Semantic Search Tool for Adverse Event Reports of Medical Devices

Lisham L Singh[1,2], Sithu D Sudarsan[1], Raoul P Jetley[1], Brian Fitzgerald[1], and Mariofanna Milanova[2]

[1]US Food and Drug Administration
[2]University of Arkansas at Little Rock
llsingh@ualr.edu, [{sithu.sudarsan, raoul.jetley, brian.fitzgerald}@fda.hhs.gov], mgmilanova@ualr.edu

Abstract

Signal detection is a critical activity carried out by US Food and Drug Administration (FDA) analysts as part of the agency's mission of public health protection, using large amount of data gathered in disparate formats. Much of this data is unstructured narrative text, limiting use of traditional data mining. Therefore, FDA analysts spend a significant time to locate appropriate documents before relevant information in them can be used. To address this, researchers at the Center for Devices and Radiological Health (CDRH) are developing a semantic search and retrieval framework (SARF) with a Semantic Search Tool (SST) to locate documents containing relevant information quickly.

SARF is capable of analyzing millions of regulatory documents to provide FDA analysts with an intuitive SST for signal detection and evaluation. SARF utilizes a range of techniques innovatively, including use of structured information available in the document for sorting and filtering, while utilizing the narrative unstructured information for context and semantics. Semantic analysis is achieved using, but not limited to, dictionaries, ontologies, and standards.

At FDA, while the value of enabling machine aided narrative text analysis is immense, the benefits of using structured data cannot be over looked. Therefore, SARF is innovatively architected and engineered to take advantage of both structured and unstructured information available with regulatory submissions.

SARF and SST, developed using open-source tools, have been tested with several millions of documents running into multiple terabytes. Yet, the time to query for specific narrative is in terms of milliseconds. A case study is presented to highlight the use of our tool.

1 INTRODUCTION

Decision making is one of the routine but critical activities performed in organizations. Such decisions include technical, scientific, economic and managerial ones. Potential public impact makes decision making more critical in regulatory organizations; hence, scientific and informed decision making is extremely important. For instance, if adverse events reported in respect of a medical device has safety implications then it may have to be recalled based on review by US Food and Drug Administration (FDA). Identifying such specific issue based on available information and evaluation is known as signal detection. Identification, evaluation and confirmation of a signal [1] is an essential ingredient of decision making. This paper outlines a *Semantic Search and Retrieval Framework* (SARF) [2] with a *Semantic Search Tool* (SST) that helps in signal detection.

Incorrect signals result in either *Type 1* or *Type 2* error. When a recall is not made where it should have been made then such an error is known as *Type 1* error; this results in allowing an unsafe device to continue in the market. When a recall is made where it should not have been made then such an error is known as *Type 2* error; this results in denying an acceptable medical device to the needy. Even though specific procedures are followed, sometimes errors do occur. One of the reasons for such errors is the inability to identify and analyze relevant documents and associated meta-data among millions of documents within the available time frame. These documents have accrued over time. SST helps the reviewers performing signal detection in reducing these errors.

Locating relevant information among millions of documents based on queries by the reviewer is a challenging task, due to several reasons. Few of them are: (i) as the size of the historical data grows, it becomes impractical to manually search for similar instances of the problem to make informed decisions within time constraints; (ii) the heterogeneity in the structure and format of data (emails, pdf's, xml, doc, txt so forth as so on) adds to the complexity of searching such data; (iii) many documents contain domain specific descriptions with specific abbreviations, acronyms or terminologies; (iv) diversity of reporting sources, including public, manufacturers, hospitals, laboratories, etc. results in the same or similar events being described in different ways; and (v) useful signals are contained in the narrative description of the event as it provides the context and requires semantic rather than just syntactic search making simple text search across these documents less effective.

At present, reviewers identify relevant document by filtering based on select structured data fields, e.g. date of report, and product code followed by manual analysis of narrative text. Given that over 200,000 devices related adverse events are received by FDA each year, and the number of relevant documents among them are few, the reviewer could be overwhelmed and potentially fatigued. Therefore, any tool that will speed up locating narrative text and reduce the fatigue by eliminating non-relevant documents for analysis enhances productivity. This makes SST a very useful tool for signal detection and evaluation.

