ISSN 2321 -9017
Volume 1, No.2, August - September 2013
International Journal of Bio-Medical Informatics and e-Health
Available Online at http://warse.org/pdfs/ijbmieh01122013.pdf

KMDC: Knowledge based Medical Document Clustering System using Association Rules Mining

Noha E. Negm[1], Passent ElKafrawy[2], Mohamed Amin[3] , Abdel-Badeeh M. Salem[4]
[1,2,3] Faculty of Science, Menoufia University, Shebin El-Kom, Egypt
[4] Faculty of Computers and Information, Ain Shams University, Cairo, Egypt

ABSTRACT

Domain knowledge plays an important role in knowledge discovery and management such as ranking search results, displaying summarized knowledge of semantics and clustering results into topics. In clustering of medical documents, domain knowledge helps to improve the quality of mined knowledge in addition to the mining efficiency. In this paper, we have proposed Knowledge based Medical Document Clustering system using association rules mining (KMDC). Association rules are generated from the informative terms that are frequently occurring and provide knowledge of the domain. KMDC system composed of four main stages: a) online query submission and document retrieval, b) text representation and preprocessing, c) mining association rules using MTHFT algorithm, and d) clustering PubMed abstracts into various clusters. Since each cluster contains relevant articles using association rules as topics. The efficiency, accuracy and scalability of KMDC system was measured using Precision, Recall and F-measure and compared to the existing clustering algorithms like Bisecting K-means and FIHC. The experimental results show that KMDC system is more applicable to scientific related literature, since we obtained higher recall rate and F-measure while handling the search results of PubMed compared to other algorithms.

Key words : Association Rules, Document Clustering, Medical knowledge Discovery, PubMed Abstracts.

1. INTRODUCTION

With the exponential growth of biomedical knowledge through WWW, life science researchers have met a new challenge - how to exploit systematically the relationships between genes, sequences and the biomedical literature [1]. MEDLINE is a major biomedical literature database repository that is supported by the U.S. National Library of Medicine (NLM). It has now generated and maintained more than 15 million citations in the field of biology and medicine, and incrementally adds thousands of new citations every day. PubMed, an information retrieval tool, is one of the most widely-used interfaces to access the MEDLINE database [2]. It allows Boolean queries based on combinations of keywords and returns all citations matching the queries. Due to the inherent complexity of ranking search results and the large numbers of citations (thousands or more) returned from MEDLINE by using general topic queries, researchers can no longer keep up-to-date with all the relevant literature manually, even for specialized topics. For example, for the query “Breast Cancer”, PubMed returns 96,292 citations. If researcher spent only 1 minutes on each abstract and worked 8 hours a day, it would take approximately seven months to find the right answers for his queries and it is overwhelming. Otherwise he commonly browses through the first screen or even the first six results hoping to find the right answers for his queries. It is essential for researchers in medicine to have quick and efficient access to up-to-date information according to their interests and requirements.

All these challenges led to the need for the development of new techniques to support users whose knowledge of medical vocabularies is inadequate to find the desired information and for medical experts who search for information outside their field of expertise. Document Clustering is one of the techniques that can play an essential role towards the achievement of this objective.

Clustering the medical documents into small number of meaningful clusters may facilitate discovering patterns by allowing us to extract a number of relevant features from each cluster. Therefore introducing structure into the data and facilitating the application of conventional data mining techniques can be possible. The produced clusters contain groups of documents that are more similar to each other than to the members of any other group. Therefore, the goal of finding high-quality document clustering algorithms is to determine a set of clusters such that inter-cluster similarity is minimized and intra-cluster similarity is maximized. Since further knowledge extraction and data mining will be applied to the produced clusters, achieving high-quality clustering solution is important [3].

The knowledge of the domain gives an idea of the search results when no prior knowledge about the collection exists. In clustering of medical documents, domain knowledge helps to improve the quality of mined knowledge in addition to the mining efficiency [4]. The controlled vocabularies, such as Gene Ontology (GO), Medical Subject Headings (MeSH) [5], Systematized Nomenclature of Medicine (SNOMED), and

Unified Medical Language System (UMLS) [6] are used as information resource for topics extraction from search results. However, these vocabularies focus on a particular domain; for example, GO for gene products and MeSH for medical topic and disease [4].

