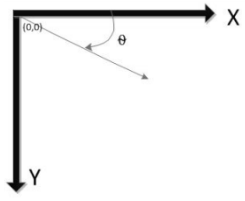


Fig. 5. Left handed Cartesian System

The end result of this process is Positive Description File (Figure 3). For each image, number of ROIs in the image, and the size and location of each ROI on each image are all collected and stored in this file.

Seed images without cavities, also called negative images, were rotated the same way as seed images with cavities to generate Images without Cavities collection (Figure 3).

3.3 Extract and Transform Regions of Interest

In this step, the ROIs on each image, both seed and rotated images, are first extracted, and then transformed to provide an extra level of variance in the training. These extracted and transformed sub-images are stored in a binary file, i.e. Solar Cavity Samples (Figure 3). In essence, sub-images of solar cavities, not the whole images, are packed to speed-up machine learning. This whole process is illustrated in Figure 6, where 6a is a whole image, 6b is the extracted regions, 6c is the transformed regions/sub-images, and 6d the regions/sub-images stored in a binary file.

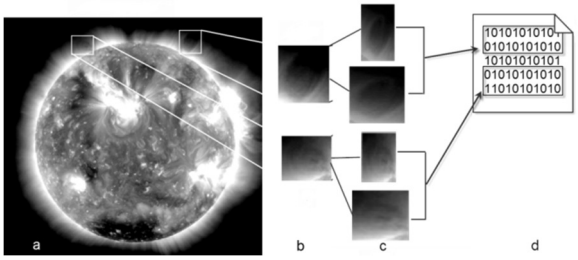


Fig. 6. ROIs extraction and transformation

3.4 Haar Training

Haar training takes three inputs generated from previous steps: Solar Cavity Samples, Images without Cavities and Negative Description File, and generates a Classification Model (Figure 3), consisting of a cascade of classifiers (Figure 7). Haar training involves multiple stages. Before the training starts, the minimum hit rate and the maximum false positive rate for each stage are specified. During training, the Haar-like features are loaded and tested against solar cavity samples. At each stage, the system builds a classifier with desired hit rate first, and then calculates its false alarm rate. If the false alarm is higher than maximum false alarm rate, the system will reject such classifier and will build the next one. Training is complete by either achieving the desired stage count or reaching the required leaf false positive rate at an intermediate stage. The leaf false positive rate is defined as the false positive rate for the cascade as a whole.

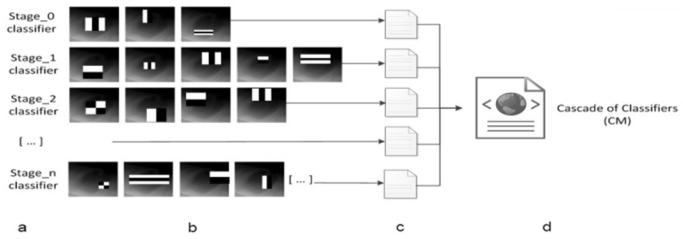


Fig. 7. The cascade classifier

3.5 Haar Classifying

This stage takes a collection of test images, applies the Classifier Model from Haar training above, and outputs solar cavity candidates. To search for the cavities in the whole image, the classifier moves the search window across the image and checks every location. The classifier is designed so that it can be easily "resized" in order to be able to find the objects of interest with different sizes, which is more efficient than resizing the image itself. So, to find an object of an unknown size in the image, the scan procedure is done several times at different scales.

3.6 Post-processing

In our study, we found that the trained classifier models often return multiple hits for one single cavity (Figure 9, 1a, 2a and 3a). There are a fair amount of detections that overlap as clusters; each one is counted as one false detection.

In our study, we reduced the impact of overlapping detection regions by grouping rectangles with similar sizes and similar locations, and return one average rectangle for each large group. In the context of this study, the size of a group is specified by the number of regions a cluster/group has. Since a Haar classifier scans the image several times at different scales and possibly with slightly shifted detection windows, a true cavity generally returns multiple hits at the close proximity of its real location; while a non-cavity solar structure is less likely to be recognized multiple times at different scales. By rejecting small clusters containing less than or equal to the specified number of neighboring rectangular regions, we can reduce the number of false positive detections.

The grouping algorithm partitions a set of rectangular regions into one or more equivalence classes as described in this book [9]. To determine if two rectangles, r1 and r2, are similar or belong to the same group/cluster, first, an adjusted threshold value, delta, is calculated based on Equation (2) where *eps* is the group threshold that is generally set at 0.2.

$$delta = \frac{eps * (\min(r1.width, r2.width) + \min(r1.height,\ r2.height))}{2} \quad (2)$$

Then, location differences (Δx and Δy) and size differences (Δw and Δh) (Figure 8) are calculated and compared with $delta$. If they are all smaller or equal to$delta$, then r1 and r2 are clustered into one group.

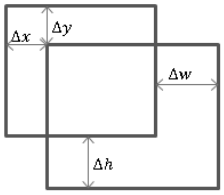


Fig. 8. Location and size differences of two overlapping regions

4 Experiment and result discussion

4.1 The Data Set

The images used for this study are publicly available data from the Solar Dynamic Observatory (SDO) (http://sdo.gsfc.nasa.gov/). The SDO spacecraft includes a suite of instruments: the Atmospheric Imaging Assembly (AIA), the Extreme Ultraviolet Variability Experiment (EVE), and the Helioseismic and Magnetic Imager (HMI). We primarily used images from AIA since the instrument is designed to provide an unprecedented view of the solar corona and delivers far more off-limb structures that the others [10].The AIA image repository contains images in multiple wavelengths and of various sizes, up to 4096x4096 along with other information such as characteristic temperature, primary ions seen, and the exact time when an image is produced.

4.1.1 Training set

For Haar training performed in the experiment, 10 positive seed images (Table 1) and 20 negative seed images were selected from SDO archive. Since solar cavities vary in size, orientation, density, arc height, shape, etc.; the selection tried to be as diverse and representative as possible. After image rotation, the complete image set contains 3600 positive images and 7200 negative images.

Table 1. Summary of positive seed images

Observation date	Total number of images	Resolution	Total number of solar cavities
January. 2, 2011	1	1024	1
February 14, 2011	1	1024	2
March 1, 2011	1	1024	2
March 17, 2011	1	1024	2
April 2, 2011	1	1024	2
April 5, 2011	3	1024	3
April 12, 2011	1	1024	1
June 27, 2011	1	1024	1
Total	10	N/A	14

The negative seed images have the same resolution as positive ones. They were picked from images observed on the following four dates: September 22, 2010, 9 images; March 9, 2011, 1 image; March 10, 2011, 6 images; and April 15, 2011, 4 images.

4.1.2 Test set

Two testing sets were used to evaluate the performance of trained Haar Classifiers. The first set has 42 images, directly pulled out from SDO archive and with observation dates ranging from May, 2010 to Feb. 2013. All solar cavities on these images were manually identified and marked. The total number of solar cavities for this set is 48. The second set has 360 images, generated via rotation from one single SDO image. The cavity locations on all but the original were calculated as described in Section 3.2. There are 360 cavities in the second set.

4.2 Performance evaluation criteria

To evaluate the performance of the proposed method, hits, misses and false detections were used. A hit means a detected region matches the manually marked ROIs with position difference and size difference below specified thresholds. Position and size differences are calculated using Equation (3) and (4), respectively. A miss means a classifier fails to detect a true cavity. A false detection indicates a detected cavity that does not actually exist.

$$D_x < P_x \ where \begin{cases} P_x = w_{manual} * p_x \\ D_x = \sqrt{(x_{detected} - x_{manual})^2 + (y_{detected} - y_{manual})^2} \end{cases} \quad (3)$$

where D_x is the calculated distance, P_x is the adjusted threshold, w_{manual} is the width of manual marked ROI, and p_x is the specified position threshold..

$$S'_x < w_{detected} < S_x \ where \begin{cases} S'_x = \frac{w_{manual}}{s_x} \\ S_x = w_{manual} * s_x \end{cases} \quad (4)$$

where $w_{detected}$ is the width of the detected ROI, S_x and S'_x are the size difference values calculated from the manually marked ROIs width and the specified size difference threshold s_x.

4.3 Study One

The goal of this study is to find out how training set sizes affect classifier performance. The following parameters were used in training: minimum desired hit rate for each stage classifier is 0.995; maximum false alarm rate for each stage classifier is 0.5; Haar-like features used is basic (upright features only); sample size is 24x24 (in pixels); number of stages targeted is 20; and boosted algorithm used is Gentle Adaboost (GAB). Although the targeted number of stages to be trained is 20, the training may finish in an intermediate stage when it exceeded your desired minimum hit rate or false alarm rate.

After training was done, 360 images from the second set were used to evaluate the six classifier models. In this study, the specified position threshold p_x, i.e. maximum position difference, is set as 0.5; the specified size difference threshold s_x, i.e. maximum size difference, is set as 2; the minimum number of neighbors required for each candidate detection is 1.

Table 2. The result of training set size experiment

Classifier Model #	Positives	Negatives	Stages completed	Training time (minutes)	Testing time (second/360 image)	Hits	Hit rate (%)	False detections
#1	1000	1000	11	27	44.7	335	93.1	2455
#2	2000	1000	14	92	56.0	352	97.7	2687
#3	2000	2000	13	101	53.7	360	100	2949
#4	3000	1500	14	172	60.6	359	99.7	3516
#5	3000	3000	13	212	59.7	360	100	3203
#6	3600	3600	14	365	42.2	360	100	3514

The training was performed on a system with an Intel® Core™ i5-2400, 3.1GHz processor, 8 GB of RAM running a 64-bit Windows 7 operating system. The training time in Table 2 ranged from half an hour (100 positives + 1000 negatives) to over 6 hours (3600 positives + 3600 positives). Training time is computationally intensive, so we expect that training time could be reduced by using a more powerful computer or cluster.

With the same hardware environment as training, the average time needed for classifying one image ranged from 117 to 168 milliseconds. This was calculated by taking "Testing time" values in Table 1 and dividing each one by 360, the total number of testing images. Considering the ever-growing large collection of solar images, this result is very encouraging,

The hit rate in Table 2, expressed as percentage, is defined as the number of detected cavities over the total number of true cavities, which is 360 in this study. All except model #1 have hit rates either close to 100% or 100%. This indicates the trained classifiers models have very high sensitivity.

4.4 Study Two

Although the hit rates in Study One are satisfying, all the models had rather high false detections (Table 2). By studying the detection results, we found that overlapping regions are one of the main factors for high false detections. To address this issue, one strategy is to cluster rectangles into groups and return an average rectangle for each large group as explained in Section 3.6. The size of a group is defined by the number of regions a cluster/group has. The goal of this study is to investigate how group size affects false positive reduction. This study also used the 360 image set and the same testing parameters as the ones in Study One. Experiment results are shown in Figures 9, 10 and 11.