In this paper we present SST that would aid reviewers in finding relevant documents by supporting various types of queries ranging from syntactic to semantic search with meta-data based filtering/sorting. Users have option to automate and customize the use of different dictionaries and reference tables while constructing the query to accommodate specific semantic requirements. Our approach enables text-mining over a huge document collection with document size ranging from few kilobytes to several gigabytes; this improves upon current large scale text mining solutions which expect specific document size, e.g., typical document size of few hundred kilobytes or about 5000 to 10000 words per document and so on. SARF supports multiple document corpora.

SST is a practical/scalable tool that facilitates efficient searching of relevant documents, using text mining techniques based on SARF. Our approach is generic and independent of the structure of data being searched. SARF allows automation as well as customization by query reformulation and expansion. It also provides resources to the users so that they can refer or look up as reference documents. The efficiency of the tool remains unaffected with increasing size of document collection, making it scalable.

The tool is designed to search across loosely-coupled corpora i.e. independent corpora, as SARF enables this feature without any significant compromise in response time. Even though we focus on medical device adverse event corpus, there are multiple corpora. This feature is very useful, for example, when an adverse event involves a device as well as, says a drug and reviewers need to locate documents from either of the corpus. Drug related adverse events form its own corpus.

Our implementation uses open source tools including Apache Lucene [3], which is a high-performance, open-source, information retrieval Java API library. This approach has made the solution to work across operating systems, while avoiding the reinventing of the wheel.

SST is web enabled and has a user friendly and interactive graphical presentation of the search result.

The rest of the paper is organized as follows, Section 2 discusses background and some related work. We outline the problem in Section 3. In Section 4, our approach is discussed. Section 5 presents a case study of the tool. Section 6 concludes the paper by summarizing the contributions proposed in this work.

2 BACKGROUND

A traditional search engine based approach could be useful in our case, but for its limitation on being just syntax based. Semantic search engine fits the requirements better as the descriptions are semantically similar. Syntax based searching is straightforward and it works on looking for documents that contain the terms or patterns specified in the query. Semantic search engines, on the other hand, aim to improve search accuracy by taking the contextual meaning of terms as they appear within the search space, to generate relevant results. However, developing such a tool is more challenging.

Though there are difficulties in developing accurate and powerful semantic search engines, the popularity of such applications are increasing and spreading in many areas. Some interesting research work on and related to semantic search are: Moldovan et al. in [4] have discussed about improving the search quality of traditional search engines by using WordNet, and later work by Guha et al. [5] mentioned how relationships of objects on the web documents can be established and exploited for semantic search. More recently, researchers in [6 - 8] have shown different approaches toward semantic search incorporating existing ontologies, taxonomies and natural language processing techniques. Lopez et al. in [6] have addressed a question-answering system which takes queries expressed in natural language and an ontology as input and returns answers drawn from the available semantic markup.

Existing semantic search applications are based on machine readable electronic documents. While electronic, our documents are not readily machine readable but mostly human readable. This makes available semantic web engines unsuitable for the task at hand and necessitates custom solution. We address handling of documents containing descriptions of medical device adverse events. These include the initial report, follow-up reports, and communications from FDA to the manufacturers and user facilities. As our tool deals with large number of documents, we use certain techniques to ensure quick response to queries while keeping it scalable. This section outlines few of them.

IR systems

Information Retrieval (IR) systems aim to identify documents that are relevant to a given query among the documents available for search, typically ranked in some order of relevance. Sometimes, they point out the location of query or its related terms within the document. To this end IR systems [9, 10] address issues concerning representation, search and manipulation of large collection of electronic documents. Here, we are concerned with documents containing narrative text. Some popular and widely used IR systems are web search engines like *Google*, *Bing*, and *Yahoo*. Additionally digital library based services enable researchers, academia and medical practitioners to learn about new research articles published in their respective areas. IR systems are not limited to web search engines and digital libraries, but extend to desktop searches to specially designed enterprise level search systems. A typical IR system can be used to address multiple concerns including *``document routing/filtering'', ``text clustering and categorization'', ``text summarization''* [11], *``information extraction''* [12], *``topic detection and tracking''* [13].