In this paper, we have proposed an efficient knowledge based medical documents clustering system (KMDC). The novelty of the system is using the association rules among the extracted informative terms from the retrieved documents as topics for clustering the search results instead of using controlled vocabularies. The system is composed of four main stages : 1) online query submission and document retrieval, 2) text representation and preprocessing, 3) mining association rules, and 4) clustering PubMed abstracts into various clusters.

The organization of the paper is as follows. Section 2 presents the related works. In Section 3, we describe the structure of KMDC system. Section 4 introduces the evaluation parameters for KMDC system. Section 5 discusses the experimental methodology and results. Section 6 concludes the work proposed and presents the future work.

2. RELATED WORK

The amount of published biomedical literature has been growing at an unprecedented rate. A few systems have been proposed to present PubMed retrieval results in a user-friendly way other than a long list, most of which are based on pre-defined categories using classification techniques in which an a priori taxonomy of categories is available rather than clustering techniques. Moreover, they aimed for quicker navigation and easier management of large numbers of returned results [7]. In [8], ***Anne O'Tate*** post-processes retrieved results from PubMed searches and groups them into one of the six pre-defined categories: *important words, MeSH topics, affiliations, author names, journals and year of publication*. Important words have more frequent occurrences in the result subset than in the MEDLINE as a whole, thus they distinguish the result subset from the rest of MEDLINE. In [9], ***McSyBi*** presents clustered results in two distinct fashions: *hierarchical or non-hierarchical*. While the former provides an overview of the search results, the latter shows relationships among the search results. Furthermore, it allows users to re-cluster results by imposing either a MeSH term or ULMS Semantic Type of her research interest. Updated clusters are automatically labeled by relevant MeSH terms and by signature terms extracted from title and abstracts. In [10], ***GOPubMed*** was originally designed to leverage the hierarchy in Gene Ontology (GO) to organize search results, thus allowing users to quickly navigate results by GO categories. Recently, it was made capable of sorting results into four top-level categories. In [11], ***ClusterMed*** can cluster results in six different ways: i) title, abstract and MeSH terms, ii) title and abstract, iii) MeSH terms, iv) author names, v) affiliations and vi) date of publication. In [12], ***XplorMed*** not only organizes results by MeSH classes, it also allows users to explore the subject and words of interest by extracting keywords and their co-occurrences. Although the previous systems can classify search results into pre-defined categories, within each category, PubMed results still consist of long lists without importance-related ranking.

Automatically clustering of PubMed results based on informative terms or phrases extracted from the retrieved abstracts gives a better understanding about the area of research [13]. A suffix-tree based clustering algorithm (STC) is proposed in [14] to identify the common phrases shared by the documents. In [15] Smith has demonstrated the usefulness of suffix tree clustering in browsing events in unstructured text. Readable and unambiguous descriptions of the thematic groups are an important factor of the overall quality of clustering. These provide the users an overview of topics covered in the search results and help them to identify the specific group of documents they were looking for. In [16] the LINGO algorithm employs suffix arrays and singular value decomposition (SVD) to capture thematic labels in a search result for clustering. A Carrot framework was created to facilitate clustering the search results by including algorithms such as STC and LINGO [17]. In [4], FNeTD method for clustering the search results was introduced. The novelty of the approach is nearer terms of the domain used as integrating resource for categorizing the retrieved abstracts. The idea behind the frequent nearer terms of the domain extraction is that terms that come nearer to the domain have some meaning in the biological literature and gives knowledge of the domain. The method provided more technical terms related to search results of the domain than frequently occurring terms in the collections. Furthermore, the generated nearer terms of the domain used as initial term list for domain ontology development.

To the best of our knowledge, all the previous algorithms that are used to automatically clustering PubMed results depend on the older techniques for mining frequent termsets or phrases. Moreover, they don't consider the improvement of the execution time i.e. increase the speed of the clustering process. There is still the need exists for a system to help biomedical researchers in quickly finding relevant, important articles related to their research fields.

3. KNOWLEDGE BASED MEDICAL DOCUMENT CLUSTERING SYSTEM STRUCTURE

The KMDC utilizes knowledge-based approach for clustering medical documents using association rules mining. Generated association rules used to solve the problem of finding clusters of similar items. Figure 1 provides a high-level overview of the proposed system, which proceeds through four stages: a) online query submission and document retrieval, b) text representation and preprocessing, c) mining association rules, and d) clustering PubMed abstracts. The last clustering stage decomposed into three sub-steps: partitions initialization; removing partitions overlapping; and building document clustering by similarity measure.

