Constructing Super Rule Tree (SRT) for Protein Motif Clusters Using DBSCAN

Bernard Chen, Sait Suer, Muhyeddin Ercan, Rahul Tada, Recep Avci, and Sinan Kockara
University of Central Arkansas, Department of Computer Science, USA

***Abstract*—**

Searching for protein sequence and structural motifs is one of the most important topics in Bioinformatics, because the motifs are able to determine the role of the proteins. A fixed window size is usually defined in advance for the most of motif searching algorithms. The fixed window size may result in generating a number of similar motifs shifted by one to several bases or including mismatches. In this study, to confront the mismatched motifs problem, we use the super-rule concept to construct a Super-Rule-Tree (SRT) which is generated by the DBSCAN clustering algorithm. This SRT recognizes the similar motifs. Analysis of the hierarchical DBSCAN generated Super-Rule-Tree shows a better quality in secondary structure similarity evaluation than the previous studies'. We believe that the combination of DBSCAN and SRT concept may provide a new point of view to similar researches which require predefined fixed window size.

Keywords: **Super-Rule-Tree (SRT), DBSCAN, protein sequence motif.**

1. INTRODUCTION

ALL living organisms require proteins to maintain chemical and physical activities. Proteins are made of 20 types of amino acids [1]. Each protein has its own unique structure and function depending on the sequence and the type of its amino acids. From the point of view of biology and bioinformatics, to reveal the functionality of a protein, it is necessary to obtain the structure of the protein. Hence, an understanding of the formation of amino acids that synthesize the protein is crucial. Analyzing the sequence of amino acids yields some sequence patterns called motifs which have biological significance and repeat frequently. One of the most important Bioinformatics research fields in sequence analysis is searching for motifs, since these recurring patterns have the potential to determine a protein's conformation, function and activities [2].

Proteins are usually grouped based on their structural similarities in order to determine their functional properties. Therefore, to group the proteins, clustering of motif sequences is important. Just like proteins, discovered protein sequence motifs are usually categorized into protein families; PROSITE [3], PRINTS [4], and BLOCKS [5] are three most popular motifs databases that follows this trend. Since sequence motifs from PROSITE, PRINTS, and BLOCKS are developed from multiple alignments, these sequence motifs only search for conserved elements of sequence alignment from the same protein family and carry little information about conserved sequence regions, which transcend protein families [6].

In order to obtain protein sequence motifs which transcend protein family boundaries, we applied our Super GSVM-FE model on all of our information granules so that we obtained 541 extracted high-quality protein sequence motifs in our previous work [7]. However, the most challenging factors of identifying the motifs by clustering them appropriately emerge from the ambiguity and the variability of their sizes. Therefore, a pre-determined size is mostly used in the motif researches. However, two major problems stem from this fixed size namely; mismatches and shifted by one base [8]. The first problem can be simply expressed as the probable similarity of two or more motif groups. The second problem 'shifted by one base' causes to identify one motif more than once as if they are two or more different motifs. For example, if a biological sequence is longer than the fixed size, it is possible to identify the front part and the rear part as two different motifs. In this paper, we try to solve 'grouping similar motifs including mismatches' problem by using super-rules concept [9]. This problem previously was dealt in [2]. In their study, they made an improvement of the HHK Clustering Algorithm [2] and by using the super-rules concept they clustered the motifs and found the similarities among them in the form of a Super-Rule-Tree (SRT).

In this paper; however, we worked out the first problem by using famous clustering algorithm so called Density Based Spatial Clustering of Applications with Noise (DBSCAN) [10] in order to acquire more accurate results. We worked on 541 high-quality protein sequence motifs extracted by Super GSVM-FE model [7]. Then we applied the DBSCAN algorithm on these motifs at different levels of hierarchy to obtain the ideal SRT. DBSCAN algorithm requires two parameters called 'Eps (epsilon)' [10] and 'MinPts (minimum points)' [10]. Eps is the maximum radius of the neighborhood which is to be examined to form a cluster and MinPts is the minimum number of elements required to form a cluster. We applied DBSCAN for all possible values of epsilon and minPts and plotted different graphs taking into consideration minPts, epsilon, number of outliers, number of clusters, and comparatively size of clusters to choose the best pair of parameters. A comprehensive quality comparison of our new Super-Rule-Tree (SRT) with the one in the previous study [2] is also presented.