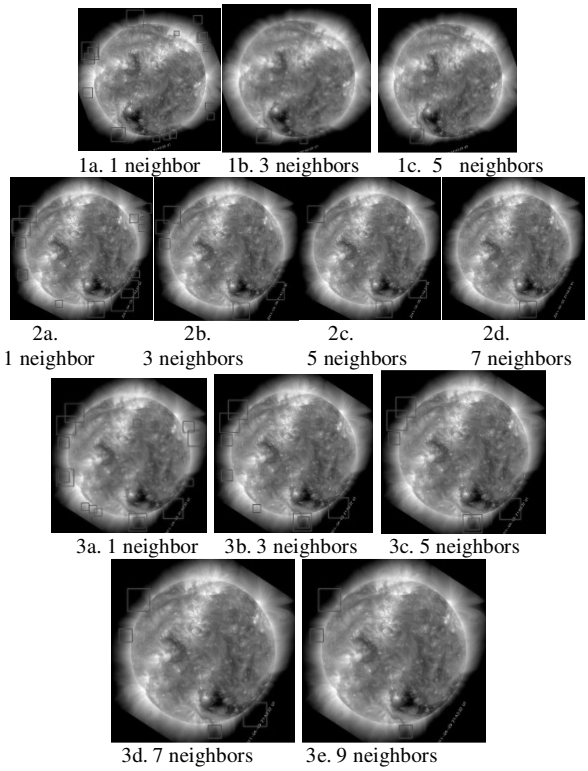


Fig. 9. The impact of neighboring regions on false detections

The three examples in Figure 9 showed that false positive detections caused by overlapping regions can be greatly reduced with increasing minimum number of neighboring regions required for retaining a cavity candidate. For situations where false positive detection cannot be effectively removed (Figure 9, 3e), it is often caused by other solar events that appear similar to solar cavities.

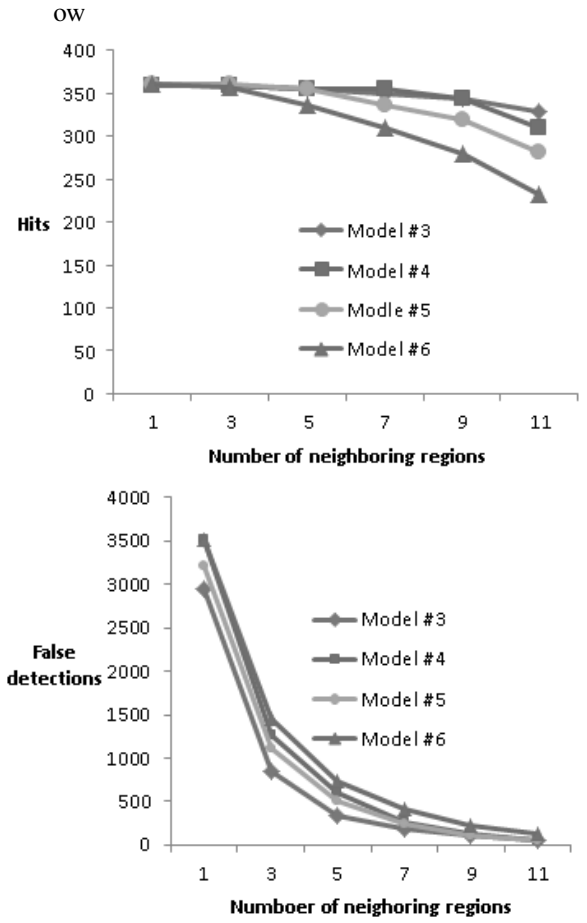


Fig. 10. The impact of neighboring regions on hits and false detections

Figure 10 shows the general trend of this impact. Generally, the more neighbors a model requires, the less false positive detections a model has. However, if the number of required neighboring regions is too high, for example 11, the model will also have more misses. In our study when the number of neighboring regions changed from 1 to 9, the number of hits for both Model #3 and Model #4 decreased very slowly, on the other hand, the false positive detections dropped significantly. Model #5 and Model # 6 exhibit similar behavior and did not perform as well as Model #3 and Model #4. Compared to Model #3 and Model #4, Model #5 and Model #6 have a much larger number of negative samples in the training set. Further study will be carried out on this aspect.

Figure 11, the Receiver operating characteristic (ROC) curve, further illustrated the overall impact of the number of neighboring regions on both hits and false positive detections. For clarity, the number of regions represented by each data point is only shown on Model #3 and Model # 6. Figure 11 shows that controlling the number of regions required for candidate cavity can significantly reduce false positive detections while maintaining satisfying hit rates.

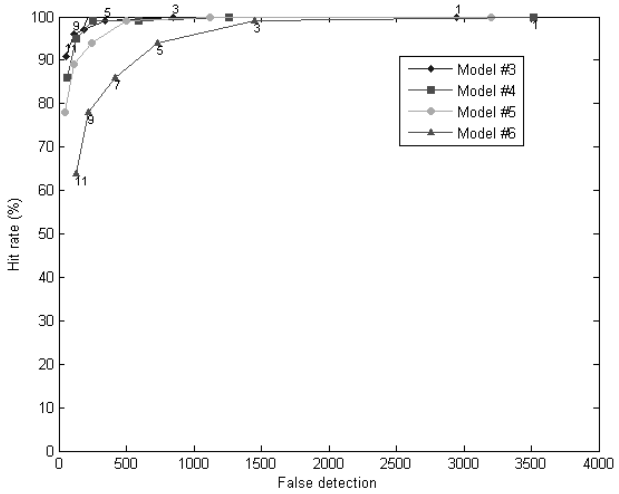


Fig. 11. The ROC curve

4.5 Study Three

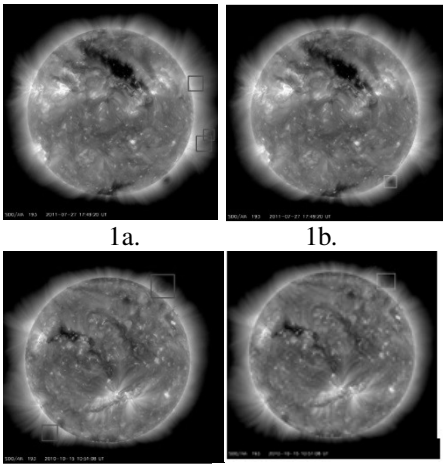
This study used 42 image set. The set has 48 cavities in total since some images have more than one cavity. Compared to previous two studies, the cavities in this set are of different sizes, shapes, density, and ark heights. The images also have high noise level. For all the tests in this study, the specified position threshold is 0.5; the specified size difference threshold is set 2; the minimum number of neighbors that each candidate detection should have is 9.

Table 3 shows that hit rates for three models range from 81% to 98%. The false detections are less than 2 per image. Model # 4 seems performing better the other.

Table 3. The result of Study Three

Classifier Model	Hits	Hit rates(%)	Misses	False
#3	39	81	9	41
#4	47	98	1	62
#5	39	81	7	63
#6	43	90	5	88
Average	42	87.5	5.5	63.5

Figure 12 shows some sample detection results. Red rectangles on 1a and 2a are actual detections; blue rectangles on 1b and 2b represent cavities manually identified. Comparing 1a with 1b, we can see that 1a has one miss and two false detections. The missed cavity is very small in size and has a very dark (strong intensity) center. We suspect it is missed because our training set does not have this kind of cavities. The sample below (2a and 2b) has one hit and one false detection. The false detections on both 1a and 2a are caused either by overlapping regions and/or solar structure that resemble solar cavity. Future improvement should address these two issues.



1a. 1b.

2a 2b

Fig. 12. Sample detection results

5 Conclusion and Future work

In this paper, we proposed and implemented a computer model for automatic solar cavity detection. We trained Haar classifiers and studied the impact of training set size on hit rates. Our studies showed that training time with our commodity platform ranged from 27 minutes (100 positives + 1000 negatives) to 6 hours 5 minutes (3600 positives + 3600 positives); the testing time was less than 0.2 second. Hit rates are close to 100% if the numbers of positive and negative samples are both over 1000. Among all the models we tried, Model #4 (positive: 3000; negative: 1500) seems performs the best.

We investigated false positive reduction via overlapping region removal. Specifically, we studied how the number of neighboring regions required for retaining a cavity candidate impacts false positive reduction. Our studies showed that the more neighbors a model requires, the less false positive detections a model will have. On the other hand, requiring each cavity candidate having high number of neighbor to retain it will cause misses in detection. Our experiment data showed that any number between 5 and 9 is a reasonable choice for minimum number of neighbors required.

Our future work will be focused on further false positive reduction from two aspects: study and selection of a set of intensity and morphological measures to reduce false detections and application of template matching to further improve the overall cavity detection hit rate.

6 REFERENCES

[1]. T. Malik. “The Sun's Wrath: Worst Solar Storms in History.” Space.com Mag., October 23, 2012. Web. Last access on Jan. 31, 2013, available from http://www.space.com/12584-worst-solar-storms-sun-flares-history.html

[2]. K. C. Fox. “Solar minimum; Solar Maximum.” NASA, November 26, 2012. Web. Last accessed on Jan. 31, 2013, available from http://www.nasa.gov/mission_pages/sunearth/news/solarmin-max.html.

[3]. A. M. Vásquez, R. A. Frazin, and F. Kamalabadi. “3D Temperatures and Densities of the Solar Corona via Multi-Spacecraft EUV Tomography: Analysis of Prominence Cavities.” *Solar Phys.* 256: 73–85, 2009.

[4]. K. K. Reeves, S. E. Gibson, T. A. Kucera, H. S. Hudson, and R. Kano. “Thermal Properties of a Solar Coronal Cavity Observed with the X-ray Telescope on Hinode.” *The Astrophysical Journal,* 746:146-156, 2012.

[5]. S. E. Gibson, et al. “Three-dimensional Morphology of a Coronal Prominence Cavity.” *The Astrophysical Journal,* 724:1133–1146, 2010.

[6]. N. Durak and O. Nasraoui. “Principal Contour Extraction and Contour Classification to Detect Coronal Loops from the Solar Images.” *Proceedings of the 20th International Conference on Pattern Recognition (ICPR)*, pp. 2403-2406, 2010.

[7]. P. Viola and M. Jones. “Robust Real-time Face Detection.” *International Journal of Computer Vision* 57(2): 137–154, 2004.