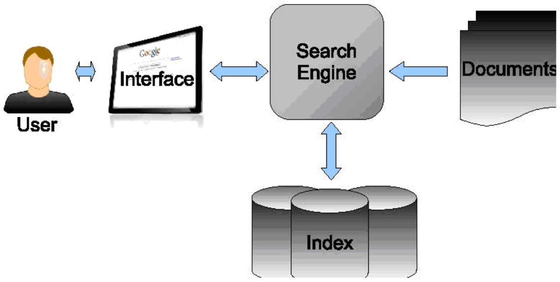

Figure 1: Components of an IR System

Core components of a typical IR system are shown in Figure 1. A search engine maintains an index of the document contents that need to be searched. Users issue queries to the IR system through user interface. Queries are typically made of terms (Note: We use “terms” instead of words due to the way indices are generated. For instance a term may be combination of number, date, wildcard characters etc.). The search engine processes the query and responds with a list of relevant documents that contain terms matching the query. The returned document list as the result of a query is ranked based on a ranking algorithm.

Inverted File Index

Searching the contents of actual documents each time a query is received is resource intensive and time consuming. Therefore IR systems typically maintain an index of documents to speed up the process. Among various indexing strategies, use of inverted file indexing [14] is well suited for large document collections. Queries are run against the inverted file index which tends to be much smaller than the documents themselves and hence results in quick identification of query terms in the index. Each term then points to documents that contain those terms. Additional information such as the term frequency helps in ranking the returned documents in specific order.

In its simplest form inverted file index maintains the mapping of terms and their location in a text collection. For instance a text document can be thought of as a collection of m words. It is made up of a sequence of n unique words such that $n<=m$. The number n is usually far less than m as most of the words is repeated while forming a document. For instance the word ‘the’ is repeated several times in this paper. The set of unique words within an index forms the "Term List" υ of the index. If a pointer (say numeric location) is associated with each word in υ to the location of that word in text document, the resultant data structure is a form of inverted file index. As the document collection grows, the number of documents matching a word in the index becomes sparser.

Text Mining and Semantic Search

The ability to retrieve related information from narrative texts enables performing more complex operations like text mining. The aim of text mining is uncovering hidden information from text documents [11]. Such hidden information could be discovered from contextual or semantic or ontological relationships among documents as reviewed in [15].

In addition to IR, techniques like Information Extraction (IE), Data Mining, Natural Language Processing (NLP) etc are used in text mining [16].

Semantic Search

A semantic search application takes user query and it returns top-k of the most conceptually relevant documents. The main phases of a semantic search could be summarized as (i) *Query Expansion*, which converts the searcher's query to a Semantic query. Some of the works on query expansion include Mitra et al. [17] and [18]. Such techniques help in increasing both recall and precision values. Techniques for fuzzy and proximity based searches increase recall but may reduce overall precision. (ii) *Search Space*: This is created during the indexing of the document collection. Techniques like stemming, or substitution of word using ontology or synonyms, domain specific tables are common during this phase. Some relevant works include [19, 7]. The cons of such method are that there is a chance of loosing the original context while replacing the word in the original document. (iii) *Searching and Ranking*: This phase depends on how the documents are modeled. Vector Space Model [20] is a common model and our application is also based on this model. (iv) *Presentation*: This phase is about how the search results are presented to the searcher and it depends on the requirement of the application. Popular display format used by Google, Yahoo etc are in one category while newer semantic search applications like Flamenco [21] may be considered as another category of display; however, the differences are blurring with time.

