World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:8, 2008

Mining Association Rules from Unstructured Documents

Hany Mahgoub

***Abstract*—**This paper presents a system for discovering association rules from collections of unstructured documents called EART (Extract Association Rules from Text). The EART system treats texts only not images or figures. EART discovers association rules amongst keywords labeling the collection of textual documents. The main characteristic of EART is that the system integrates XML technology (to transform unstructured documents into structured documents) with Information Retrieval scheme (TF-IDF) and Data Mining technique for association rules extraction. EART depends on word feature to extract association rules. It consists of four phases: structure phase, index phase, text mining phase and visualization phase. Our work depends on the analysis of the keywords in the extracted association rules through the co-occurrence of the keywords in one sentence in the original text and the existing of the keywords in one sentence without co-occurrence. Experiments applied on a collection of scientific documents selected from MEDLINE that are related to the outbreak of H5N1 avian influenza virus.

***Keywords*—**Association rules, information retrieval, knowledge discovery in text, text mining.

I. Introduction

THE information age is characterized by a rapid growth for information available in electronic media such as databases, data warehouses, intranet documents, business emails and www. This growth has created a demanding task called Knowledge Discovery in Databases (KDD) and in Texts (KDT). Therefore, researchers and companies in recent years [7, 13] focused on this task and significant progress has been made. Text Mining (TM) and Knowledge Discovery in Text (KDT) are new research areas that try to solve the problem of information overload by using techniques from Data Mining, Machine Learning, Natural Language Processing (NLP), Information Retrieval (IR), Information Extraction (IE) and Knowledge Management (KM) [7].

The main goal of text mining is to enable users to extract information from large textual resources. The final output of the mining process varies and it can only be defined with respect to a specific application. Most Text Mining objectives fall under the following categories of operations: Feature Extraction, Text-base Navigation, Search and Retrieval, Categorization (supervised classification), Clustering (unsupervised classification), Summarization, Trend Analysis, Association Rules and Visualization [7].

Paper submitted, 22-6-2006.

Hany Mahgoub is with the Knowledge and Language Engineering department, Faculty of Computer Science, Otto-von Guericke University Magdeburg, Germany (e-mail: mahgoub@iws.cs.uni-magdeburg.de).

Association rule is one of the important techniques of data mining. Association rules highlight correlations between keywords in the texts. Moreover, association rules are easy to understand and to interpret for an analyst. In this paper, we focus on the extraction of association rules amongst keywords labeling the documents.

Since collections of documents are a valuable source of knowledge and considered as assets, it is worthwhile to invest in efforts to get access to these sources. With the expanding WWW many documents first appear online before there are printed versions available; sometimes there even only exist online versions. Therefore any support offered by automatic analysis tools and techniques like *document mining* will be highly valued [3]. This is the main motivation for this paper. The availability of huge amounts of electronic documents allows for a new approach to knowledge acquisition. *Document mining* or *text mining* is the attempt to support and –at least partially- automate the process of finding and extracting interesting pieces information from documents.

The problem with text mining is that unlike tabular records in databases, documents are not structured and normalized so that they could be easily recognized by computers. The *lack of explicit structure* raises the difficulty of uncovering the implicit knowledge inside the documents. It is hard to extract and represent abstract concepts from a natural text because the same concept may be expressed in many ways. The emerging standard eXtensible Markup Language (XML) and its supporting techniques will help to structure and automate indexing of document. This makes part of the semantics of a document explicit and thus machine processable.

This paper presents EART (Extract Association Rules from Text), a system for discovering association rules from collections of textual documents. EART depends on word feature to extract association rules. Where, several researchers generally use the word feature to represent text [1, 4, 15]. We concentrate in our work on the analysis of the keywords in the extracted association rules from two sides. The first side is the co-occurrence of the keywords in one sentence in the original text. The second side is the existing of the keywords in one sentence without co-occurrence.

The outline of the paper is as the follows: in section II, we present the EART system. Experimental results are present in section III. Section IV presents the related work. Section V provides conclusion and future work.

II. EART System Architecture

The proposed EART system for extraction of association rules from texts (collection of documents) is shown in Fig. 1.