[8]. Robin Hewitt. “How face detection works.” *SERVO Magazin,* 2007. Available online. Last accessed on Feb. 12, 2014 from http://www.cognotics.com/opencv/servo_2007_series/part_2/sidebar.html

[9]. G. R. Bradski and A. Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library.* Sebastopol, CA: O'Reilly, 2008.

[10]. S. Régnier, R. W. Walsh, & C. E. Alexander."A new look at a polar crown cavity as observed by SDO/AIA." Astronomy 1 :1-4, 2011

Processing of Kuala Lumpur Stock Exchange Resident on Hadoop MapReduce

H. Law[1], **S. Aghabozorgi**[1], **S. Lim**[1], **Y. Teh**[1] **and T. Herawan**[1]
[1]Department of Information System, Faculty of Computer Science & Information Technology, University of Malaya, Kuala Lumpur, Malaysia

Abstract - *The Kuala Lumpur Stock Exchange (KLSE) is the big data that need to be stored, processed and analyzed as it trade day-to-day. Analyzing and finding similar components (stock market price) may assist investor. However, it is not easy to find the similar components in the KLSE. This is because the components in the KLSE are changing everyday in the market. This paper focus is on using the Hadoop MapReduce to store and process the KLSE big data, use the k-means algorithm to perform the calculation, then find the companies that had similar KLSE closing bids pattern to help the investors to predict a company's next closing bid based on another company that have the similar trends. To facilitate the investors, the similar trend among companies will be shown on the Graphical User Interface (GUI). All the storing, processing and analyzing will be run automatically behind the scene of the GUI.*

Keywords: Kuala Lumpur Stock Exchange (KLSE), Hadoop, MapReduce, closing bids, pattern, Graphic User Interface (GUI).

1 Introduction

KLSE is formerly known as Kuala Lumpur Stock Exchange and is established in the year 1964. After that, it has been renamed to Bursa Malaysia in the year 2004. Bursa Malaysia is an exchange holding company approved under Section 15 of the Capital Markets and Services Act 2007. It operates a fully integrated exchange, offering the complete range of exchange-related services including trading, clearing, settlement and depository services that are traded on day-to-day. The Prices of the trade are determined by the market forces. The buyers and sellers quote the bid and ask prices and if prices are matched, in the case of KLSE, by its automated trading. Due to the KLSE trade is carried out every day, so there is a dynamic data for the KLSE day-by-day. This big data need to be stored, processed and analyzed so that investors able to see the trend of the stock exchange, and they able to identify when and what stocks to buy and sell, by aware the track of upswings and downswings over the history of one's company according to the sector.

The BigData is the data sets that are large in volume, high velocity, and is complex with variety information assets. The Big Data is in petabytes that consists of billions to trillions of records of millions of people from the different sources such as sales, social media like Facebook, Twitter, patients' record, mobile data, digital pictures and video and more. The Big Data is simply a matter of size[1].

The MapReduce is the programming paradigm for processing large data sets. It consists of two functions, the map function and the reduce function. The map function responsible to partitioning every request into smaller request which are sent to many server, while the reduce job responsible to processing the smaller request using the algorithm provided, and give the best output result to the user.

However, it is not easy to analyze and finding the similar components (stock market price) in the KLSE. This is because the components in the KLSE are changing every day in the market. Hence, the components are highly dynamic to determine the similar trend of the stock prices. The similar trend example is if one's component goes up, the other company's component will also go up as well, and vice versa. Hence, this paper proposed is to use the k-means clustering to determine the similarities among companies, then by using the companies past history of the stock time-series to predict the future closing bids of a company.

The rest of the paper is organized as follows. In section 2, we review the installation of the Ubuntu operating system, in order to use and run the HadoopMapReduce. Section 3 is to review about the k-means algorithm and how this algorithm work in determining the similarities among companies, the Section 4 review about the Graphical User Interface (GUI) in assisting the investor to see a company trend and the k-best similar companies. Finally, in Section 5, we offer and suggest the direction for future work as conclusion.

2 Literature review

The categorization of companies in the stock market is very useful for managers, investors, and policy makers. It can be performed based on several factors, such as the size of the companies, their annual profit, and the industry category. However, these features usually change over the course of time; thus, they are improper for categorization purposes. Industry-based categorization is also not preferable due to

evidence that financial analysts are biased by industry categorization [2]. Identifying homogeneous groups of stocks where the movement in one market affects the stock prices in another market. The literature shows that the similarity of stock market in a country is affected by the movement of other stocks in that country or in other regions [3]–[5]. As a result, numerous studies have been performed on the recognition of co-movements among different countries [6]–[8]. Most of these studies consider the co-movement of the stock market between different reigns or countries but not among different industries or companies in a stock market.

Assessment of the stock market similarity among companies in a stock market (e.g. the Kuala Lumpur stock market) can be very helpful for predicting the stock price, based on the similarity of a company to other companies in the same cluster. Based on the others literature review [9]–[17] on the KLSE stock prices forecasting, it could be notice that most of the existing stock market prediction system is just to forecast the further movement or next stock bids by looking at the company past history, by using the Artificial Neural Network models or the neural network prediction, without referring to any company that have similar trends with it. Therefore, the next bid prediction by the system may forecast inaccurately.

In the time series literature review, [18]–[26] , the author tries to cluster the time series of data efficiently by customer segmentation and developing a novel method for clustering time series incrementally based on its ability to accept new time series and also able to update the underlying clusters. While in the other time series literature review [21], the author stated the significant problem of traditional clustering – defining prototype and explained the benefits of the proposed prototype by customer transaction clustering as well as present a prototype for time series clusters efficiency that can be updated based on a fuzzy concept through iterations.

There are several numbers of literatures that has been published about the BigData and Hadoop as well as the stock market over the Internet. Among these publications, one of the literatures is about Evaluation of Data Processing Using MapReduce Framework in Cloud and Stand-Alone Computing [27]. This article described about the comparison of the data processing speed and time in the cloud computing environment and the stand alone system environment. To establish the experiment, the authors compare and concluded that the Linux environment is more suitable to develop the MapReduce than the windows as the windows had problem connection to a distributed cluster. [27].

3 Environment Setup

Firstly, before storing and processing the KLSE big data, the installation and configuration for the Hadoop MapReduce in the personal computer (stand alone system) are needed. From the above literature review [28], it is determined that the Hadoop MapReduce is more suitable to install on the Linux environment than the windows environment. To provide a test platform on the windows environment, it is recommended to install the Ubuntu operating system version 10.4 Long Term Support in the personal computer in order to run the Hadoop MapReduce. This Ubuntu Long Term Support operating system is a complete desktop Linux-based operating system that allows the Linux application to be compiled and run on a windows operating. The installation of the Ubuntu operating system enables the Hadoop MapReduce to run on the windows laptop over the Ubuntu. After installation of the Ubuntu operating system, the Hadoop MapReduce in the Ubuntu operating system needs to be configured before it can be used by executing the command[29]. Then, the KLSE stock market price components can be loaded into the Hadoop MapReduce, and user needs to key in the Java coding to extract the desired data such as company name, date and closing bids of the KLSE as the output.

4 K-Means Algorithm

The extracted output from the Hadoop MapReduce will be passing to the k-means clustering for further analysis by performing series of calculation on the closing bids, to determine the similarities among companies. Conventional clustering and similarity measures which are applied to static data are not practical for the time-series datasets because they are essentially not designed for time-series data. Hence, various techniques have been recommended for the clustering of time-series data. Most of them try to customize the existing conventional clustering algorithms such that they become compatible with the nature of time-series data. In this cases, usually the distance measure is modified to be well-matched with the time-series data [30].

K-means clustering is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The main idea is to define the k centroids, one for each cluster. In this paper, the following example illustrates the k-means clustering and forecasting using a simulated data set containing a time series components. The reasons of choosing the k-means clustering and algorithm as big data analytics and decision making is because the KLSE big data analysis is within one year time series and its focus only for the closing bids. It is focused on the closing bid because the closing bids are the most real data of the day and this closing bid will be brought to the next day's open bids. The figure 1 below shows the whole process of clustering.

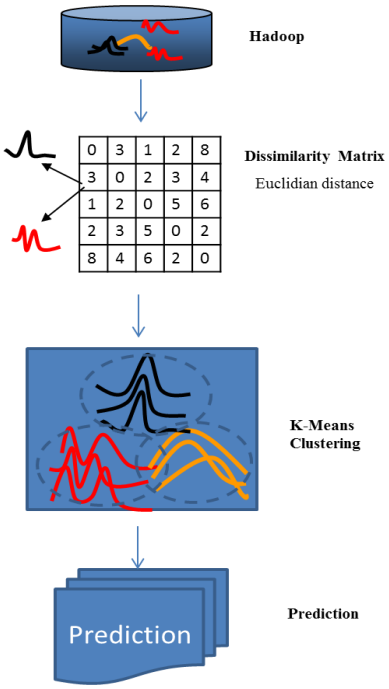


Figure 1. The clustering process

4.1 The Identification of clustering stage

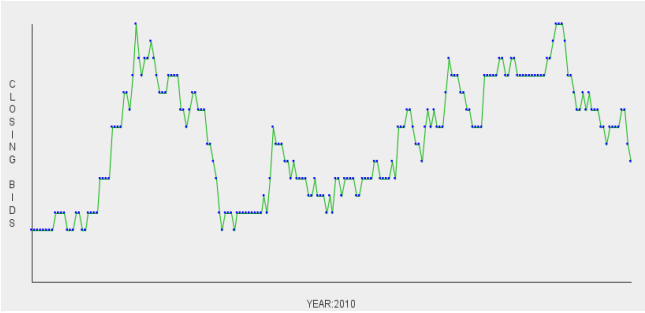


Figure 2. Simulated ARIMA Series KLSE components (From Jan 04, 2010 to Dec 14, 2010).

The starting of the identification stage is to specify the input data set in the k-means clustering. The input data set is the KLSE components. Then use an identify statement to read the KLSE close bids in time series and plot a graph. The graph that has been plotted is shown in the figure 2 above and the table 1 of the data below shows the example of KLSE components data set.

Ticker	Per	Date	Open	High	Low	Close	Vol	O/I
AFG	D	12/14/10	0.23	0.24	0.23	0.23	0	0
AFG	D	12/13/10	0.24	0.24	0.23	0.24	512	0
AFG	D	12/10/10	0.25	0.26	0.25	0.26	3711	0
AFG	D	12/09/10	0.26	0.26	0.26	0.26	88	0
AFG	D	12/08/10	0.25	0.25	0.25	0.25	88	0
AFG	D	12/06/10	0.25	0.25	0.25	0.25	0	0
AFG	D	12/03/10	0.25	0.25	0.25	0.25	100	0
AFG	D	12/02/10	0.25	0.25	0.25	0.25	167	0
AFG	D	12/01/10	0.24	0.25	0.24	0.24	0	0

Table 1. Company AFG's components data set.