3.1 Online query submission and document retrieval

KMDC system is designed to accept different formats of retrieved documents such as: HTML web pages, XML and TEXT documents. In our prior research [22], the largest dataset, Reuters, is chosen to exam the efficiency and scalability of our clustering approach. With the standardization of XML as an information exchange language over the web, we found that documents formatted in XML have become quite popular therefore the need for online clustering process especially on XML documents. In this paper we focus on clustering XML documents.

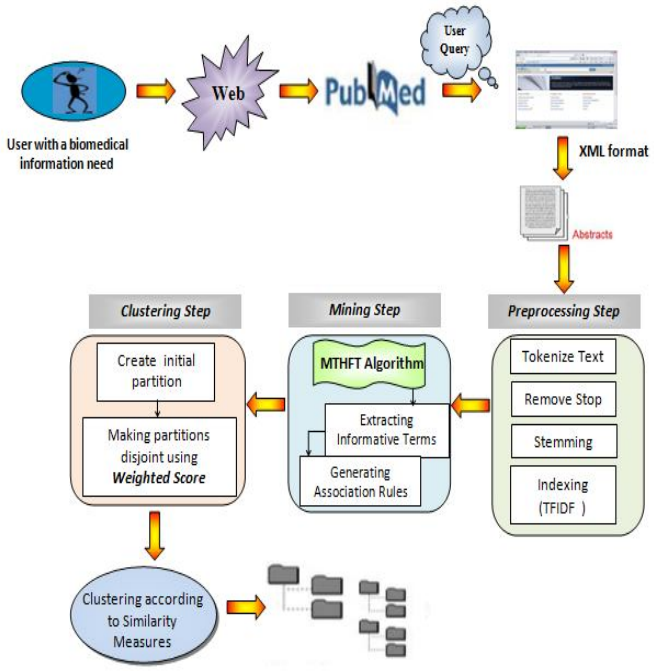


Figure 1: Architecture of KMDC System.

From the interface of the system, the user can submit his query to online PubMed search engine to retrieve up-to-date medical information, as shown in Figure 2. Furthermore, from the PubMed screen, the user selects the characteristics of the documents such as the type, the number, and the sort of documents. The retrieved XML documents from each query are automatically loading into the system interface. Each retrieved PubMed document comprises one abstract. In many cases, the abstract of a paper is not available in PubMed database, we remove these documents from the retrieved set. Once the online XML documents download into the system, their tags are automatically extracted in a combo box. After that the user can determine his specific part of the documents (for example the abstract part, </Abstract Text>) to work on it, as in Figure 3. Therefore the system is flexible to work on specific or all parts of documents.

3.2 Text Representation and Preprocessing step

The kind of linguistic features used in this paper to represent documents are single words. Single words are the structural units of language made up of one individual term. The most frequently used method to represent text is Vector-Space Model (VSM) since each of the unique word represents an axis in the vector space and each document is a vector in the space. The set of terms is defined as the set of all unique words or phrases occurring across all retrieved documents and no ordering of words or any structure of text is used. Let the size of the term set be N. The term set can be represent as $\{t_1, t_2, \ldots, t_N\}$[18].

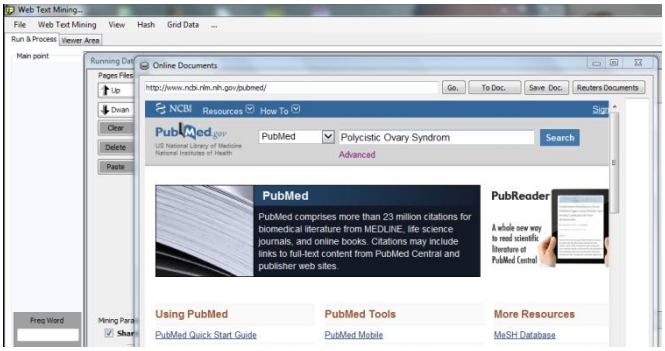


Figure 2: Submit Query to Online PubMed Search Engine from KMDC System.