The remainder of the paper is organized as follows. Section 2 describes the DBSCAN and Super Rule Tree (SRT). Section 3 discusses how we setup the experiment with the DBSCAN and an explanation for determination of parameters. The SRT with comparisons and conclusions are given in section 4 and section 5.

2. METHODOLOGY

2.1 DBSCAN

Density Based Spatial Clustering of Applications (DBSCAN) with Noise is a notable clustering algorithm. It requires two parameters namely Eps and MinPts. Important terms and their definitions are listed below.

a) *Eps:* Maximum radius of the neighborhood to be considered while forming clusters.
b) *MinPts:* Minimum number of points required to form a cluster.
c) *Eps-neighborhood [10]*: A point q is said to be in the Eps-neighborhood of the point p, if the distance between p and q is less than or equal to Eps.
d) *Core points and Border points [10]*: Points inside the cluster are called core points and points on the border of the cluster are called border points.
e) *Directly density-reachable [10]*: A point q is directly density-reachable from a point p w.r.t Eps and MinPts, if q belongs to the Eps-neighborhood of p and the number of points in the Eps-neighborhood of p is greater than or equal to MinPts (see Figure 2.1). If p and q are core points, then directly density-reachable is symmetric i.e., p is directly density-reachable from q and vice versa. However, this condition fails if either p or q is a border point.
f) *Density-reachable [10]*: A point p is density-reachable from a point q w.r.t Eps and MinPts, if there exists a set of points between q and p such that every point in this set is directly density-reachable from its precede.
g) *Density-connected [10]*: If there exists a point x such that the points, p and q are both density-reachable from x, then p is said to be density-connected to q w.r.t Eps and MinPts.
h) *Noise:* Noise is a set of points in a database that does not belong to any cluster. These points are also called as *outliers*.

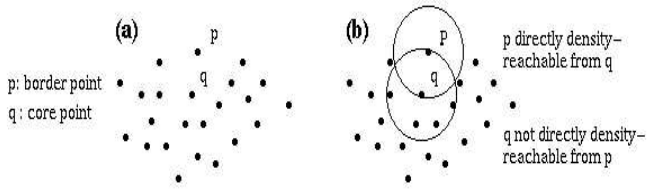

Figure 2.1: DBSCAN application on a 2D data set [10]

This clustering algorithm follows the procedure of finding all points density-reachable from an arbitrary starting point, depending on the Eps and MinPts. If the starting point is a core point then the procedure begins building a cluster. On the other hand, if it is a border point the algorithm cannot go further, i.e., it cannot find any point density-reachable from the starting point. This procedure is followed until all of the points in the Eps-neighborhood are touched or visited at least once. After all of the points in a cluster are visited, the algorithm chooses a new arbitrary starting point to generate other clusters.

For the given example in Figure 2.1, it is not complicated to find the range of parameters and it is not difficult to visualize the data so that the parameters can be determined by starting from 0 to the extreme value, i.e. the distance between the farthest elements. However, in our case, the elements (points) have 180 dimensions or attributes; so, it is difficult to visualize a data in 180 dimensions and challenging to determine the ideal parameters as well as determining a range for parameters. Thus, for Eps, we started from 0 in which every element was found as an outlier. Then we use brute-force approach to reach a point where all the elements form just a single cluster. This approach helped us to find the extreme values for parameters. We further investigated to find the best parameters. Parameters are considered the best possible when the cluster to outlier ratio becomes maximum. This is explained in section 4 with details. 'Manhattan Distance' was used as a distance measure which is the sum of absolute differences between attributes of two elements.