4.2 Estimation and diagnosis checking stage

The estimate statement next prints a table of correlations of the parameter wanted, as shown on the table 2 below.

Ticker	Date	Close
AFG	12/14/10	0.23
AFG	12/13/10	0.24
AFG	12/10/10	0.26
AFG	12/09/10	0.26
AFG	12/08/10	0.25
AFG	12/06/10	0.25
AFG	12/03/10	0.25
AFG	12/02/10	0.25
AFG	12/01/10	0.24

Ticker	Date	Close
ASTINO	12/14/10	0.62
ASTINO	12/13/10	0.62
ASTINO	12/10/10	0.62
ASTINO	12/09/10	0.63
ASTINO	12/08/10	0.62
ASTINO	12/06/10	0.62
ASTINO	12/03/10	0.62
ASTINO	12/02/10	0.63
ASTINO	12/01/10	0.62

Table 2. Company AFG's close bids and company ASTINO's close bids are extracted.

When the output is extracted from the Hadoop MapReduce, then use formulas to perform the calculation to calculate the entire closing bids distances between companies.

$$\text{Distance } (t_1, t_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

$$= \sqrt{\begin{matrix}(0.62-0.23)^2+(0.62-0.24)^2\\+(0.62-0.26)^2+(0.63-0.26)^2\\+(0.62-0.25)^2+(0.62-0.25)^2\\+(0.62-0.25)^2+(0.63-0.25)^2\\+(0.62-0.24)^2\end{matrix}}$$

$$= \sqrt{\begin{matrix}0.1521+0.1444+0.1296\\+0.1396+0.1396+0.1396\\+0.1396+0.1444+0.1444\end{matrix}}$$

$$= \sqrt{1.2733}$$

$$= 1.128$$

When the distances between the two companies are known, next is to normalize the distance to the values between 0 and 1 for the standardization purpose.

$$\text{Normalized Distance} = \frac{\text{Distance } (t_1,t_2)}{\sum(x_1+y_1+\ldots+z_1)}$$

$$= \frac{1.128}{\begin{matrix}0.23+0.24+0.26+0.26+0.25\\+0.25+0.25+0.25+0.24\end{matrix}}$$

$$= \frac{1.128}{2.23}$$

= 0.506

When the distance values between companies had been standardized, the similarities between the companies can be determined.

Similarities (t_1, t_2) = 1 – Normalized Distance
= 1 – 0.506
= 0.494
= 0.49

After a series of calculation, it can be seen that the similarities between both company AFG and company ASTINO are 0.49. Therefore, it can be concluded that the smaller the similarities (t_1, t_2) between both companies, the both companies' trends are similar, in contrast, the larger the similarities (t_1, t_2) between both companies, the both companies' trends are not similar. From the similarities (t_1, t_2) between company AFG and company ASTINO, it can be concluded that the both companies are neither similar nor not similar. We expect to see the clusters as shown in the figure 3 below.

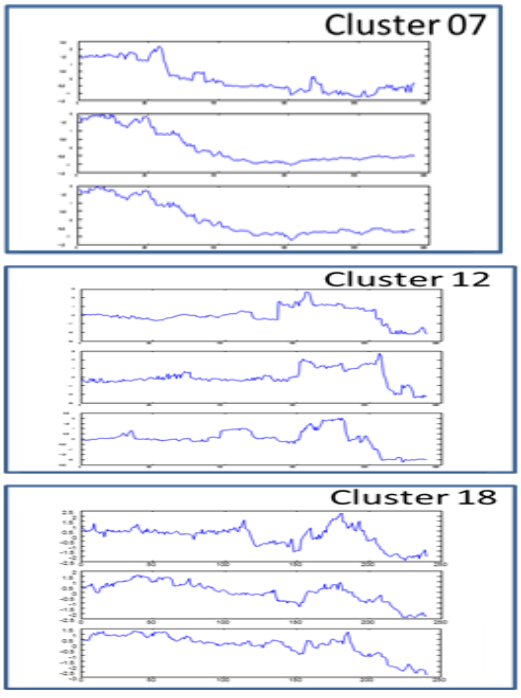


Figure 3. Three sample stock market of three clusters of KLSE datasets

4.3 Forecasting of the KLSE stock prices Stage

After get done in finding the similarities(t_1, t_2) between companies, it is suitable to categorized or rearrange the companies that have the most similar trend to less similar trend based on one's company. For an example: the company that has the most similar trend with Maybank are the CIMB bank (most similar trend), followed by the RHB (similar trend), Public bank (similar trend) and Ambank (less similar trend). To produce forecast, company A's next bid will be predicted based on the other company such as company H that has most similar trend with company A because they have the similarity shape of the stock price or they are co-movement that move together in the same trend. The figure 4 below shoes the daily stock price index prediction of KLSE in the graph form.

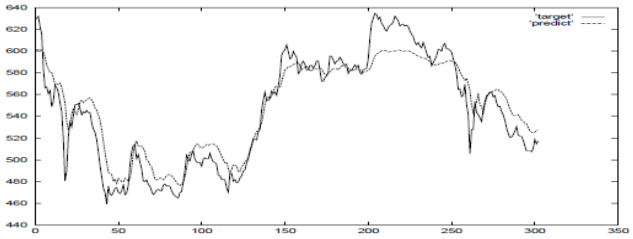


Figure 4. Daily stock Price Index Prediction of KLSE (Out of Sample Data: From July 30, 1990 (horizontal scale 0) to OCT 16, 1991(304))

5 GRAPHIC USER INTERFACE (GUI)

In this paper, the user module will be the Java Graphic User Interface (GUI). The purpose of this module is to provide the user selection on their preference company's stock market prices graph and the similar trend of companies with that particular company, then predict the next closing bids accordingly. The GUI performance is shown in the figure 5, 6 and 7 below.

Figure 5. The use interface that allows user to make selection on their desired company and year.

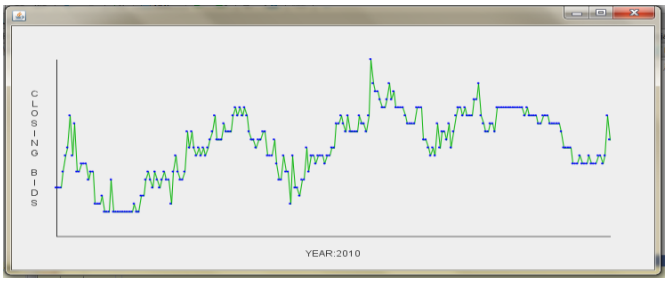


Figure 6. The graph is generated based on the user selection.

Figure 7. The user able to see the k-best similar trend for the selected company.

6 Conclusions

The language for the user module is Java. In this paper, for KLSE components as the input of Hadoop MapReduce, the output is the company's closing bids, then passing to the ARIMA model for the series of calculation, and determined the companies similarities. A simple and clear methodology is used to investigate the similar trends of the KLSE for companies. From the calculation, we found out that the series of the calculation should be integrated into one algorithm to facilitate the calculation, and it should be insert it into the Hadoop MapReduce's reducer part, to minimize the time and get the accurate output in the shortest possible of time. This paper's prediction system are useful to the investors in the future as it able to forecast the company's next bid accurately based on the other companies that have similar trend with it.

7 Acknowledgment

This work is supported by University of Malaya High Impact Research Grant no vote UM.C/625/HIR/MOHE/SC/13/2 from Ministry of Education Malaysia.

8 References

[1] P. Russom, *BIG DATA ANALYTICS (TDWI BEST PRACTICES REPORT)*, FOURTH QUA. TDWI, 2011, p. 40.

[2] P. Krüger, A. Landier, and D. Thesmar, "Categorization Bias in the Stock Market," *Available SSRN 2034204*, 2012.

[3] D. Collins and N. Biekpe, "Contagion and interdependence in African stock markets," *South African J. Econ.*, vol. 71, no. 1, pp. 181–194, 2003.

[4] A. Antoniou, "Modelling international price relationships and interdependencies between the stock index and stock index futures markets of three EU countries: a multivariate," *J. Business, Financ. Account.*, vol. 30, pp. 645–667, 2003.

[5] A. Masih and R. Masih, "Dynamic modeling of stock market interdependencies: an empirical investigation of Australia and the Asian NICs," *Rev. Pacific Basin Financ. Mark. Policies*, vol. 4, no. 2, pp. 1323–9244, 2001.

[6] A. Rua and L. Nunes, "International comovement of stock market returns: A wavelet analysis," *J. Empir. Financ.*, vol. 16, no. 4, pp. 632–639, 2009.

[7] M. Graham and J. Nikkinen, "Co-movement of the Finnish and international stock markets: a wavelet analysis," *Eur. J. Financ.*, vol. 17, no. 5, pp. 409–425, 2011.

[8] L. Norden and M. Weber, "The Coâ€ • movement of Credit Default Swap, Bond and Stock Markets: an Empirical Analysis," *Eur. Financ. Manag.*, vol. 15, no. 3, pp. 529–562, 2009.

[9] J. L. Ford, W. C. Pok, and S. Poshakwale, "The Return Predictability and Market Efficiency of the KLSE CI Stock Index Futures Market," *J. Emerg. Mark. Financ.*, vol. 11, no. 1, pp. 37–60, Mar. 2012.

[10] H. Poh and J. T. Yao, "EQUITY FORECASTING : A CASE STUDY ON THE KLSE INDEX," *Neural Networks Financ. Eng. Proc. 3rd Int. Conf. Neural Networks Cap. Mark.*, pp. 341–353, 1995.

[11] H. Feng and H. Chou, "Evolutionary fuzzy stock prediction system design and its application to the Taiwan stock index," *... J. Innov. Comput. Inf. ...*, vol. 8, no. 9, pp. 6173–6190, 2012.

[12] P. A. Idowu, C. Osakwe, A. A. Kayode, and E. R. Adagunodo, "Prediction of Stock Market in Nigeria Using Artificial Neural Network," *Int. J. Intell. Syst. Appl.*, vol. 4, no. 11, pp. 68–74, Oct. 2012.

[13] B. B. Nair, N. M. Dharini, and V. P. Mohandas, "A Stock Market Trend Prediction System Using a Hybrid Decision Tree-Neuro-Fuzzy System," in *2010 International Conference on Advances in Recent Technologies in Communication and Computing*, 2010, pp. 381–385.

[14] R. P. Schumaker and H. Chen, "A quantitative stock prediction system based on financial news," *Inf.*

Process. Manag., vol. 45, no. 5, pp. 571–583, Sep. 2009.