It discovers association rules patterns of co-occurrence amongst keywords labeling the collection of textual documents. The main characteristic of EART is that the system integrates XML technology (to transform unstructured documents into structured documents) with information retrieval scheme (TF-IDF) and data mining technique for generating association rules. Since most of the previous works only applied on the title and abstract of documents. Our system is focus on the whole and specific parts of the documents that have the same structure. The EART system treats each sentence in a document as a "basket" ignoring the grammatical information. The system begins with selecting collections of documents from the web or internal file systems. The EART system consists of four phases: *structure phase* (transformation, filtration and stemming, on the unstructured documents), *index phase* (calculate the weighting scheme for all keywords in all documents), *text mining phase* (applying our algorithm for generating association rules) and *visualization phase* (visualization of results).

A.. Structure Phase

This phase begins with the transformation process of the original unstructured documents. This transformation aims to obtain the desired representation of documents in XML format. After that, the documents are filtered to eliminate unimportant words (e.g. articles, determiners, prepositions and conjunctions, etc.) by using a list of stop words. The resulting documents are processed to provide basic information about the content of each document.

1. Transformation

The system accepts a different number of documents formats (doc, txt, rtf, etc.) to convert them into the XML tags amenable for further processing. The result of transformation process should preserve and make explicit (markup) structural information. This requirement holds independent of the specific format of the source.

In this process, the original unstructured documents written in different format are transformed into structured documents in XML format (tags).

Each tag contains a text and the documents are representing in nested tree or hierarchical structure as follows:

```
<DOCUMENTS>
  <DOCUMENT>
     <TITLE> text <\TITLE>
     <AUTHOR> text <\AUTHOR>
     <ABSTRACT> text <\ABSTRACT>
  <INTRODUCTION>text<\INTRODUCTION>
                        ….
     <CONCLUSION>text<\CONCLUSION>
     <REFERENCES>text <\REFERENCES>
  <\DOCUMENT>
<\DOCUMENTS>
```

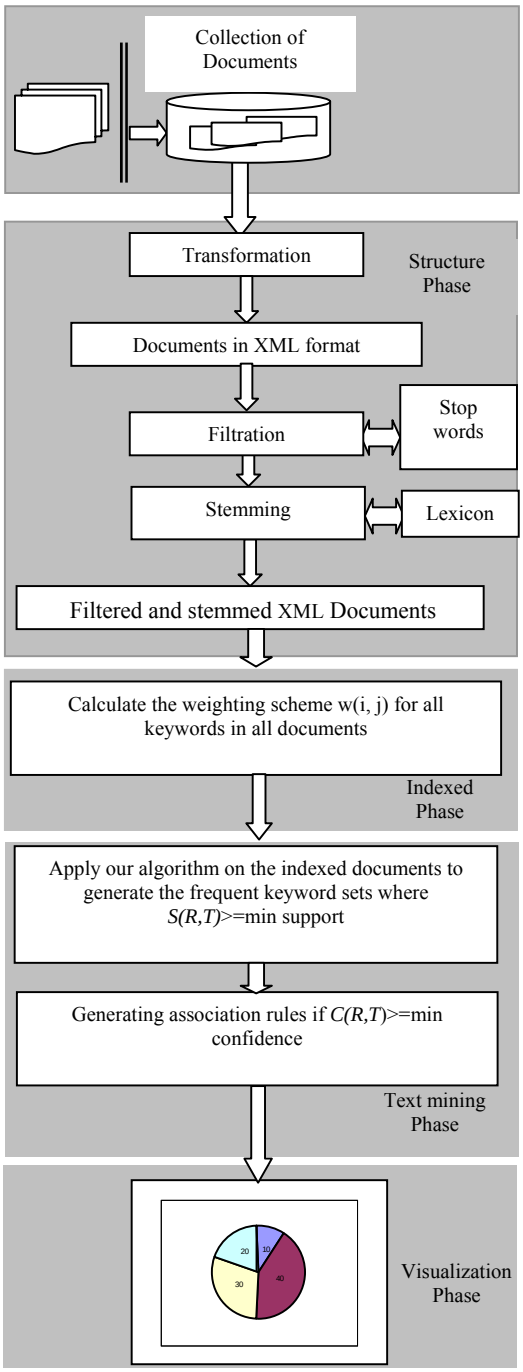

Fig. 1 EART system architecture

2. Filtration

In this process, the documents are filtered by removing the unimportant words from documents content. Therefore, the unimportant words get discarded or ignored (e.g. articles, pronouns, determiners, prepositions and conjunctions, common adverbs and non-informative verbs (e.g., be)) and more important or highly relevant words are single out. We build a list of unimportant words called stop words, where the system checks the documents content and eliminate these unimportant words from the documents content.