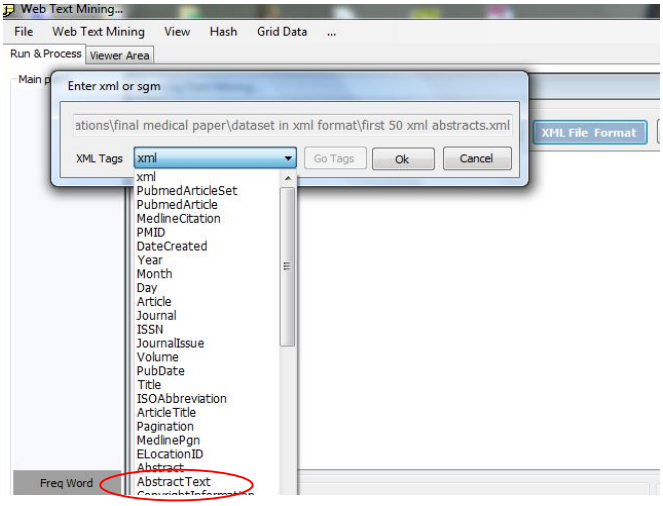


Figure 3: Determine the Specific Part of the XML Documents.

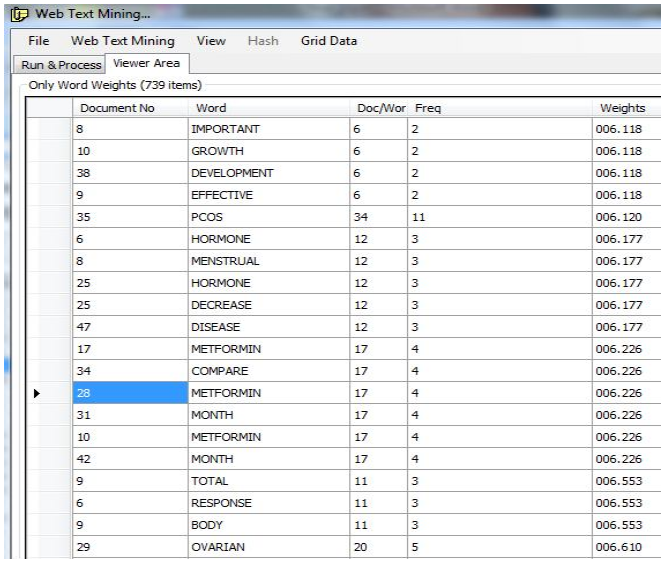
The preprocessing step is very complex and plays an important role in the subsequent clustering. In order to obtain all words that are used in retrieved documents, a tokenization process is required, i.e. a document is split into tokens (single words) or terms. After that, all unimportant words like articles, conjunctions, prepositions, etc. are removed from documents content that can affect the frequency count using stop words list. In addition, KMDC system replaces special characters, parentheses, commas, etc., with distance between words in the documents. We applied the Porter stemmer [19] to remove the prefix and the suffix of words. The effect of stemming is to reduce the number of distinct types in a text corpus and to increase the frequency of occurrence of some individual types. This tokenized representation is then used for further processing. The set of different words obtained by merging all text documents of a collection is called the dictionary of a document collection.

After the preprocessing step, each document d can be represent as a N-dimensional vector: $d = (w_1, w_2, \ldots, w_N)$,

where w_i is a weight representation of the significance of the term i in document d. The standard term frequency-inverse document frequency (TFIDF) function was used to assign weights to each word in each document from (1) [20].

$$w(i,j) = Nd_{i,t_j} * \log_2 \frac{|C|}{Nt_j} \quad (1)$$

where 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. Then each document was modeled as an N-dimensional TFIDF vector, where N is the number of distinct words in all of the abstracts. We introduce a weighted threshold value in order to obtain a reduced set of terms; this constrain also permits to reduce the computational time for large dataset. At the end of this stage, only top M of words are selected that satisfying the weighted threshold value, as in Figure 4.



Document No	Word	Doc/Wor	Freq	Weights
8	IMPORTANT	6	2	006.118
10	GROWTH	6	2	006.118
38	DEVELOPMENT	6	2	006.118
9	EFFECTIVE	6	2	006.118
35	PCOS	34	11	006.120
6	HORMONE	12	3	006.177
8	MENSTRUAL	12	3	006.177
25	HORMONE	12	3	006.177
25	DECREASE	12	3	006.177
47	DISEASE	12	3	006.177
17	METFORMIN	17	4	006.226
34	COMPARE	17	4	006.226
28	METFORMIN	17	4	006.226
31	MONTH	17	4	006.226
10	METFORMIN	17	4	006.226
42	MONTH	17	4	006.226
9	TOTAL	11	3	006.553
6	RESPONSE	11	3	006.553
9	BODY	11	3	006.553
29	OVARIAN	20	5	006.610