[15] P.-C. Chang and C.-H. Liu, "A TSK type fuzzy rule based system for stock price prediction," *Expert Syst. Appl.*, vol. 34, no. 1, pp. 135–144, Jan. 2008.

[16] M. B. I. Reaz, S. Z. Islam, M. A. M. Ali, and M. S. Sulaiman, "FPGA realization of backpropagation for stock market prediction," in *Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP '02.*, 2002, vol. 2, pp. 960–964.

[17] P. Gomide and R. L. Milidiu, "Assessing Stock Market Time Series Predictors Quality through a Pairs Trading System," *2010 Elev. Brazilian Symp. Neural Networks*, pp. 133–139, Oct. 2010.

[18] T. W. SR AGHABOZORGI, MR SAYBANI, "Incremental Clustering of Time-Series by Fuzzy Clustering," vol. 688, pp. 671–688, 2012.

[19] S. Aghabozorgi and T. Y. Wah, "Dynamic Modeling by Usage Data for Personalization Systems," *2009 13th Int. Conf. Inf. Vis.*, pp. 450–455, Jul. 2009.

[20] S. Aghabozorgi and T. Y. Wah, "Using Incremental Fuzzy Clustering to Web Usage Mining," in *2009 International Conference of Soft Computing and Pattern Recognition*, 2009, pp. 653–658.

[21] S. Aghabozorgi, T. Y. Wah, A. Amini, and M. R. Saybani, "A new approach to present prototypes in clustering of time series," in *The 7th International Conference of Data Mining*, 2011, vol. 28, no. 4, pp. 214–220.

[22] S. Aghabozorgi and Y. Teh, "Clustering of Large Time-Series Datasets," *J. Intell. Data Anal.*, vol. 18, no. 5, 2014.

[23] S. Aghabozorgi and T. Wah, "Effective Clustering of Time-Series Data Using FCM.," *Int. J. Mach. Learn. Comput.*, vol. 4, no. 2, pp. 170–176, 2014.

[24] S. Aghabozorgi, A. S. Shirkhorshidi, T. Hoda Soltanian, U. Herawan, and T. Y. Wah, "Spatial and Temporal Clustering of Air Pollution in Malaysia: A Review," in *International Conference on Agriculture, Environment and Biological Sciences*, 2014, pp. 213–219.

[25] A. K. Nassirtoussi, S. Aghabozorgi, T. Y. Wah, and D. N. C. Ling, "Text Mining for Market Prediction: A Systematic Review," *Expert Syst. Appl.*, 2014.

[26] Saeed Aghabozorgi and T. Y. Wah, "Shape-based Clustering of Time Series Data," *J. Intell. Data Anal.*, vol. 18, no. 5, 2014.

[27] S. Daneshyar and A. Patel, "Evaluation of Data Processing Using MapReduce Framework in Cloud and Stand - Alone Computing," *Int. J. Distrib. Parallel Syst.*, vol. 3, no. 6, pp. 51–63, Nov. 2012.

[28] S. Daneshyar, "Evaluation of Data Processing Using MapReduce Framework in Cloud and Stand - Alone Computing," *Int. J. Distrib. Parallel Syst.*, vol. 3, no. 6, pp. 51–63, Nov. 2012.

[29] T. White, *Hadoop : The Definitive Guide*, 3rd editio. O'Reilly Media / Yahoo Press, 2012, p. 688.

[30] T. Warrenliao, "Clustering of time series data--a survey," *Pattern Recognit.*, vol. 38, no. 11, pp. 1857–1874, Nov. 2005.

Generating Well-Behaved Learning Curves: An Empirical Study

Gary M. Weiss and Alexander Battistin
Department of Computer & Information Science, Fordham University, Bronx NY, USA

***Abstract*—*Data mining is an important discipline that helps extract useful knowledge from data in business, science, health, and engineering domains. Classification is one of the most common and important data mining tasks. Achieving good classification performance is critical and performance is known to be linked to the amount of available training data. Learning curves, which describe the relationship between training set size and classifier performance, can be used to help determine the optimal amount of training data to use when there are costs associated with procuring labeled data. For learning curves to be helpful, they should be good predictors of future performance, which means that they should be "well-behaved" (i.e., smooth and monotonically non-decreasing). This paper describes how various factors, such as the classification algorithm and experiment methodology (e.g., random sampling vs. cross validation), affect the behavior of learning curves.*

Keywords: classification, learning curves, methodology

1. Introduction

Classification is one of the most important and common data mining tasks. It is well known that classification performance improves with increasing amounts of training data. This can be visually demonstrated via a learning curve, which plots training set size on the x-axis and classifier performance (e.g., accuracy) on the y-axis. The prototypical learning curve is described as follows: performance improves quickly at the start when there is not sufficient data to properly learn the underlying concept well, then the learning curve's slope begins to decrease as an adequate amount training data becomes available, then in the last phase the curve begins to flatten out and the slope approaches 0, as additional training data provides little additional information. However, as was shown in prior research, even in this last phase, small improvements in classifier performance can persist for quite a long time [3].

Data mining work often assumes that there is a fixed amount of training data, available at no cost, and that additional data cannot be procured. This situation undoubtedly fits some real-world situations, but not all. In reality, it is often possible to procure additional training data, but at a cost. This cost could be related to the cost of procuring the data itself, labeling the data (requiring a human and often a human domain expert), or both. In such a situation, a learning curve can be utilized to assess the costs and benefits of obtaining additional training data, and then make the optimal decision. Of course, in practice one cannot form a learning curve without first obtaining training data, so one will always have to predict future learning curve behavior based on the current available training data. In order to make such predictions accurate, it is best if the learning curve is well-behaved—smooth and monotonically non-decreasing.

There has been relatively little related work on utilizing learning curves or generating well-behaved learning curves. Provost, Jensen and Oates [2] evaluated progressive sampling strategies in order to efficiently identify the point where learning curve performance begins to plateau. Weiss and Tian [3] looked at how learning curves can be used to optimize classifier utility when the utility includes classifier performance, CPU time, and data acquisition costs. Both of these research studies would benefit from well-behaved learning curves. Weiss and Tian [3] specifically acknowledged this in their paper when they said, "Because the analyses are all driven by the learning curves, any method for improving the quality of the learning curves (i.e., smoothness, monotonicity) would improve the quality of our results, especially the effectiveness of the progressive sampling strategies." The work in this paper can be viewed as addressing this prior research challenge.

In this paper, we will generate learning curves for six data sets, using four different classification algorithms, and two methodologies for partitioning the data and running the experiments (random sampling and cross-validation). We will visually inspect several of the learning curves to check for monotonicity, but will also look at the variance of the classification performance results. Our hope is that this work will bring attention to the importance of learning curves and will show which factors tend to produce good learning curves and consistent results with low variance. Classification algorithms are currently judged on a number of factors: quality of results, speed of model generation, speed of model application, scalability, and the understandability of the induced model. We would like the consistency (i.e., variance) of the results, which impacts the quality of the learning curves, to be considered as an additional characteristic when evaluating learning methods and learning methodologies.

2. Experiment Methodology

The experiments in this paper are used to assess how various factors impact the quality of generated learning curves. Learning curves are generated by varying the training set

sizes for the data sets listed in Table I. Most of these data sets are fairly large, which enables us to generate learning curves that span a large range of training set sizes—which will help with the evaluation of the quality of the learning curves. Training set sizes are sampled at regular 2% intervals, based on the total amount of data available for training. For 10-fold cross validation, the total amount of data available for training is 90% of the total in Table I, while for random sampling 75% of the total is available, since for all of our experiments random sampling initially allocates 75% of the data for training and 25% for testing.

The Adult, Kr-vs-kp, German, and Arrhythmia datasets are from the UCI Machine Learning Repository [1] while the Coding, Blackjack, \Boa1, Network1, and Move data sets were obtained from researchers at AT&T and can be obtained from the authors.

TABLE I
DESCRIPTION OF DATA SETS

Dataset	# Examples	# Classes	# Attributes
Adult	32,561	2	14
Coding	20,000	2	15
Blackjack	15,000	2	4
Boa1	11,000	2	68
Network1	3,577	2	30
Kr-vs-Kp	3,196	2	36
Move	3,029	2	10
German	1,000	2	20
Arrhythmia	452	2	279

Learning curves are generated using three classification algorithms from the WEKA data mining suite [4]: J48, Random Forest (RF), and Naïve Bayes (NB). J48 is a WEKA implementation of the C4.5 decision tree algorithm. Weka's experimenter mode, as described in an online tutorial [4], was utilized to facilitate the generation of the learning curves. Unless otherwise specified, all results in this paper are based on 10 runs.

3. Results

In this section we evaluate the quality of the learning curves with respect to learning algorithm and experiment methodology (i.e., partitioning strategy). However, well-behaved learning curves are not very useful if classifier performance is not good. Thus, it is important to also know how well the learning methods perform. While the learning curves encode classifier performance, because we use a different graph for each classification algorithm, the relative performance of each learning method may not be apparent from the learning curves. Therefore we display the classifier accuracy for each learning algorithm, at the maximum training set size, in Table II (for 10-fold cross validation). The results show that Random Forest and J48 have the highest average accuracies and significantly outperform Naïve Bayes (Random Forest has a slight overall advantage over J48 even though J48 performs best on 4 of 9 data sets).

TABLE II
ACCURACY WITH LARGEST TRAINING SIZE

Dataset	J48	Random Forest	Naïve Bayes
Adult	86.3	84.3	83.4
Coding	72.2	79.3	71.2
Blackjack	72.3	71.7	67.8
Boa1	54.7	56.0	58.0
Network1	77.3	77.6	74.8
Kr-vs-kp	99.4	98.7	87.8
Move	76.0	80.3	65.2
German	71.1	74.1	75.2
Arrhythmia	65.4	65.2	62.0
Average	75.0	76.4	71.7

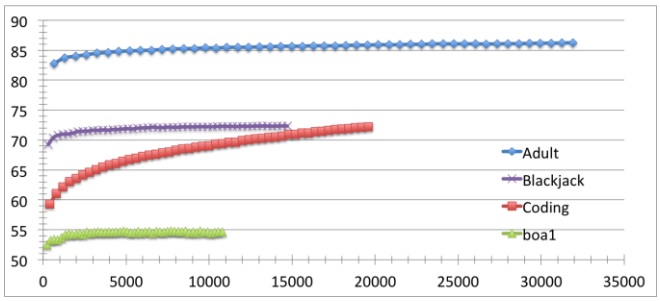
The quality, or monotonicity, of a learning curve can be assessed visually, but a more objective and easily summarized measure is the "variance" of the learning curve. The variance of a learning curve is computed by determining the variance in classifier performance for each evaluated training set size (based on multiple runs) and then averaging these individual variances. The results of the learning curve variances, using 10 runs of 10-fold cross validation, are displayed in Table III.