In addition, the system replaces special characters, parentheses, commas, etc., with distance between words in the documents.

After the filtration process the system does *word stemming*, a process that removes a word's prefixes and suffixes (such as unifying both isolated and isolating to isolate). Instead of using a generic stemming algorithm such as Porter's, we used a stemming dictionary (lexicon) that we have been designed for the medical domain.

B. Index Phase

If the textual data is indexed, either manually or automatically, the indexing structures can be used as a basis for the actual knowledge discovery process. As a manual indexing is a time-consuming task, it is not realistic to assume that such a processing could systematically be performed in the general case.

Automated indexing of the textual document base has to be considered in order to allow the use of association extraction techniques on a large scale. Techniques for automated production of indexes associated with documents can be borrowed from Information Retrieval field [10]. Each document is described by a set of representative keywords called index terms. An index term is simply a word whose semantics helps in remembering the document's main themes [1]. It is quite obvious that different index terms have varying relevance when used to describe document contents in particular document collection. This effect is captured through the assignment of numerical weights to each index term of a document. The techniques for automated production of indexes associated with documents usually rely on frequency-based weighting schemes.

The weighting scheme TF-IDF (Term Frequency, Inverse Document Frequency) is used to assign higher weights to distinguished terms in a document, and it is the most widely used weighting scheme which is defined as (cf. [2] [9] [10]):

$$w(i,j) = tfidf(d_i, t_j) = \begin{cases} Nd_i, t_j * \log_2 \dfrac{|C|}{Nt_j} & \text{if } Nd_i, t_j \geq 1 \\ 0 & \text{if } Nd_i, t_j = 0 \end{cases} \quad (1)$$

where $w(i,j) \geq 0$, Nd_i, t_j denotes the number the term t_j occurs in the document d_i (term frequency factor), Nt_j denotes the number of documents in collection C in which t_j occurs at least once (document frequency of the term t_j) and $|C|$ denotes the number of the documents in collection C. The first clause applies for words occurring in the document, whereas for words that do not appear ($Nd_i, t_j = 0$), we set $w(i,j)=0$.

Document frequency is also scaled logarithmically. The formula: $\log_2 \dfrac{|C|}{Nt_j} = \log C - \log Nt_j$ gives full weight to words that occur in 1 document ($\log C - \log Nt_j = \log C - \log 1 = \log C$). A word that occurred in all documents would get zero weight ($\log C - \log Nt_j = \log C - \log C = 0$).

This weighting scheme includes the intuitive presumption that: the more often a term occurs in document, the more it is representative of the content of the document (term frequency) and the more documents the term occurs in, less discriminating it is (inverse document frequency). Once a weighting scheme has been selected, automated indexing can be performed by simply selecting for each document the keywords that satisfy the given weight constraints. The major advantage of an automated indexing procedure is that it reduces the cost of the indexing step.

1. Weight Constraints

The notation of term relevance with respect to a document collection is a central issue in Information Retrieval. We assign for each keyword its score (weight value) based on maximal TF-IDF (maximal with respect to all the documents in the collection).

Our aim is to identify and filter the keywords that may not be of interest in the context of the whole document collection either because they do not occur frequently enough or they occur in a constant distribution among the different documents. Our system uses a statistical relevance-scoring function that assigns a score to each keyword based on their occurrence patterns in the collection of documents, and the top *N* taken as the final set of keywords to be used in the text mining phase. The system sort the keywords based on their scores and select only the top *N* frequent keywords up to *M* % of the number of running words (for a user specified *M*). This is the criteria of using the weight constraints.