Figure 4: Only Weighted Words that Satisfying Weighted Threshold w = 80%

3.3 Mining Association Rules step

The purpose of generating association rules is to help the user who doesn't have any prior knowledge about the domain to gain the knowledge from the association between the commonly co-occurring terms of the domain. This knowledge helps them to understand about the domain and narrow down their search and retrieval. Moreover, association rules can be used to solve the problem of finding clusters of similar items. Figure 5 shows the flowchart of our MTHFT algorithm. In [21], MTHFT is implemented to extract all frequent informative terms that are presenting in more than one document and greater than the minimum threshold support furthermore to speed up the mining process. While in [22] we extended the algorithm to generate from all large frequent terms, all strong association rules that satisfies minimum confidence threshold. MTHFT algorithm has many advantages summarized in [21]. It is basically different from all the previous algorithms since it overcomes the drawbacks of Apriori algorithm by employing the power of data structure called Multi-Tire Hash Table. Moreover it uses new methodology for generating frequent termsets by building the hash table during the scanning of documents only one time. Consequently, the number of scanning on documents decreased. Once the frequent termsets from documents have been generated, it is straightforward to generate all strong association rules from them (where strong association rules satisfy both minimum support and minimum confidence), as in Figure 6. It permits the end user to change the threshold support and confidence factor without re-scanning the original documents to generate new association rules since the algorithm saves the hash table into secondary storage media.

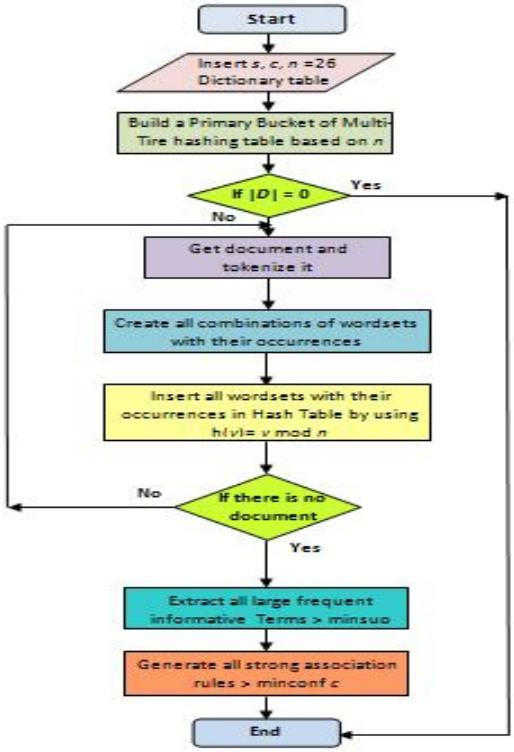
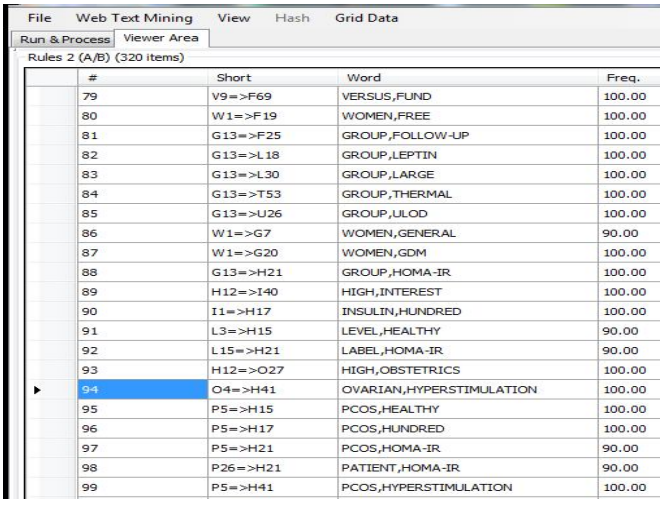


Figure 5: Flowchart of MTHFT Algorithm.