The results in Table III clearly show that Naïve Bayes generates the lowest variance overall, although for the Boa1 data set it actually has the highest variance (but all values are so low that this may not be too meaningful). Overall J48 and Random Forest seem to perform similarly. Given that the data in Table II showed that J48 and Random Forest produced the most accurate results, J48 would seem to be the best classifier when factoring in accuracy and consistency of results. It should be pointed out that because variance measures the consistency of results for a given training set size, it is theoretically possible to have low variance but have a curve that is not smooth. However, this is very unlikely given that consistent results should lead to the expected behavior—a learning curve that is monotonically non-decreasing.

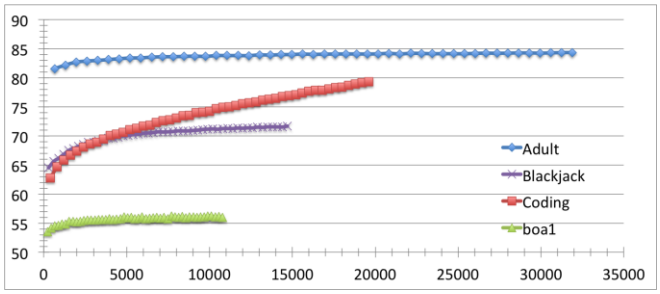
TABLE III
VARIANCES FOR LEARNING CURVES USING 10-FOLD CV

Dataset	J48	Random Forest	Naïve Bayes
Adult	0.51	0.32	0.01
Coding	9.78	17.08	0.19
Blackjack	0.36	2.81	0.01
Boa1	0.20	0.31	0.73
Network1	2.37	2.19	0.10
Kr-vs-kp	3.54	12.08	4.34
Move	28.65	24.73	1.02
German	4.38	1.97	3.48
Arrhythmia	41.46	15.87	9.90

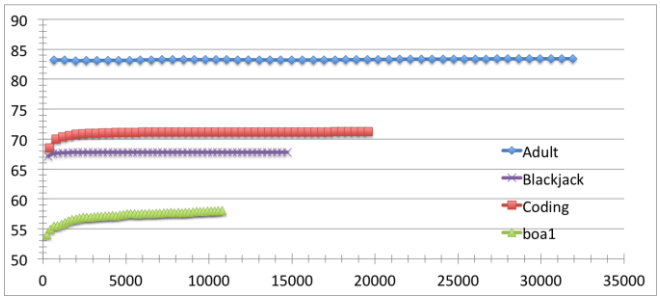
Our first set of learning curves, comprising the four largest data sets, is presented in Figure 1. The curves seem to be well-behaved in that they all appear to be monotonically non-decreasing. Although most of the curves seem quite smooth, the curves for Naïve Bayes appear to be smoother.



(a) J48



(b) Random Forest



(c) Naïve Bayes

Figure 1. Learning curves generated using 10-fold cross validation on the four large data sets. Each chart (a-c) shows the results for a different learning algorithm.

Since J48 and Random Forest are close in variance and accuracies, it is worth taking a more detailed look at each for a specific dataset. In Figure 2 we compare these two algorithms for the adult data set. The results clearly show that J48 generates more accurate results, but also a much better behaved learning curve—with far fewer "blips" where a larger training set size yields a decrease in accuracy.

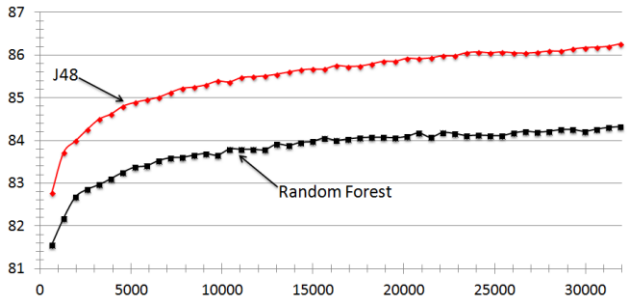
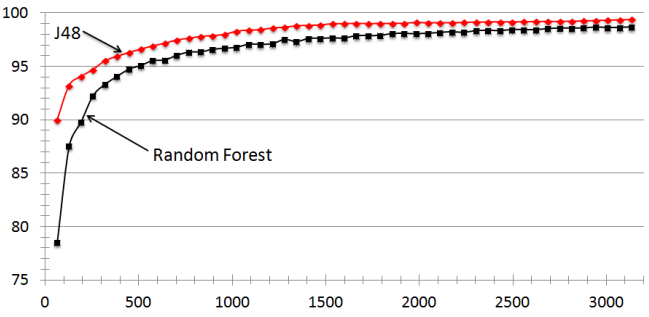
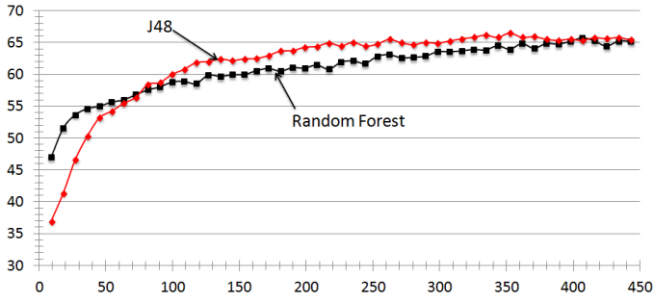


Figure 2. Comparison of J48 and RF cross-validation learning curves for Adult data set.

Next we take a look at the two smaller data sets: Kr-vs-kp and Arrhythmia. We focus on the learning curves for J48 and Random Forest. The results are displayed in Figure 3. We see that for the Kr-vs-kp data set J48 appears to provide a smoother learning curve, which is consistent with the results in Table III that shows that J48 has lower variance. The results for the Arrhythmia data set are not so clear: the results in table III suggest that Random Forest produces better learning curves but based on Figure 3b this is unclear. However, the difference could be explained by the fact that J48 performs extremely poorly for very low training set sizes (worse than guessing the majority class) and the poor performance permits increased variance in results. In the future such small data sets perhaps should be omitted, or the training set sizes should not be permitted to become so small—the smallest evaluated training set size in Figure 3b corresponds to a training set with just nine examples.



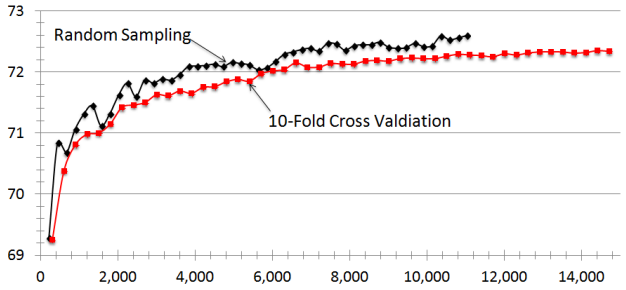
(a) Kr-vs-kp data set



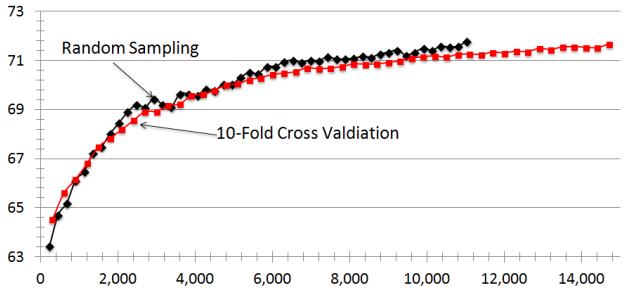
(b) Arrhythmia data set

Figure 3. Comparison of J48 and RF cross validation learning curves.

Thus far we have only examined learning curves generated via 10-fold cross validation. The learning curves generated for Random Sampling are generally not as well behaved as those generated using cross-validation (CV). Due to space concerns we do not show the learning curves for every data set, but instead focus on the Blackjack data set. The learning curves for this data set, shown in Figure 4, indicate that for both J48 and Random Forest, the learning curves generated using cross validation are better behaved.



(a) J48 algorithm



(b) Random Forest algorithm

Figure 4. Cross validation versus random sampling for Blackjack data set

The results in Figure 4 support our general claim that cross validation produces better-behaved curves, although in this case for a given training set size they produce slightly lower accuracies. However, the variance results, displayed in Table IV, are not quite so clear. The variance results show that cross validation is consistently better than random sampling when the models are induced using J48, but that when the models are induced using Random Forest, the two sampling schemes yield similar, although inconsistent, results.

TABLE IV
VARIANCES FOR CROSS VALIDATION AND RANDOM SAMPLING METHODOLOGIES FOR J48 AND RANDOM FOREST ALGORITHMS

Dataset	J48 CV	J48 RS	RF CV	RF RS
Coding	9.78	9.47	17.08	12.97
Adult	0.51	0.53	0.32	15.81
Blackjack	0.36	0.38	2.81	0.33
Boa1	0.20	0.29	0.31	3.80
Arrhythmia	41.46	51.37	15.87	0.19
Kr-vs-kp	3.54	5.36	12.08	15.49
Network1	2.37	2.37	2.19	1.78
Move	28.65	31.74	24.73	27.05
German	4.38	4.51	1.97	3.05

4. Conclusion

In this paper we examined how various factors impact how "well behaved" a learning curve is, based on monotonicity and low variance in classification performance. We focused on the how different classification algorithms and experiment methodologies impact the learning curves and then drew some conclusions based on our empirical results.

Of the learners that we evaluated, Naïve Bayes seems to produce the best-behaved learning curves. However, we do not recommend Naïve Bayes for two reasons: 1) based on Table II its accuracy is not competitive with J48 and Random Forest and 2) examination of the learning curves in Figure 1c indicates that Naïve Bayes' learning curves reach a plateau much earlier than the other methods, suggesting that perhaps the low variance is a consequence of achieving a consistent (but poor) level of performance. While we cannot prove this latter point, it makes sense that once additional data does not improve results, the exact subset of examples used for training may not matter. Given the issues with Naïve Bayes, our recommendation is to use J48 and Random Forest. The comparison of variance results for these two methods is inconsistent: in some cases J48 performs best and in others Random Forest performs best. Therefore based on the results in this paper over a limited number of data sets, we cannot conclude which method generates the best-behaved learning curves. In terms of methodology, the learning curves indicate that cross validation yields better-behaved learning curves than random sampling, as supported by the results in Figure 4. However, the variance results in Table IV are not nearly as conclusive. Thus, this also bears further investigation.