C. Text Mining Phase

This phase presents a way for finding information in a collection of indexed documents by automatically retrieving relevant association rules between keywords. A formal definition of association rules mining follows [10]. Given a set of keywords $A = \{w_1, w_2, ..., w_n\}$ and a collection of indexed documents $T = \{t_1, t_2, ..., t_m\}$, where each document t is a set of keywords such that $t \subseteq A$. Let w_i be a set of keywords. A document t is said to contain w_i if and only if $w_i \subset t$. An association rule is an implication of the form $w_i \Rightarrow w_j$ where $w_i \subset A$, $w_j \subset A$ and $w_i \cap w_j = \phi$. There are two important basic measures for association rules, supports (s) and confidence (c). The rule $w_i \Rightarrow w_j$ has support s in the collection of documents T if s% of documents in T contains $w_i \cup w_j$. The support is calculated by the following formula:

$$s(R,T) = \frac{s(w_i \cup w_j)}{|T|} \quad (2)$$

The rule $w_i \Rightarrow w_j$ holds in the collection of documents T with confidence c if among those documents that contain w_i, c % of them contain w_j also. The confidence is calculated by the following formula:

$$c(R,T) = \frac{s(w_i \cup w_j)}{s(w_i)} \quad (3)$$

An association rule *R* generated from a collection of texts *T* is said to satisfy support and confidence constraints σ and γ respectively if

$$c(R,T) \geq \gamma \text{ and } s(R,T) \geq \sigma$$

To simply notations, a rule satisfying given support and confidence constraints will be simply written as:

$$w_i \Rightarrow w_j \quad s(R, T) / c(R, T).$$

Our algorithm is based on the Apriori algorithm [11] for generating association rules. The algorithm not make multiple scanning on the original documents as Apriori algorithm but scan only the generated XML file which contains all keywords that satisfy the threshold weight value and their frequencies in each document. Our algorithm is as follows:

1. Let *N* denote the number of top keywords that satisfy the threshold weight value.

2. Store the top *N* keywords in index XML file along with their frequencies in each document, TF-IDF and documents ID. Each index is a 4-XML tag of <doc-id, keyword, keyword-frequency, TF-IDF>.

3. Scan the index XML file and find all keywords that satisfy the threshold minimum support. These keywords are called large frequent keyword sets. First of all, we find all large frequent 1-keywordsets, L_1.

4. In $k \geq 2$, the candidate keywords of size k are generated from large frequent (*k*-1)-keyword sets, L_{k-1}.

5. Scan the index file, and compute the frequency of candidate keyword sets that generated in step 4.

6. Large frequent *k*-keyword sets L_k, which satisfy the minimum support, is found.

7. For each frequent keyword set, find all the association rules that satisfy the threshold minimum confidence.

D. Visualization Phase

This phase visualizes the results of EART system in different format such as IF-Then rules and web form.

III. Experimental Results

To investigate the use of EART system to extract association rules in text, we applied it on a selecting sample of 100 recent scientific documents from Medline abstracts that are related to the outbreak of H5N1 avian influenza virus. In our experiments, we applied the system on the abstract part of the 100 documents. The collection of the 100 abstract (corpus) is 220 KB in size and contained on average 18,351 unique words. Each abstract contained on average 183 unique words. After the filtration process the collection of abstracts contained on average 8000 unique word.

A. Representation of Documents in XML Format

The EART system begins after loading the documents, the user enter the three threshold values (weight, support and confidence) and determine the system deals with all parts or specific parts of documents. Fig. 2 shows the transformation process into XML format (tags) for only abstract part in the collection of 100 documents at threshold weight value 60%, support 13% and confidence 50%.

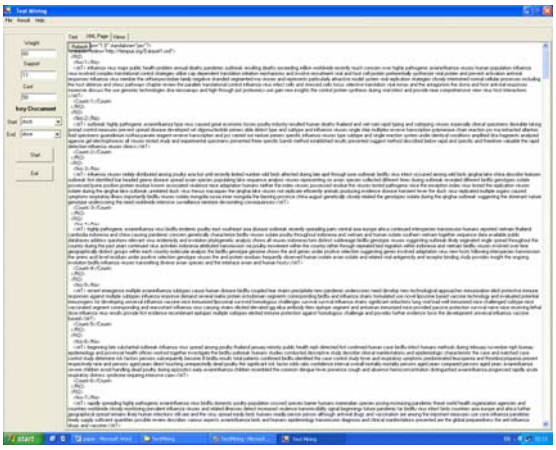

Fig. 2 Documents in XML format

B. The Effect of the TF-IDF Scheme on Keywords

Table I shows an example of the results of the indexed phase for TF-IDF measure. The table shows the scores of the keywords for one abstract of one document with threshold weight value *M*=60%.