2.2 Super Rule Tree (SRT):

The data set contains 541 motifs, in which each motif has some rules. DBSCAN was used to cluster these motifs based on similarity and then assemble the rules in each motif to generate super rules. Once the rules are generated, it is possible to form another layer of super rules (super-super rules). By this manner, a tree like structure (Super-Rules-Tree structure) is formed using these super rules. These super rules represent a harmonic rule pattern and the essential underlying relationship of classification [9]. Because the super-rules are generated from each of the motifs, it is easy to understand the general trend and ignore the noise and also interactively focus on the important aspects of the domain by using super-rules and selectively view the original detail rules in the corresponding motif [9].

3. EXPERIMENTAL SETUP

3.1 Data set:

The original data set including 2710 protein sequences had been obtained from Protein Sequence Culling Server (PISCES) by Wang and Dunbrack [11]. This data set was used in [2] and [7] to generate protein sequence motifs. No sequence in this database shares more than a 25 per cent sequence identity. We also obtained the secondary structure from DSSP [12] which is a database of secondary structure assignments for all protein entries in PDB. In this database there are 8 different classes of for secondary structures. Chen et al. replaced those 8 classes with 3 classes by assigning H, G and I to H (Helices); B and E to E (Sheets); and all others to C (Coils).

541 different sequence motifs were generated in [7] with a window size of nine from the original data set. Each window is represented by a 9x20 matrix plus additional nine corresponding representative secondary structure information

and it corresponds to a sequence segment. Twenty amino acids are represented by 20 columns and each position of the sliding window is represented by 9 rows. Chen et al. has obtained 541 high quality clusters extracted by super GVSM-SE model and each cluster is represented in 180 dimensions in the first data set. In this study, the 541 clusters obtained from [7] have been used as the data set. In addition to these clusters, the data set which includes the secondary structure of these clusters have also been used.

3.2 Dissimilarity Measure

In this paper Manhattan distance has been used as the dissimilarity measure. Manhattan distance indicates a grid-like path while traveling from one point to another. It is also known as the city block metric. According to Zhong et al. [6], this dissimilarity measure is more suitable for this field of study since all positions of the frequency profile are considered equal.

The Manhattan Distance for the data set is calculated by the following formula:

$$\text{Dissimilarity}=\sum_{i=1}^{L}\sum_{j=1}^{N}\left|F_k(i,j)-F_c(i,j)\right|$$

Where L is the window size and N is 20 representing 20 different amino acids. *Fk(i,j)* is the value of the matrix at row *i* and column *j* and represents the sequence segment. *Fc(i,j)* is the value of the matrix at row *i* and column *j* and represents the centroids of a give sequence cluster. The lower the dissimilarity value, the higher similarity the two segments have.

3.3 Structure Similarity Measure

In order to get the secondary structure and measure the quality of each cluster the following formula has been used.

$$\text{Secondary structural similarity}=\frac{\sum_{i=1}^{ws}\max(\,p_{i,H}\,,p_{i,E}\,,p_{i,C}\,)}{ws}$$

Where ws is the window size , C_i, E_i and H_i correspond to the frequency of Coils, Sheets and Helices respectively and $P_{i,H}$ shows the frequency of occurrence of helix among the segments for the cluster in position i. $P_{i,E}$ and $P_{i,C}$ are defined in a similar way.

If the structural homology for a cluster exceeds 70%, the cluster can be considered structurally identical. If the structural homology for the cluster exceeds 60% and is lower than 70%, the cluster can be considered weakly structurally homologous [6].

3.4 Cluster-Outlier Ratio

A ratio has been used as a criterion to find the ideal parameters Eps and MinPts for DBSCAN. The ratio is calculated by using the following formula:

Cluster_Outlier_ratio = num_cluster / num_outliers

As the ratio increases, the optimum parameters are obtained. However, this ratio is considered in an interval where number of outliers does not equal to zero or the number of elements.