3 PROBLEM

Given a set of documents *D* and a set of query terms *Q* the problem is to select an ordered list of documents $L_{D'}$ such that the set of documents *D'* forming $L_{D'}$ is subset of *D*

$$D' \subseteq D$$

Furthermore documents in $L_{D'}$ should be ordered by rank(*R*) or decreasing order of relevancy with respect to *Q*. Since *R* is subjective to the specific needs of a user, we attempt to quantify *R* as a function *f* of term frequency (TF) and inverse document frequency (IDF) [22] of *Q* in the $L_{D'}$.

$$TF(d) = \frac{No.\ of\ times\ Q\ appears\ in\ d}{Total\ no.\ of\ terms\ in\ d}, \quad d \in D'$$

$$IDF = \frac{No\ of\ documents\ in\ D}{Total\ no.\ of\ documents\ containing\ Q}$$

$R(d) = f(TF, IDF)$, where *f* is a function on *TF* and *IDF*.

The problem is to select $L_{D'}$ such that following optimization requirements are met:

Precision Maximization: Given a set of documents (D_r) such that $D_r \subset D$ and D_r is the set of established relevant documents with respect to Q and D' is the set of documents selected by the system as relevant documents with respect to Q. Precision *P* is defined as

$$P = \frac{D^{'} \cap D_r}{D^{'}}$$

P quantifies the measure the fraction of documents in $L_{D'}$ that is relevant. The requirement is to maximize P

$$P \approx 1.$$

Recall Maximization: Given a set of documents (D_r) such $D_r \subseteq D$ and D_r is the set of established relevant documents with respect to Q and D' is the set of documents selected by the system as relevant documents with respect to Q. Recall *Rec* is defined as

$$\mathrm{Re}c = \frac{D_r \cap D'}{D_r}$$

Rec quantifies the measure the fraction of relevant documents that appear in result set $L_{D'}$. The requirement is to maximize *Rec*

$$\mathrm{Re}c \approx 1.$$

4 APPROACH

We now present our approach to semantic mining. Figure 2 shows a high level overview of SARF. *Indexing* and *Searching* form the core components of SARF.

Indexing module accepts documents and associated meta-data of a corpus to generate its index. Similarly, indices for related dictionaries, ontologies, and synonyms are also generated. These indices are stored in the *index repository*. Users can issue queries to the Searcher module via the *query interface* provided by the Semantic Search Tool (SST). Statistical information related to document repositories and their indices are generated and are made available through the user interface of SST.

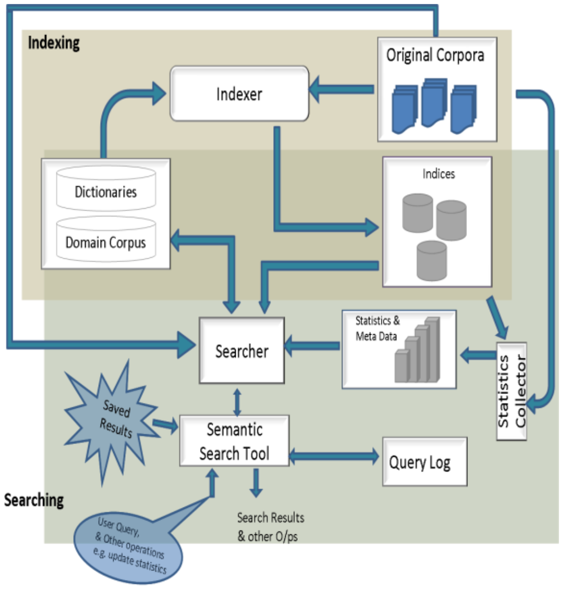

Figure 2: System Architecture

SARF is designed to handle multiple document repositories. As shown in Figure 2, indexing of each repository is done while taking into account relevant meta-data and it is done by extending and customizing Lucene APIs [3]. As new documents keep getting added in each of the repository, our indexer updates corresponding indices periodically to account for them. Thus those new documents are available for searching. The frequency of such updates is repository dependent.

Our searcher supports Lucene query syntax. Searcher has overloaded methods to handle a variety of search requests including filtering and sorting options. It is designed to search across several indices and is multithreaded for enhanced performance. Documents matching the query are ranked by the searcher using TF-IDF based relevance ranking algorithm.