There are various areas for future research that we intend to pursue. First, we intend to analyze more data sets so that we can form stronger conclusions based on a larger sample size. We also plan to analyze a few additional learning algorithms. Better metrics can also help by measuring the "well-behavedness" of learning curves and we have some ideas on how to construct such metrics. Finally, we will vary the number of runs to see how this impacts the learning curves. Once some of these extensions have been implemented, we feel it is likely that stronger conclusions will be possible.

5. References

[1] K. Bache, and M. Lichman, M. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science, 2013.

[2] F. Provost, D. Jensen, and T. Oates, :Efficient progressive sampling," in *Proceedings of the Fifth International Conference on Knowledge Discovery and Data Mining*, 1999, pp. 23–32.

[3] G. M. Weiss and Y. Tian, "Maximizing classifier utility when there are data acquisition and modeling costs," *Data Mining and Knowledge Discovery*, 17(2): 253-282, 2008.

[4] Weka Learning Curves, http://weka.wikispaces.com/Learning+curves.

[5] I. H. Witten, E. Frank, M. A. Hall, "Data Mining: Practical Machine Learning Tools and Techniques,"3rd Edition, Morgan Kaufmann, 2011.

Determination of Fake Reviews in Hospitality Sector

Manish Goswami, S.K. Gupta
Department of Computer Science and Engineering, Indian Institute of Technology Delhi, India

Abstract - *With ever-increasing reliance of customer on user-generated opinions (e.g., Trip Advisor and Yelp) in decision making, there comes an increasing potential threat of fake or fraudulent reviews for monetary gain. This has attracted the academician and industry to substantiate the prevalence of the fake online reviews and weed-out or detect the fake reviews. There has been a lot of work in this field in recent times and researchers have explored varied dimensions to solve the problem. We attempt to integrate most prevalent fake identification techniques to find out a robust classification model to classify reviews as fake or truthful.*

Keywords - Opinion spam, Fake review detection, Behavioural analysis

1. Introduction

"Consumers increasingly rate, review and research products online" [1] [2]. In today's era of widespread internet and online shopping friendly customers, large numbers of product reviews are posted on the Internet. Such reviews are equally important to customers and to businesses. Customers use the reviews for deciding quality of product to buy. Businesses or vendors use opinions to draft their strategies. With ever-increasing popularity of e-shopping and customers sharing their overwhelming experiences online, plethora of information is available for making an informed decision. The service sectors such as restaurants, hotels etc where the service is irreversible, depend on current environmental factors and human involvement has a major role to play in ensuring the quality of service, online opinion from the actual recent users is more relevant. In light of this, hospitality sector customers heavily depend on the online reviews in decision making, assessment of the hotel property and the quality of services offered. Various travel websites allows customers to rate the services used and post their experience online. These websites use the customer feedback to rank hotels. If a hotel is able to secure good ranking, it is assured of better business, which gives ample reason to sway the online image of the business. Due to ease of posting an online review and customer's reliance on them (e.g., TripAdvisor[1], Expedia[2], Makemytrip[3], and Yelp[4]) for decision making, there comes an increased threat of false and fabricated reviews. With minimal input cost, widespread reach and impact on prospective customer, such fake reviews are intended to earn profits for the businesses. As online reviews have become a preferred tool to reach out to people, consequently, to perk up online image of a business posting fake reviews have become more rampant. The faked reviews can be positive to promote own business or negative to discredit competitive businesses. To preserve the customer's trust in online opinions, these reviews must echo genuine user experiences and therefore there is a need to detect and remove such falsely fabricated reviews from the web. Online posting of fabricated or faked reviews is a deliberate review fraud, as it has been reported in a recent case of April 2013 where in Samsung was fined $340,300 by Taiwan's Fair Trade Commission for paying people to post messages online that attacked HTC products while praising Samsung's. In yet another attempt to wrestle against the nuisance of extensive posting of fake reviews, New York Attorney General's office conducted a "sting" operation to publicly shame businesses that buy and write fake reviews [3].

To enable customers to establish the authenticity of an online review, many web sites have collected various manual verification methods for identifying fake reviews [4]. The automated approaches of finding out fake reviews can be mainly categorised into two categories. First, using the Meta data and user associated information with the review such as the reviewer-id, IP address; time of post, rating, helpful index counts etc. Second approach looks at the reviewed text and draws the inferences based on the style of writing, use of specific words and frequencies of the words in the text etc. The second problem of fake review identification can be further categorised as: (a) A standard n-gram–based text categorization problem where text classifiers are used to label opinions as deceptive or truthful (Joachims, 1998 [5]; Sebastiani, 2002 [6]); (b) As an instance of lie detection based on psycholinguistic analysis, where deceptive statements are expected to exemplify the lied emotions to sound

1- http://tripadvisor.com 2- www.expedia.com
3- www.makemytrip.com 4 - http://yelp.com
5-http://abcnews.go.com/Technology/samsung-fined-paying-people-criticize-htcs-

realistic. The psychological effects of lying, tempt increased negative emotion and psychological distancing (Hancock et al., 2008 [7]; Newman et al., 2003 [8]). (c) A problem of genre identification, in which we view deceptive and truthful writing as sub-genres of imaginative and informative writing, respectively (Biber et al., 1999 [9]; Rayson et al., 2001 [10]).

We concentrate on integrating these techniques to determine a classification model that can deliver more authoritative inference to label a review as truthful or deceptive. Truthful reviews can be marked accordingly to help online user to obtain unbiased genuine opinion from the real users. Reviews declared deceptive then can be taken off the web and respective user-id's can be blacklisted. Less probable fake reviews can be marked accordingly so as to be used with caution.

2. Related Work

Spamming in online world started with e-mail (Drucker et al., 2002[11]), and the Web (Gyongyi et al., 2004 [12]; Ntoulas et al., 2006 [13]). The e-mail spam is characterised by the fact that such mails are unsolicited and sent in bulk to numerous recipients with nearly identical content. The objective of these e-mails is to somehow trap recipients into inadvertently installing some malware, or divulge information related to his personal or financial identity. Generally, these sorts of e-mails contains some warnings and cautionary messages in favour of the recipients and ask them to follow some of the links provided to safeguard their interests, or demand to confirm their identity to claim huge prize money. These e-mails carry the links which lead to look-alike phishing web sites, sites that are hosting malware or may include malware as scripts or other executable file as attachments. The objective of Web spam is to make search engines to rank the target pages high in order to attract people to visit these pages. Web spam is generally categorized into 2 main types: content spam and link spam. Link spam is spam of hyperlinks, which is rare in case of reviews as generally there are no hyperlinks among them. Whereas, in case of Content spam irrelevant or remotely relevant words are added in target pages to fool search engines to rank the target pages high.

The ease of posting an online review, minimal input cost, widespread reach, customer's reliance on them for decision making, and impact on prospective customers, have lured the spammers to take advantage of the potential of online reviews. To improve online image of a business, spammers are posting large numbers of fake reviews online. Therefore a need was felt to take steps to weed out or identify the fake reviews to instil the trust of the customers in the online reviews. Jindal and Liu (2008) [14] has established that opinion spam is different in nature from both e-mail and Web spam. With increased usage of e-shopping and the influence of the online opinion the researchers got inclined towards looking into opinion spam (Jindal and Liu, (2008) [14]; Yoo and Gretzel, (2009) [16]; Wu et al., (2010) [15].

The first ever published study by Jindal and Bing Liu [14], focused on identifying three types of spam namely (a) untruthful opinions (b) reviews on brands only (c) non-reviews which have two main sub-types: (1) advertisements and (2) other irrelevant reviews containing no opinions (e.g., questions, answers, and random texts). Duplicate and near duplicate reviews were assumed to be fake reviews for training. An AUC (Area under the ROC Curve) of 0.78 was reported using logistic regression. Jindal and Liu (2008) [17] used the Meta data information associated with the review, and identified features such as review text, reviewer information, and product id's, to distinguish between duplicate opinions (which they considered as deceptive spam) and non-duplicate opinions (considered truthful). They were able to establish that the opinion spam is widespread in the online business. Another study by Mukharjee, Bing Liu [13] proposed a technique to detect spammer groups who work together to write and promote fake reviews. Mukherjee, Liu, Glance [18] proposed a behavioural approach to detect review spammers who try to manipulate review ratings of targeted products or product groups and identifying fake reviewer groups in consumer reviews. Wu et al. (2010) [15] proposed another strategy for detecting deceptive opinion spam based on the distortion of popularity rankings. They attempted to identify items that are targets of spamming by identifying singleton reviews on the reviewed items. Proportion of positive singleton reviews, concentration of positive singleton reviews, and rating distortion caused by singleton reviews are thus used to analyse possibly spammed hotel reviews in Trip Advisor. However, the spammers have gained enough experience to write deliberate fake reviews to sound authentic and to make the job of human evaluators more difficult. Yoo and Gretzel (2009) [16] derived another approach and gathered 42 deceptive and 40 truthful hotel reviews and, using a standard statistical test, manually compare the psychologically relevant linguistic differences between them.

This work was further extended by Ott et al., 2011; [19] wherein they have created a dataset of 800 gold standard opinions. Specifically, they mined all 5-star reviews from the 20 most popular hotels on TripAdvisor in the Chicago area. They filtered out all the opinions of first timers and those which were less than 150 characters. They select 20 truthful reviews from a log-normal (left truncated at 150 characters) distribution fit to the lengths of the deceptive reviews. Deceptive opinions were then gathered for the same 20 hotels using crowd sourcing services of Amazon Mechanical Turk (AMT). In total a dataset of 20 truthful and 20 deceptive opinions for each of the 20 chosen hotels (800 opinions total) were created.
Jeffrey T. Hancock, Myle Ott [19], [12], implemented research findings of psycholinguistic deception detection in social sector by Newman et al. (2003) [8],

which anticipated that a fabricated statements would exemplify the psychological effects of lying, such as increased negative emotion and psychological distancing. They also discussed and implemented n-gram based classification techniques to identify the fake reviews. They attempted to make use of genre identification for fake detection in which deceptive and truthful writing is viewed as sub-genres of imaginative and informative writing, respectively (Biber et al., 1999; Rayson et al., 2001). The paper attempted one half of the problem i.e. the identification of the fake reviews from only positive online reviews. A later paper by Ott et al. [20] attempted the remaining half of the problem i.e. identification of fake reviews from a dataset of negative online reviews using the same approach. They compared the performance of each approach and concluded that machine learning classifiers trained on features for psychological studies of deception and genre identification were both outperformed by n-gram based text categorization techniques.

In parallel, researches were also conducted in the field of social sciences in psycholinguistic deception detection Newman et al. (2003) [8], (2009) [23], Zhou et al. (2004; 2008) [22]. Ott et al., 2011; [19] used these studies to draw a comparative analysis with respect to the n-gram–based deception classifiers along with the performance of human judges. They found out that the n-gram based deception classifiers fares better as compared to others.