4. EXPERIMENTAL RESULTS

4.1 Determination of Eps and MinPts for Super-Rule-Tree (SRT) construction

Clusters are formed by applying the DBSCAN algorithm on the original data set. But, before that, the most important issue is to determine the values of Eps and MinPts. To determine a logical Eps and MinPts value, the DBSCAN is applied on the original data with Eps ranging from 100 to 500 and MinPts ranging from 2 to 7. These possible parameter pairs were chosen in this range because beyond these boundaries the algorithm accumulates all elements into one cluster or it determines all the elements as outliers. Graphs were plotted for all the values of Eps and MinPts based on the number of clusters formed and the number of outliers. Since the logical Eps and MinPts cannot be determined based on the mentioned criteria, the Cluster-Outlier ratio has been used. This ratio was compared for each Eps and MinPts value within the range and determined its maximum values so that the number of clusters is higher and the number of outliers is less. After graphs were plotted based on different parameters it was determined that the appropriate MinPts value is 2, otherwise the number of clusters declines significantly as shown in the figures below.

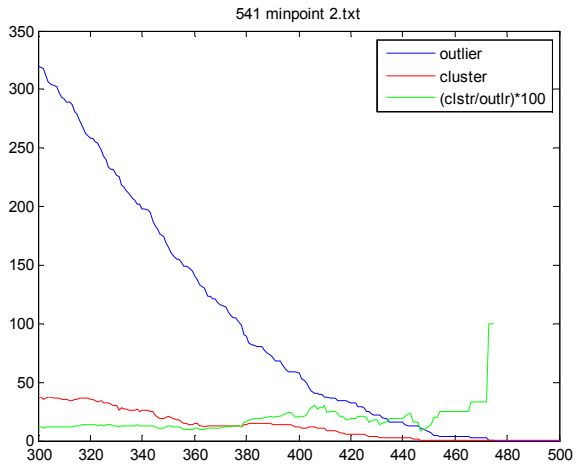

Figure 4.1: Graph for 541 clusters with MinPts=2

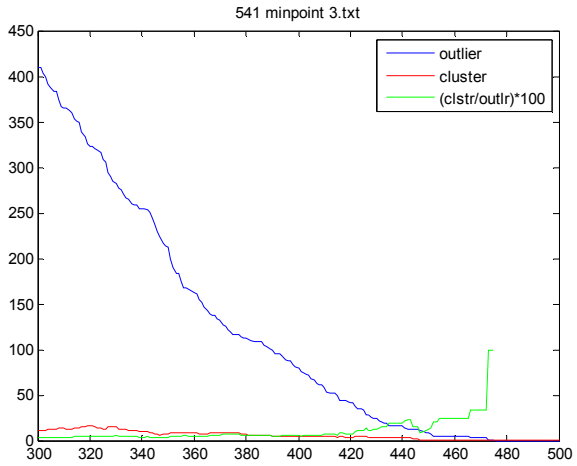

Figure 4.2: Graph for 541 elements with MinPts=3

The x-axis represents Eps. In Figure 4.1, it is revealed that at Eps =406, the clusters to outliers ratio is maximum and at the same time the number of clusters is reasonably high (greater than 1). In Figure 4.2, the ratio of cluster to outlier decreases significantly. A similar trend is observed for MinPts greater than 3, so the parameters are Eps =406 and MinPts =2 for 541 clusters.

4.2 Applying DBSCAN on the sub clusters

As the DBSCAN is applied with Eps=406 and MinPts=2, 12 sub clusters have been found, where the first sub clusters holds 463 elements i.e. 85 percent of the total elements accumulated in one sub cluster. Therefore, we believe it is necessary to cluster these 463 elements and form SRT structure.

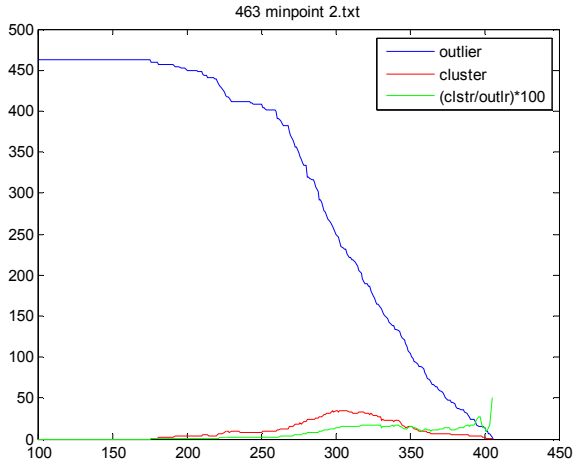

Figure 4.3: Graph for 463 clusters with MinPts=2

In order to apply DBSCAN on this sub cluster we followed the same procedure to determine the Eps and MinPts. From Figure 4.3, the optimum Eps value was empirically found to be 396 and MinPts 2. DBSCAN was applied with these parameters and found 4 sub clusters, where the first sub cluster holds 438 elements, which is majority of the data. Needless to say, we cluster these elements via DBSCAN again.