#	Short	Word	Freq.
79	V9=>F69	VERSUS,FUND	100.00
80	W1=>F19	WOMEN,FREE	100.00
81	G13=>F25	GROUP,FOLLOW-UP	100.00
82	G13=>L18	GROUP,LEPTIN	100.00
83	G13=>L30	GROUP,LARGE	100.00
84	G13=>T53	GROUP,THERMAL	100.00
85	G13=>U26	GROUP,ULOD	100.00
86	W1=>G7	WOMEN,GENERAL	90.00
87	W1=>G20	WOMEN,GDM	100.00
88	G13=>H21	GROUP,HOMA-IR	100.00
89	H12=>I40	HIGH,INTEREST	100.00
90	I1=>H17	INSULIN,HUNDRED	100.00
91	L3=>H15	LEVEL,HEALTHY	90.00
92	L15=>H21	LABEL,HOMA-IR	90.00
93	H12=>O27	HIGH,OBSTETRICS	100.00
94	O4=>H41	OVARIAN,HYPERSTIMULATION	100.00
95	P5=>H15	PCOS,HEALTHY	100.00
96	P5=>H17	PCOS,HUNDRED	100.00
97	P5=>H21	PCOS,HOMA-IR	90.00
98	P26=>H21	PATIENT,HOMA-IR	90.00
99	P5=>H41	PCOS,HYPERSTIMULATION	100.00

Figure 6: Snapshot of Generated Association Rules using MTHFT Algorithm at Minimum Confidence 90 %.

Since a number of association rules can be generated from each large termsets at each level which often results in a very large association rules. The minimum support and confidence threshold are critical factors for generating association rules. A low support threshold and high confidence threshold result in too many and more useful discovered associations. Increasing the support threshold significantly reduces the number of rules discovered, but risks losing useful associations.

3.4 Clustering PubMed abstracts step

The majority of the previous methods used the conceptual structure of NCBI's Medical Subject Headings (MeSH) terms for clustering PubMed abstracts and extracting topics; in our work we use the association rules as our information source to improve document clustering performance. The clustering process steps of PubMed abstracts are:

- Partition Initialization.
- Removing Partitions Overlapping
- Building Document Clustering by Similarity Measures

A. Partition Initialization

By using MTHFT algorithm, we can generate different sets of association rules as input to the clustering process easily. We start with a set of strong association rules R_s generated from the set of 2-large frequent termsets with high confidence threshold since $R_k = A_i \rightarrow B_j$. Initially, we sort the set of all strong association rules R_s in descending order in accordance with their confidence level, as in (2):

$$Conf(R_1: A_1 \rightarrow B_2) > Conf(R_2: A_2 \rightarrow B_4) > \ldots \ \ldots Conf(R_k: A_i \rightarrow B_j) \quad (2)$$

An initial partition P_1 is constructed for first association rule in R_s. Afterward, all the documents containing both termsets that constructed the rules are included in the same cluster. Next, we take the second association rules whose confidence is less than the previous one to form a new partition P_2. This partition is formed by the same way of the partition P_1. This procedure is repeated until every association rules moved into partition P_i. Finally we have a set of association rules and a set of all documents that contain the terms constructed the association rules, as in (3):

$$P_i = < R_i (A_i \rightarrow B_j), doc [R_i] > \quad (3)$$

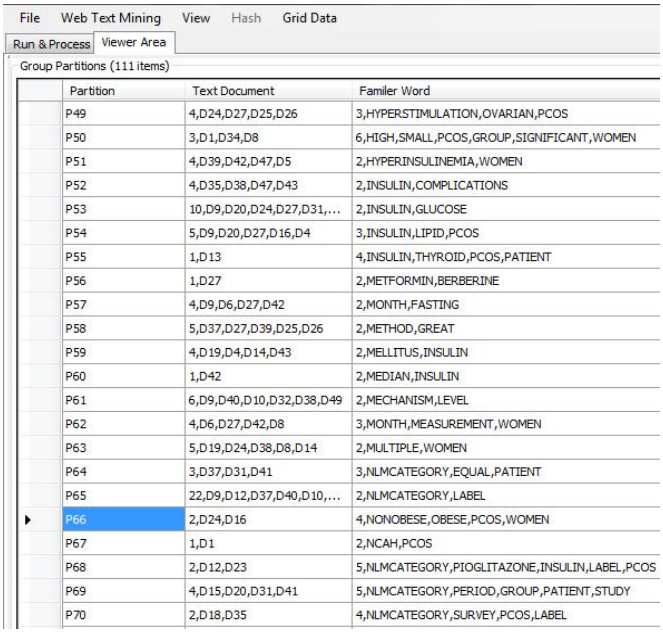
The purpose of constructing initial partitions is to ensure the property that all the documents in a cluster contain all the terms in the association rules that defines the partition. After that, all partitions that contain the similar documents are merged into one partition to reduce the number of resulted partitions, as shown in Figure 7. Since a document usually contains more than one frequent termset, the same document may appear in multiple initial partitions, i.e., initial partitions are overlapping.