Also there have been other attempts to identify the review quality and determine opinion deception (Weimer et al., 2007). Another study was based on the contexts of helpfulness index provided by the readers of the online reviews as well (Danescu-Niculescu-Mizil et al., 2009; Kim et al., 2006; O'Mahony and Smyth, 2009). However, most measures of quality had employed human judges, which are found to be poor in detecting deceptive opinion spam and has no source to identify if the credibility or the helpfulness index was also influenced.

3. Methodology

The problem of finding out fake or deceptive reviews can be mainly categorised into two different streams. First, using the Meta data and user information associated with a review such as the reviewer information, IP address, time of post, rating, helpful index counts etc, and here the text of the review is not taken into account. Second approach looks at the reviewed text and draws the inferences based on the style of writing, use of specific words and frequencies of the words in the text etc. As the later approach depends on the reviewed text, it is more likely to capture the deceptiveness of the review. The review text based approaches for the identification of the fake reviews which have been used in our implementation are discussed below.

3.1 Assumptions

We have come across situation which forced us to make some assumptions to effectively implement our ideas for finding out fake reviews. We make an attempt to list these assumptions. The first and foremost assumption is that the main motive for writing or posting of the fake reviews is the monetary gains by either promoting own / client's businesses or demoting the competitor's businesses. This helps in understanding why a fake review will have increased levels of emotions to sound realistic and have an impact on its readers, and therefore it will be of a different genre as well when compared to the truthful ones.

3.2 Genre identification

Genre identification mainly relies on POS (part-of-speech) tags to identify or classify the input to different classes. In corpus linguistics, part-of-speech tagging (POS tagging or POST), is algorithmic process of marking up a word in a given text as corresponding to a particular part of speech, based on both its definition, as well as its context—i.e. relationship with adjacent and related words in a phrase, sentence, or paragraph. Biber et al., 1999 [9]; Rayson et al., 2001 [10] have established through their work in computational linguistics that the frequency distribution of POS tags in a text is dependent on the genre of the text. For each review we have calculated the features based frequency distribution of each POS tag in the given text. These features in solitary do provide a good baseline with which we attempt to compare our results of integrated approach.

3.3 Psycholinguistic deception detection

The Linguistic Inquiry and Word Count (LIWC) (Pennebaker et al., 2007) is an automated text analysis tool which uses wide range of words categorised in 80 psychological dimensions. This has been widely used in social sciences and legal cases in lie detection. It has also been used to detect personality traits (Mairesse et al., 2007), and, most relevantly, to analyze deception (Hancock et al., 2008 [12]; Mihalcea and Strapparava, 2009 [23]; Vrij et al., 2007).

The LIWC does provide only the psycholinguistic dimensions for a word and does not include a text classifier. We use these LIWC dimensions, in line with the features we have selected to give out the weights and thereby determining the impact of a particular sentence towards the overall score calculation for the given review. The feature are derived after due deliberations and are in line with the standards defined by the International Hotel & Restaurant Association (http://ih-ra.com/).

3.4 Text categorization

In text categorization approach to deception detection we have taken the features derived above in line with IH&RA along with the weights learned from the psycholinguistic analysis. Here we have modelled weighted n-gram features. We consider the three n-gram feature sets, i.e. unigrams, bigrams, and the trigrams. To reap in the total effect we have included a few sentences which define the overall view of a reviewer about the hotel property.

4 Integrated Approach to Deceptive Opinion Spam Detection

We attempted integrating two approaches (genre identification and text categorization) with a view to improve accuracy of detecting deceptive opinion spams. Our rationale behind this approach was that while the n-grams consider the occurrence of specific n-grams and the sequence in which the tokens appear exactly. Using POS along with them as feature would give some more information to the classifier about the frequency and sequence of the POS in a document. Also, if some n-gram occurs in the test data set that wasn't present in the training set, one requires smoothing for the same and it wouldn't be considered as the actual word, treated as some-unknown word. However, POS would have a semantic relationship established with the actual word. This approach is similar to the back-off technique, where instead of backing off to an (n-1) gram, we back-off to its POS tag.

So while the usage of POS alone would overfit the training set, usage of n-gram+ would be useful if the words in the test set are a subset of the words in the training set. POS+NGRAM+ would have the benefit of the specificity of the n-gram approach while being able to retrieve some information from the new n-grams that occur in the test set. We used dataset provided by Ott et al., 2011; [19]. For the work of determination of fake reviews in hospitality sector, we have utilized two publicly available opinion spam datasets provided by Ott et al., 2011; [19]. A total of 1,600 pre-labeled opinions were collected for 20 most popular hotels of Chicago. It comprises of 20 reviews for each hotel in four categories i.e Positive-Truthful, Negative-Truthful, Positive-deceptive, and Negative-deceptive comprising of 400 reviews each. A detailed data collection methodology has been explained by [19].

4.1 Classifiers

In our experiments, combined features from the POS tagging and n-gram based text classification approaches are used to train Support Vector Machine classifiers, which have performed well in related work (Jindal and Liu, 2008 [14]; Mihalcea and Strapparava, 2009 [23]; Zhou et al., 2008 [22]). Each POS is considered as feature along with the traditional unigram features utilized in n-gram text classification. Thus, features used to train SVM classifier consists of features taken from n-gram based text classification approach and POS tagging (Genre Identification) approach.

			Truthful			Deceptive		
Approach	Features	Accuracy	P	F	R	P	F	R
Genre Identification	POS_{SVM}	73.0%	75.3	68.5	71.7	71.1	77.5	74.2
Psycholinguistic Deception Detection	$LIWC_{SVM}$	76.8%	77.2	76.0	76.6	76.4	77.5	76.9
Text Categorization	$UNIGRAMS_{SVM}$	88.4%	89.9	86.5	88.2	87.0	90.3	88.6
	$BIGRAMS_{SVM}$	89.6%	90.1	89.0	89.6	89.1	90.3	89.7
	$TRIGRAMS_{SVM}$	89.0%	89.0	89.0	89.0	89.0	89.0	89.0
Integrated Approach	POS+ Uni-Gram $Features_{SVM}$	89.38%	**91.56**	86.75	89.09	87.41	**92.00**	89.65
	POS + Bi-Gram $Features_{SVM}$	**90.62%**	90.32	91.00	90.66	90.93	90.25	90.59
	POS + Tri-Gram $Features_{SVM}$	90.50%	88.76	92.75	**90.71**	92.41	88.25	90.28

Table 1: Classifier performance for various approaches.

5 Results and Discussion

We implemented the above described techniques in an integrated way and evaluated the performance using a 5-fold cross-validation procedure (Quadrianto et al., 2009) [25], here we select the model parameters for each test fold based on the features derived from the standard experiments on the training folds. Each folder contains the reviews from four different hotels and thus five such folders contain the reviews from 20 hotels. Thereby we have ensured that the learned models are always evaluated on reviews from unseen hotels.

For diagnostic comparisons, we evaluated our model against a trained SVM classifier with POS as features, a trained SVM classifier with LIWC as features, and a trained SVM classifier with n-gram (unigram, bigram or trigram) as features. Results for all appear in Table 1. The results achieved for the integrated approach fairs slightly better than in comparison to existing text categorization approaches. The performance measures (of those in bold) are higher in comparison to existing text categorization approaches. Accuracy of both pos+bigrams+$_{svm}$ and pos+trigrams+$_{svm}$ are better than those (our implementations) of bigrams+$_{svm}$ and trigrams+$_{svm}$ by 0.24% and 0.12% respectively.

Among the automated classifiers, baseline performance is given by the simple Genre identification approach, which can be attributed to the increased awareness of the deception detection techniques which would have cautioned the fake reviewers. Here, we find that a simple psycholinguistic classifier fares slightly better as compared to the Genre identification technique. This can be best explained by theories of reality monitoring (Johnson and Raye, 1981), which suggest that truthful and deceptive opinions can be respectively classified as the fake reviewer try to increase the negativity or positivity in a review to make it more effective and also to sound realistic. Finally the integrated approach provides the best results. It will be interesting to see how an integration of all the above three approaches would result in comparison to the integration what we have been able to achieve.

6 References

[1] J. Jansen. 2010. Online product research. Pew Internet & American Life Project Report.

[2] S.W. Litvin, R.E. Goldsmith, and B. Pan. 2008. Electronic word-of-mouth in hospitality and tourism management. Tourism management, 29(3):458–468.

[3] M. Hu and B. Liu. Mining and summarizing customer reviews. In KDD, 2004.

[4] K. JÄarvelin and J. KekÄalÄainen. Ir evaluation methods for retrieving highly relevant documents. In SIGIR, 2000.

[5] N. Jindal and B. Liu. Review spam detection. In WWW (poster), 2007.

[6] N. Jindal and B. Liu. Opinion spam and analysis. In WSDM, 2008.

[7] N. Jindal, B. Liu, and E.-P. Lim. Finding unusual review patterns using unexpected rules. In CIKM, 2010.

[8] Hancock, J. T., Curry, L.E., Goorha, S., and Woodworth, M. 2008. On lying and being lied to: A linguistic analysis of deception in computer-mediated communication. Discourse Processes, 45(1):1–23.

[9] Mukherjee, A., Venkataraman, V., Liu, B., and Glance, N. 2013. Fake Review Detection: Classification and Analysis of Real and Pseudo Reviews. UIC-CS-03-2013. Technical Report.

[10] Popken, B. 30 Ways You Can Spot Fake Online Reviews. http://consumerist.com/2010/04/ 14/how-you-spot-fake-online-reviews/. The Consumerist. April 2010.

[11]http://www.theguardian.com/world/2013/sep/23/new-york-fake-online-reviews-yoghurt.

[12]http://www.cs.uic.edu/~liub/FBS/opinion-spam-WSDM-08.pdf

[13] Mukherjee,A.,Liu,B.,Glance,N. Spotting fake reviewer groups in consumer reviews. WWW. 2012.

[14] Lim, E. Nguyen, V. A., Jindal, N., Liu, B., and Lauw, H. Detecting Product Review Spammers Using Rating Behavior. CIKM. 2010.

[15] Technical Report UCD-CSI-2010-04, University College Dublin, 2010. Distortion as a validation criterion in the identification of suspicious reviews.

[16] Ott, M., Choi, Y., Cardie, C. Hancock, J. Finding Deceptive Opinion Spam by Any Stretch of the Imagination. ACL. 2011.

[17] M. Ott, C. Cardie, and J.T. Hancock. 2013. Negative Deceptive Opinion Spam. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.