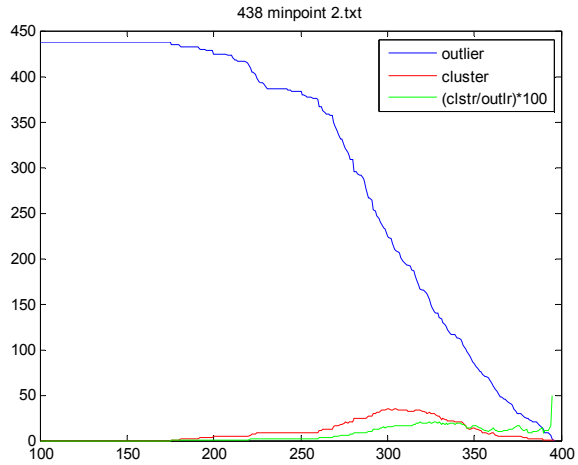

Figure 4.4: graph for 438 elements with MinPts = 2

After the determination procedure was followed for Eps and MinPts and their values are found to be 327 and 2 respectively (as shown in Figure 4.4). The DBSCAN was applied with these parameters and found 29 sub clusters with the first sub cluster holding 126 elements. We stopped further clustering after level 4 (the parameters are determined through figure 4.5 with Eps=319 and MinPts=2) because there is no sub-clusters with more than 100 elements after that. Figure 4.6 shows the multi-layered DBSCAN generated SRT structure, with all the super rules in each motif at each level of DBSCAN application.