B. Removing Partitions Overlapping

We found that there are some documents belong to one or more initial partitions so we attempt to remove the overlapping of partitions (*make partition disjoint*). We assign a document to the "Optimal" partition so that each document belongs to exactly one partition. This step also guarantees that every document in the partition still contains the mandatory identifiers. We propose in [22] the ***Weighted Score*** $(P_i \leftarrow doc_j)$ in equation (4) to measure the optimal initial partition P_i for a document doc_j.

$$(P_i \leftarrow doc_j) = \sum_k w_k * m_i / n_w \quad (4)$$

where $\sum_k w_k$ represents the sum of weighted values of all words constructed the association rules from doc_j, m_i represents the number of documents in the initial partition P_i, and n_w represents the number of words that construct the partition P_i from doc_j. The weighted values of words w_k are defined by the standard inverse document frequency (*TF-IDF*) in the indexing process.


File Web Text Mining View Hash Grid Data

Run & Process | Viewer Area

Group Partitions (111 items)

Partition	Text Document	Familer Word
P49	4,D24,D27,D25,D26	3,HYPERSTIMULATION,OVARIAN,PCOS
P50	3,D1,D34,D8	6,HIGH,SMALL,PCOS,GROUP,SIGNIFICANT,WOMEN
P51	4,D39,D42,D47,D5	2,HYPERINSULINEMIA,WOMEN
P52	4,D35,D38,D47,D43	2,INSULIN,COMPLICATIONS
P53	10,D9,D20,D24,D27,D31,...	2,INSULIN,GLUCOSE
P54	5,D9,D20,D27,D16,D4	3,INSULIN,LIPID,PCOS
P55	1,D13	4,INSULIN,THYROID,PCOS,PATIENT
P56	1,D27	2,METFORMIN,BERBERINE
P57	4,D9,D6,D27,D42	2,MONTH,FASTING
P58	5,D37,D27,D39,D25,D26	2,METHOD,GREAT
P59	4,D19,D4,D14,D43	2,MELLITUS,INSULIN
P60	1,D42	2,MEDIAN,INSULIN
P61	6,D9,D40,D10,D32,D38,D49	2,MECHANISM,LEVEL
P62	4,D6,D27,D42,D8	3,MONTH,MEASUREMENT,WOMEN
P63	5,D19,D24,D38,D8,D14	2,MULTIPLE,WOMEN
P64	3,D37,D31,D41	3,NLMCATEGORY,EQUAL,PATIENT
P65	22,D9,D12,D37,D40,D10,...	2,NLMCATEGORY,LABEL
P66	2,D24,D16	4,NONOBESE,OBESE,PCOS,WOMEN
P67	1,D1	2,NCAH,PCOS
P68	2,D12,D23	5,NLMCATEGORY,PIOGLITAZONE,INSULIN,LABEL,PCOS
P69	4,D15,D20,D31,D41	5,NLMCATEGORY,PERIOD,GROUP,PATIENT,STUDY
P70	2,D18,D35	4,NLMCATEGORY,SURVEY,PCOS,LABEL

Figure 7: Merging all Partitions that contain the Similar Documents.

The Weighted Score measure used the weighed values of frequent termsets instead of the number of occurrences of the terms in a document. Since the weighted values are an important piece of information based on the intuitive presumption of the weighting schema that is: the more often a term occurs in a document, the more representative of the content of the document (term frequency). Moreover the more documents the term occurs in, the less discriminating it is (inverse document frequency). To make partitions non-overlapping, we assign each doc_j to the initial partition P_i of the highest score$_i$. After this assignment, if there are more than one P_i that maximizes the Weighted Score $(P_i \leftarrow doc_j)$, we will choose the one that has the most number of words in the partition label. After this step, each document belongs to exactly one partition.

C. Building Document Clustering by Similarity Measures

In this step, we don't require to pre-specified number of clusters as previous standard clustering algorithms. In addition, we noticed that the number of clusters is independent to the number of documents. Once the initial

partitions are formed and removing the overlapping, we apply a certain similarity measure to merge similar documents and to yield non-overlapped clusters. There are many similarity measures such as Tanimoto [23], Cosine [24], and Correlation and Jaccard Coefficient [25]. We used Cosine method because it is the most common method to measure the similarity between two documents in the Vector-Space Model. Some of the reasons for the popularity of Cosine similarity are that it is very efficient to evaluate, especially for sparse vectors, and produce the highest cluster quality furthermore it works well when documents are viewed using Vector-Space Model as it explained in [26]. Given two documents, *A* and *B*, the cosine similarity, *Cos(θ)*, is represented using a dot product and magnitude, as in (5):