TABLE I
TFIDF OF THE KEYWORDS OF ONE ABSTRACT OF SCIENTIFIC PAPER

Keyword	TF-IDF	Keyword	TF-IDF
avaininfluenza	33.21928	threat	4.058894
H5N1	30.2535	disease	4.058894
influenza	28.21928	emergence	3.643856
Pathogenic	19.93157	severe	3.643856
Highly	16.93157	produced	3.321928
transmission	15.34601	results	3.058894
virus	11.28771	against	2.943416
…	…	…	…
outbreak	10.42179	antiviral	2.736966
infect	8.00734	target	2.643856
Vietnam	7.673003	need	2.556393
drugs	7.643856	increased	2.395929
protein	6.643856	public	2.395929
amino	6.643856	health	2.184425
Acid	6.643856	resistant	2
cells	5.643856	analysis	1.943416
isolate	5.643856	subtype	1.68966
birds	5.643856	new	1.434404
…	…	…	…
cause	4.321928	level	1.120294

The total number of running keywords before computing the weight is 84 keyword and the number of keywords *N* after computing the weight is 60 keyword. The keywords that appear in the shaded region had discarded in this process because they did not satisfy the weight constraint.

After applying the system on the 100 Medline abstracts with the same threshold weight value *M*=60%, it is noticed that the total number of running keywords before computing the weight is 7870 keyword and the number of keywords *N*

after computing the weight becomes 4928 keyword as shown in Fig. 3.

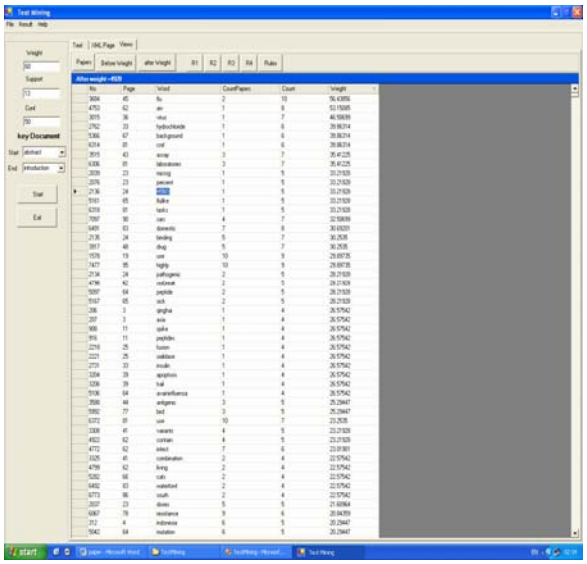

Fig. 3 Number of keywords that satisfy threshold weight *M*= 60%

In our experiments, we applied the system on the first 25 abstracts from the collections of 100 abstracts. Then apply it on other 25 abstract in addition to the first 25 abstract (i.e. apply it on 50 abstract). After that, apply the system on all 100 abstract. The experiments applied at different threshold weight values such as 30%, 60% and 80%. From Fig. 4 the results reveal that the number of the top *N* of keywords is always greater than the *M*% of the running keywords.

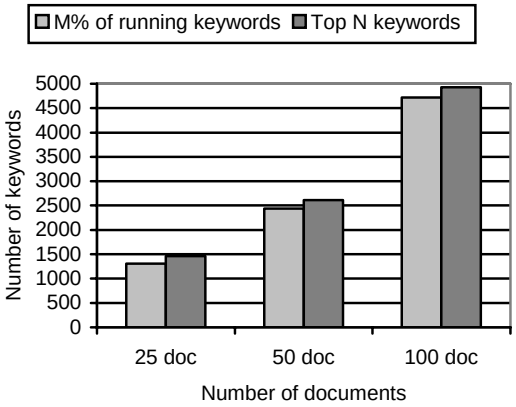

Fig. 4 The number of keywords at *M*=60%

Fig. 4 shows for example when applying the system on the corpus of 100 abstract with the threshold weight value *M*=60% that the number of top *N* equals to 4928 word, while the actual number of running keywords at *M*=60% must be 4722 word. The reason of this is that there are many keywords of the top *N* share with other keywords in the same computing weight value as shown in Table I. For example the words "protein, amino and acid" has the same weight value 6.643856. This always make *N* is greater than *M*.

C. Generation of Association Rules

The selected documents provided information in XML elements, e.g., title author(s), date, publication status (yes, no), and abstract (cf. Fig. 5).