SST supports varying levels of query complexity from simple pattern matching queries to complex ones requiring query expansion, boosting, sorting as well as filtering. For example, user could expand the query to include synonyms using say WordNet. SST accepts the results and displays them to the user. SST has the ability to present the results in multiple ways. For example users may choose to view results chronologically rather than using relevance ranking. The result set presented displays select segments from the narrative text containing query terms. After going through the result set users have the option to view the entire document using a clickable link. On clicking the document link, the query interface fetches the document from the document repository for viewing.

Specific information like the earliest and latest date is used as optional search options for filtering purposes. Documents do contain multiple narrative fields. For example in MedWatch 3500 form [23], which is for reporting medical device related adverse events, the narrative field "Event Description" is perhaps the most important ones for text mining. In case of follow-up reports the narrative field "Manufacturer Narrative" could be more important than other narratives. Thus ranking of the documents can be dynamic according to the type of the document. This concept is extended to all the corpora.

SST also has components for logging user queries, exporting search results and user account/access management. Online help and documentation are available. All user queries are maintained in a querylog. Querylog is designed to serve two purposes - firstly it enables 'autosuggestion' feature and secondly enables understanding of usage pattern to help optimize user interface in future. Account management module addresses management of user accounts and access control issues of the framework.

In summary, SARF with SST is a customizable, scalable and web-enabled system addressing the needs of handling different types of documents for text mining.

5 CASE STUDY

Reviewers at the Center for Devices and Radiological Health spend days, going through narratives of medical device adverse event reports, to identify reports relevant to specific issue being looked into. Typically, even among the filtered reports based on structured information like date range, and product code, only less than ten percent of them are relevant. With our tool, it is now possible to identify those ten percent relevant documents within minutes, and the reviewer needs to look only into those documents. In a specific instance, a reviewer had to go through over 2000 documents to look for specific information. The reviewer found 203 documents that had the relevant information, in 4 days. By using the same criteria used by the reviewer to identify the relevant documents, our tool returned the same 203 documents in couple of minutes. Most of the time was in getting the query right and few seconds for getting the search results. Use of SARF could save as much as 90% of time, which is spent in locating relevant documents. We believe that reviewers responsible for signal detection and adverse event analysis would benefit the most.

6 DISCUSSION AND CONCLUSIONS

In this paper, we presented a generic, practical and scalable approach to assist decision makers/regulators in searching for relevant textual information from large scale data repository (in terabytes). Our approach is based on text mining and it is independent of the type of document.

Our SST with SARF is capable of analyzing millions of regulatory documents to provide FDA analysts with an intuitive web-based tool for signal detection and evaluation. Our approach utilizes a range of techniques innovatively, including use of structured information available in the document for sorting and filtering, while utilizing the narrative unstructured information for context and semantics. Semantic analysis is achieved using, but not limited to, dictionaries, ontologies, and standards.

Our tool is powerful and it provides a wide range of queries ranging from simple to very strict. Strictness is obtained using meta-data information or query syntax or in combined. Historical user queries are maintained and used as suggestions to future users. User can export the selected search results and view details later by importing them using the application. The tool also provides dictionaries and look-up table which are specific to repository domain to assist users in constructing appropriate queries. Adding new dictionary or look up table is very easy. Most frequent words in the top-k search results are shown to the users which in turn help in reformulating new queries. Our application serves common trend of querying i.e. starts with a naive query and narrow down the search with the help of the meta-data or with using those frequent words. Search results can be displayed in interactive time series and relevance scores graphs. These features are very help to the regulators.

While automatic query modification has its advantages but it also suffers from the problem of synonyms not controlled well. It means that by letting allowing the system to select automatic modification lets using incorrect concepts/semantic and that reduces the precision. But regulators are domain experts in most of times, hence allowing them selecting the appropriate synonyms would be more helpful thereby reducing the unwanted results i.e. increasing the precision. This is exactly our application provides. Hence user has the option to start with automatic and rejects the unwanted concepts system suggested from the query. Searching across multiple corpuses is also allowed in our approach but the result for this part is now mentioned in the paper due to confidentiality of the data.