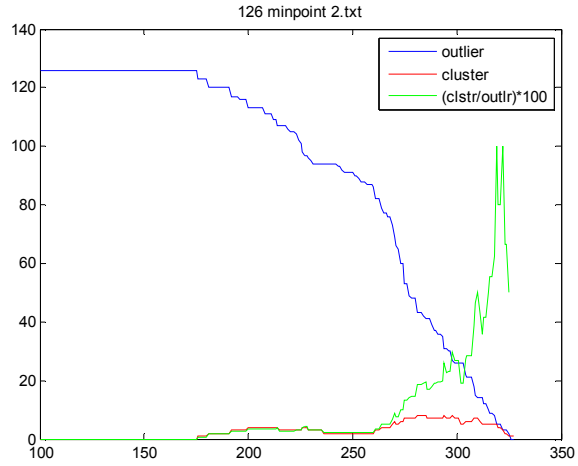

Figure 4.5: Graph for 126 elements with MinPts=2

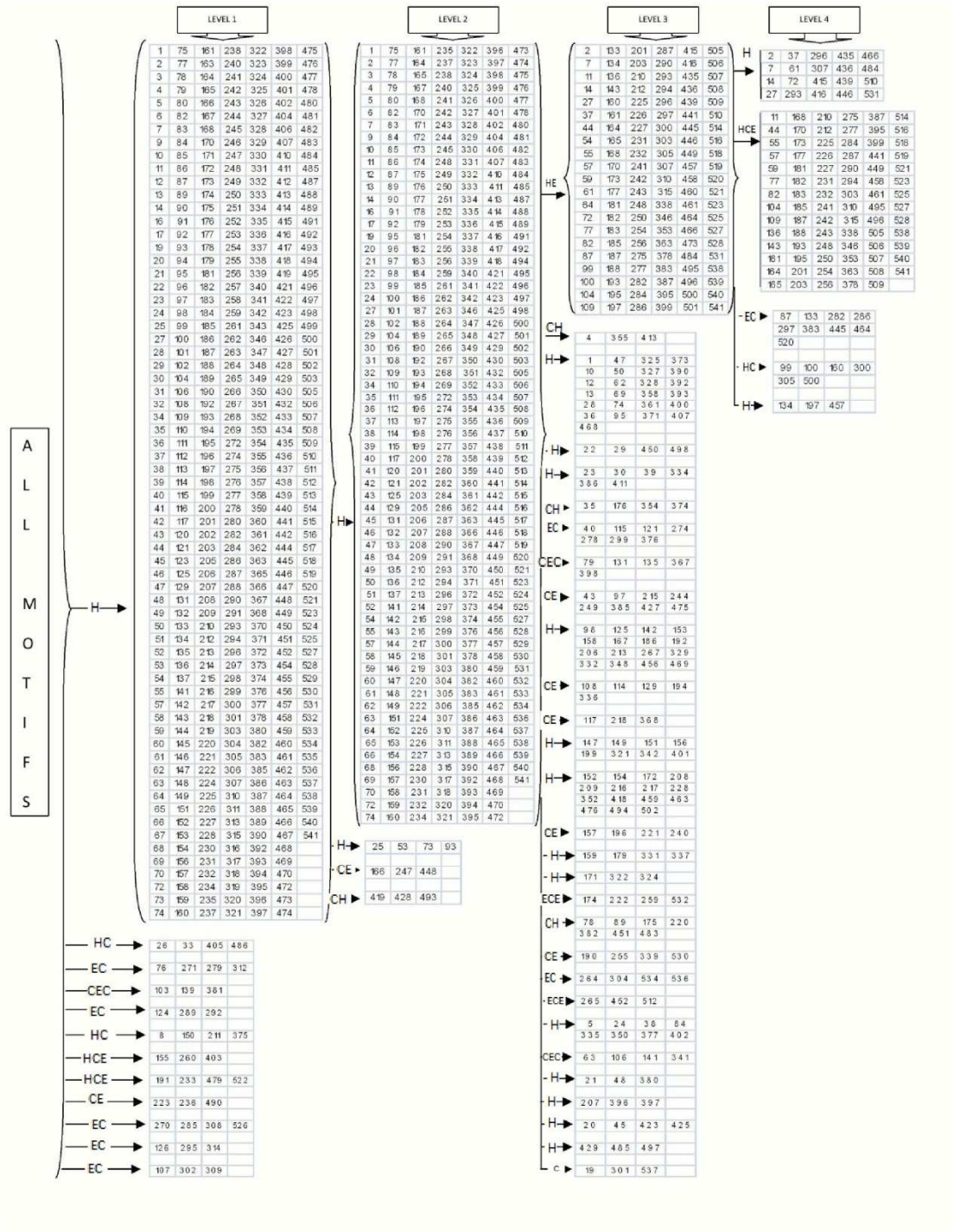

Figure 4.6: A Super Rule Tree with 4 levels of hierarchy

4.2 Super-Rule-Tree comparison

The Super-Rule-Tree generated in this paper is based on the top-down approach; while the Super-Rule-Tree made in [2] is based on the bottom-up method. The major reason causes the difference is according to the number of clusters generated from the clustering algorithms. In [2], the HHK clustering requires no parameters and generates high number of clusters. For example, the HHK clustering algorithm generates 108 clusters when it is applied on 541 protein sequence motifs. Due to the fact that the number of clusters is too large to handle, another level of clustering is applied; thus, a Super-Rule-Tree is formed to have a more generalized view. On the contrary, DBSCAN generates 12 clusters when it is applied on 541 protein sequence motifs with first cluster contain over 85% protein sequence motifs. Clearly, it is necessary to apply DBSCAN on the first cluster. Therefore, a Super-Rule-Tree is formed to have a more specialized view.

"Which SRT is better?" In order to answer this question, we evaluate the SRT level by level using secondary structural similarity. Table 4.1 demonstrates the average cluster quality for each level. Level 1 indicates the first clustering results applied on original 541 protein sequence patterns. Level 2 demonstrates the clustering results on the next level. Since the SRT in [2] contains only 2 levels, we can not compare both Super-Rule-Trees directly. However, it is clear to see that the SRT constructed in this paper is better than the previous works in secondary structure point of view. This mainly because the DBSCAN has the ability to filter out several outliers by setting up Eps and MinPts; while the HHK clustering algorithm can not sieve out outliers because it is a non-parameter approach.