Text: # 24
Title: Investigation of avianinfluenza outbreak in humans
Author: Kash JC, Goodman AG, Korth MJ, Katze MG.
Abstract: Recently, there is much concern over the highly pathogenic avian influenza H5N1 viruses into the human population. Influenza virus has evolved complex translational control strategies that utilize cap-dependent translation initiation mechanisms and involve the recruitment of both viral and host-cell proteins to preferentially synthesize viral proteins and prevent activation of antiviral responses.
Infected birds have been the primary source of influenza infection in humans in Asia. But the transmission from poultry to humans is very limited at present, and requires a direct exposure to live birds, whereas there was no significant risk related to eating well-cooked poultry meat…

Fig. 5 Example of Medline abstract (abbreviated)

In our experiments, only one textual field (abstract) is used to generate the association rules between keywords. Extraction of association rules was achieved by using our above algorithm in section 2. The EART extracted two types of association rules based on the analysis of the keywords in the extracted rules. This analysis has been done through the co-occurrence of the keywords in one sentence in the original text and the existing of the keywords in one sentence without co-occurrence in the original text. Fig. 6 shows a sample of generated association rules (using a weight 60%, support 13%, and confidence threshold 50%), where the number presented at the end of each rule is the rule's confidence.

After the analysis of the keywords in the extracted association rules, we noticed that the appearing sequence of the keywords in some extracted association rules is the same as the co-occurrence of these keywords in the original text (cf. Fig. 5).

highly, pathogenic, avianinfluenza --> H5N1	80 %
outbreak, H5N1, poultry --> Asia	72 %
Viruses, H5N1, isolate --> Vietnam	69%
avianinfluenza, virus , poultry--> human	70 %
outbreak, avianinfluenza, Thailand --> Vietnam	85 %
epidemic, H5N1, avianinfluenza--> Asia	83 %
infect, humans, Asia --> influenza	83 %
pandemic, virus --> human ,transmission	60 %
virus ,isolate--> influenza, H5N1	50 %
spread--> highly, pathogenic, avianinfluenza	67 %

Fig. 6 A sample of the generated association rules

Some examples of these rules are:
"*highly, pathogenic, avianinfluenza --> H5N1* 80%"
which tells us that in 80 percent of the texts, where the three words (*highly, pathogenic, avianinfluenza*) occurred within 3 consequent words, the word *H5N1* co-occurs within 4 words.
"*outbreak, H5N1, poultry --> Asia 72%"*

which tells us that in 72 percent of the texts, where the three words (*outbreak, H5N1, poultry*) occurred within 3 consequent words, the word *Asia* co-occurs within 4 words.

"Viruses, H5N1, isolate --> Vietnam 69%"

which tells us that in 69 percent of the texts, where the three words (*Viruses, H5N1, isolate*) occurred within 3 consequent words, the word *Vietnam* co-occurs within 4 words.

In addition, there are other extracted association rules that get the relation of the existing of the keywords in one sentence and it is not dependent on the keywords sequence in the sentence.

Some examples of these association rules are:

"infect, humans, Asia --> influenza 83%"

which tells us that in 83 percent of the texts, the four words (*infect, humans, Asia, influenza)* occurred within one sentence without consequence .

"pandemic, virus --> human, transmission 60%"

which tells us that in 60 percent of the texts, the four words (*pandemic, virus, human, transmission)* occurred within one sentence without consequence.

IV. Related Work

Much text mining or knowledge discovery in text paradigms have been based on simple forms of text categorization as in KDT [13]. However, recently several researchers have applied traditional rule induction methods to discover relationships from textual data. In [14], the authors find associations between the keywords or concept labeling the documents using background knowledge about relationships among the keywords. The purpose of the knowledge base is to supply unary or binary relations amongst the keywords labeling the documents. In [10] the authors presented two examples of text mining tasks, association extraction and prototypical document extraction, along with several related NLP techniques. In the case of association extraction task, they had extracted association rules from a collection of indexed documents, designed to answer specific queries expressed by the users. A way of finding information in a collection of indexed documents by automatically retrieving relevant associations between keywords was presented. In addition, there are several researchers [5, 6, 8] applied existing data mining techniques to discover episode rules from text. Where Episode rule mining is used for language analysis because it preserve the sequential structure of terms in a text document.