ACKNOWLEDGEMENT

This work is supported in part by an appointment to the Research Participation Program for the US Food and Drug Administration, Center for Devices and Radiological Health, Office of Science and Engineering Laboratories, administered by the Oak Ridge Institute of Science and Education and an agreement between the Department of Energy and FDA.

REFERENCES

[1] Weekly Epidemiological Record, 7 July 2006 http://www.who.int/wer/2006/wer8127.pdf

[2] S D Sudarsan, R P Jetley, B Fitzgerald and S Ramaswamy, "Implementing an information retrieval for an organizational repository" ApacheCon US 2009, 2-6 November 2009

[3] Apache Lucene, http://lucene.apache.org/java/docs/index.html

[4] D I Moldovan, and R Mihalcea, "A WordNet-Based Interface to Internet Search Engines" In Proc of the Eleventh Intl. Florida Artificial Intelligence Research Society Conference FLAIRS-98, pp. 275-279

[5] R Guha, R McCool and E Miller, "Semantic Search" In Proc of the Twelfth Intl Conf on World Wide Web, WWW'03, pp.700-709, ACM Press 2003

[6] V Lopez, M Pasin and E Motta, "Aqualog: An ontology-portable question answering system for the semantic web" In Proc of the 2nd European Semantic Web Conference, Greece 2005

[7] P M Kruze, A Naujoks, D Roesner and M Kunze, "Clever search: A wordnet based wrapper for internet search engines" Journal of Computing Research Repository - CORR, vol. abs/cs/050, 2005

[8] H Zhao, W Meng, Z Wu, V Raghavan and C Yu, "Fully automatic wrapper generation for search engines" In proc of the 14th Intl Conf on World Wide Web WWW'05, pp.66-75

[9] E S de Moura, G Navarro, N Ziviani and R Beaza Yates, "Fast searching on compressed text allowing errors" In proc of the 21st annual intl ACM SIGIR conf on Research and Development in information retrieval SIGIR'98, pp.298-306

[10] R A Baeza Yates and B Ribeiro Neto, "Modern Information Retrieval" Addison-Wesley Longman Publishing Co., Inc., Boston, MA USA 1999

[11] M A Hearst, "Untangling text data mining" In proc of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics" ACL'99 pp.3-10

[12] C H Chang, M Kayed, M R Girgis and K F Shaalan, "A survey of web information extraction systems"

IEEE Trans on Knowledge and Data Eng., 18:1411-1428, Oct'06

[13] J Allan, editor, "Topic detection and tracking: event-based information organization" Kluwer Academic Publishers, Norwell MA USA 2002

[14] Moffat, Alistair, and Z Justin, ”Self-indexing inverted files for fast text retrieval” ACM Trans. Inf. Syst. Vol. 14, issue 4 October 1996

[15] S Anna, A Periklis and N Nicolas, “Overview and semantic issues of text mining” SIGMOD Rec, vol.36, issue 3, September 2007 ACM

[16] Mooney, R Bunescu, “Mining knowledge from text using information extraction” SIGKDD Explorations, 2005, pp.3-10

[17] M Mitra, A Singhal, C Buckley, “Improving automatic query expansion” Proc. of the 21st annual intl ACM SIGIR conf on Research and development in information retrieval, SIGIR '98 pp.206-214

[18] C Mangold, “A survey and classification of semantic search approaches” Intl. J. Metadata Semant. Ontologies, vol. 2, issue 1, September, 2007, pp.23-34

[19] W Dakka, P Ipeirotis and KR Wood, “Automatic construction of multifaceted browsing interfaces” Proc of the 14th ACM intl conf on Information and knowledge management, CIKM '05, pp.768-775

[20] Vector Space Model (VSM) http://en.wikipedia.org/wiki/Vector_space_model

[21] Flamenco http://flamenco.berkeley.edu

[22] J Sparck Karen, “A statistical interpretation of term specificity and its application in retrieval” Document retrieval systems 1988 pp.132-142

[23] MedWatch 3500 Form http://www.fda.gov/Safety/MedWatch/HowToReport/DownloadForms/default.htm