Table 4.1 Secondary structure similarity evaluations on SRT level by level

Average Cluster Quality	LEVEL 1	LEVEL 2	LEVEL 3	LEVEL 4
SRT in this paper	75.98%	76.69%	75.64%	73.50%
SRT in [2]	69.02%	63.48%	NA	NA

5. CONCLUSION

In this paper, we propose that DBSCAN can be utilized to form the Super-Rule-Tree structure. We demonstrate a detailed process and a high quality Super-Rule-Tree, which gives a clear big picture of relations between protein sequence motifs. The improved secondary structure similarity on the SRT provides a better insight of the discovered protein sequence motifs that transcend protein family boundaries. We believe many further researches can be derived from this work.

REFERENCES

[1] Fan, K. and Wang, W. (2003) '**What is the minimum number of letters required to fold a protein?**', *J. Mol. Biol*, 328, 921-926.

[2] Bernard Chen, Jieyue He, Stephen Pellicer and Yi Pan. (2010) *"***Using Hybrid Hierarchical K-means Clustering Algorithm for Protein Sequence Motif Super-Rule-Tree (SRT) Structure Construction***"*, International Journal of Data Mining and Bioinformatics (SCI indexed), Volume 4 - Issue 3, pp. 316-330.

[3] N. Hulo, C. J. A. Sigrist, V. Le Saux, P. S. Langendijk-Genevaux, L. Bordoli, A. Gattiker, E. De Castro, P. Bucher, and A. Bairoch, (2004) "**Recent improvements to the PROSITE database**," *Nucleic Acids Res*., vol. 32, Database issue: D134-137, 2004

[4] T. K. Attwood, M. Blythe, D. R. Flower, A. Gaulton, J. E. Mabey, N. Naudling, L. McGregor, A. Mitchell, G.Moulton, K. Paine, and P. Scordis, (2002) "**PRINTS and PRINTS-S shed light on protein ancestry,**" Nucleic Acid Res. vol. 30, no. 1, pp. 239-241.

[5] S. Henikoff, J. G. Henikoff and S. Pietrokovski, (1999) "**Blocks+: a non redundant database of protein Alignment blocks derived from multiple compilation**," B*ioinformatics*, vol. 15, no. 6, pp. 417-479

[6] Zhong, W., Altun, G., Harrison, R., Tai, P.C. and Pan, Y. (2005) '**Improved K-means clustering algorithm for exploring local protein sequence motifs representing common structural property'**, *NanoBioscience, IEEE Transactions on*, Vol. 4, pp.255–265.

[7] Chen, B., Pellicer, S., Tai, P.C., Harrison, R. and Pan, Y. (2008) '**Efficient super granular SVM feature elimination (Super GSVM-FE) model for protein sequence motif information extraction**', *Int. J. Functional Informatics and Personalized Medicine*, Vol. 1, pp.8–25.

[8] Ohler, U. and Niemann, H. (2001) '**Identification and analysis of eukaryotic promoters: recent computational approaches**', *Trends in Genetics*, Vol. 17, pp.56–60.

[9] He, J., Chen, B., Hu, H.J., Harrison, R., Tai, P.C., Dong, Y. and Pan, Y. (2005) '**Rule clustering and super-rule generation for trans membrane segments prediction**', *IEEE Computational Systems Bioinformatics Conference Workshops (CSBW'05)*, Stanford University, California, USA

[10] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu (1996). *"***A density-based algorithm for discovering clusters in large spatial databases with noise***"*. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)

[11] Wang, G. and Dunbrack, R.L. (2003) *PISCES:* **a Protein Sequence Culling Server**, Bioinformatics,Vol. 19, No. 12, pp.1589–1591.

[12] Kabsch, W. and Sander, C. (1983) '**Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features**', *Biopolymers*, Vol. 22, pp.2577–2637.