V. Conclusion and Future Work

This paper has presented the EART system for extract association rules from collection of textual documents based on word feature. EART system implemented using C# and XML. The EART extracted two types of association rules depending on the analysis of the keywords in the extracted rules. Our experiments applied on the 'abstract' part of collection of Medline documents but our system can be applied on overall parts of the documents. So our system is flexible to work on all parts or specific parts of documents. This work is designed to support English language.

We are currently in the stage of optimizing the performance of the system and carrying out extensive testing of applying EART system on the overall parts of scientific documents. EART is domain-independent so we intend to carry out tests in other domains.

References

[1] B. Lent, R. Agrawal, and R. Srikant, "Discovering trends in text Databases," *KDD'97*, 1997, pp.227-230.

[2] C. Manning and H Schütze, Foundations of statistical natural language processing (MIT Press, Cambridge, MA, 1999).

[3] D. Rösner and M. Kunze, "The XDOC Document Suite -- A Workbench for Document Mining," *In Text Mining – Theoretical Aspects and Applications, Advances in Soft Computing, Physica – Verlag*, 2003, 113-130.

[4] G. W. Paynter, I. H. Witten, S. J. Cunningham, and G. Buchanan, "Scalable browsing for large collections: a case study," *5th Conf. digital Libraries, Texas*, 2000, 215-218.

[5] H. Ahonen, O. Heinonen, M. klemettinen, and A. Inkeri Verkamo, "Mining in the phrasal frontier," *Proc. PKDD'97.1st European Symposium on Principle of data Mining and Knowledge Discovery, Norway, June, Trondheim,* 1997.

[6] H. Ahonen, O. Heinonen, M. Klemettinen, and A. Inkeri Verkamo, "Applying data mining technique for descriptive phrase extraction in digital document collections,"*Proc. of IEEE Forum on Research and technology Advances in Digital Libraries, Santa Barbra CA,* 1998, 2-11.

[7] H. Karanikas and B. Theodoulidis, "Knowledge discovery in text and text mining software," *Technical Report, UMIST Departement of Computation, January* 2002.

[8] H. Mannila, H. Toivonen and A. I. Verkamo, "Discovery of frequent episodes in event sequences," *Data Mining and Knowledge Discovery*, 1(3), 1997b, pp. 259-289.

[9] J. Paralic and P. Bednar, "*Text mining for documents annotation and ontology support* (*A book chapter in: "intelligent systems at service of Mankind," ISBN 3-935798-25-3, Ubooks, Germany*, 2003*).*

[10] M. Rajman and R. Besancon, Text mining: natural language techniques and text mining applications. *Proc. 7th working conf. on database semantics (DS-7), Chapan &Hall IFIP Proc. Series. Leysin, Switzerland Oct.* 1997, 7-10.

[11] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," *In Jorge B. Bocca, Matthias Jarke, and Carlo Zaniolo, editors, Proc. 20th Int. conf. of very Large Data Bases, VLDB, Santigo, Chile*, 1994, 487-499.

[12] R. Baeza-Yates and B. Ribeiro-Neto, *Modern information retrieval (Addison-Wesley, Longman publishing company, 1999).*

[13] R. Feldman and I. Dagan, Knowledge discovery in textual databases (KDT), *Proc. 1st nt. Conf. on Knowledge Discovery and Data Mining, 1995.*

[14] R. Feldman and H. Hirsh, "Mining associations in text in the presence of background knowledge," *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, USA,* 1996.

[15] S. Brin, R. Motwani, and C. Silverstein, "Beyond market baskets: generalizing association rules to dependence rules," *KDD'98*, 1998, 39-68.

Hany Mahgoub born in Elmeoufia , Egypt in 1972. He received the B. Sc. and M. Sc. in Computer Science from Faculty of Science, Menoufia University, Egypt in 1993 and 1999 respectively with very Good degree. His research interests include text mining, information extraction, and data mining.

This author became Demonstrator of Computer Science at Math. and Computer Science department, Faculty of Science Menoufia University in 1994, an Assistant lecture of Computer Science at Math. and Computer Science department, Faculty of Science Menoufia University 2000, and Assistant lecture of Computer Science at Computer Science department, Faculty of Computers and Information Menoufia University in July 2002- until Now.

Mr. Mahgoub Since September 2005 became a PhD researcher at Department of Knowledge and Language Engineering, Faculty of Computer Science, Otto-von Guericke-University of Magdeburg, Germany.