$$\text{Similarity} = Cos\ (\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}} \quad (5)$$

the Cosine similarity of two documents will range from 0 to 1, since the term frequencies (*tf-idf* weights) cannot be negative. The angle between two term frequency vectors cannot be greater than 90°. As shown in Figure 8, the resulting similarity ranges from 0 usually indicating independence, to 1 meaning exactly the same, and in-between values indicating intermediate similarity or dissimilarity [27]. Based on the similarity measure, a new cluster is formed from the partitions i.e. each cluster will contain all documents that have the similar similarity measures. Furthermore documents are merged if their similarity value is higher than a pre-defined threshold. Currently we use 0.4 as our threshold. We are experimenting with different ways of finding a better threshold.

For each cluster, we merge all association rules for each document contained in the cluster. These rules can be considered as the mandatory identifiers for every document in the cluster. We use these association rules as the topic to identify the cluster. The main purpose of presenting the topics is to facilitate browsing for the user.

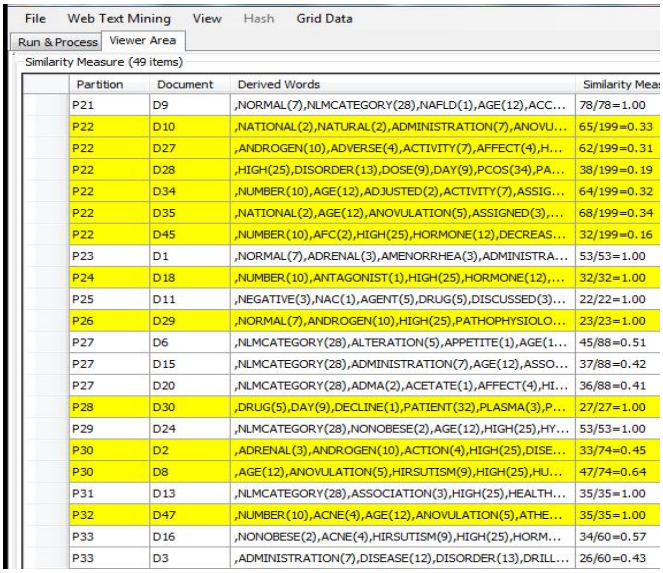
4. MEASURMENT ACCURACY OF THE SYSTEM

For evaluation the performance of the proposed system, the three evaluation metrics are used: Precision, Recall and F-measure. The Precision and Recall are defined here in terms of a set of retrieved terms of the domain from the PubMed abstracts and a set of relevant terms of the domain, as in (6) and (7):

$$Recall\ (K_i, C_j) = \frac{n_{ij}}{|K_i|} \quad (6)$$

$$Precision\ (K_i, C_j) = \frac{n_{ij}}{|C_j|} \quad (7)$$

The F-score measure considers both the precision and the recall to test the accuracy and it was computed using formula, as in (8):



Partition	Document	Derived Words	Similarity Mea
P21	D9	,NORMAL(7),NLMCATEGORY(28),NAFLD(1),AGE(12),ACC...	78/78=1.00
P22	D10	,NATIONAL(2),NATURAL(2),ADMINISTRATION(7),ANOVU...	65/199=0.33
P22	D27	,ANDROGEN(10),ADVERSE(4),ACTIVITY(7),AFFECT(4),H...	62/199=0.31
P22	D28	,HIGH(25),DISORDER(13),DOSE(9),DAY(9),PCOS(34),PA...	38/199=0.19
P22	D34	,NUMBER(10),AGE(12),ADJUSTED(2),ACTIVITY(7),ASSIG...	64/199=0.32
P22	D35	,NATIONAL(2),AGE(12),ANOVULATION(5),ASSIGNED(3),...	68/199=0.34
P22	D45	,NUMBER(10),AFC(2),HIGH(25),HORMONE(12),DECREAS...	32/199=0.16
P23	D1	,NORMAL(7),ADRENAL(3),AMENORRHEA(3),ADMINISTRA...	53/53=1.00
P24	D18	,NUMBER(10),ANTAGONIST(1),HIGH(25),HORMONE(12),...	32/32=1.00
P25	D11	,NEGATIVE(3),NAC(1),AGENT(5),DRUG(5),DISCUSSED(3)...	22/22=1.00
P26	D29	,NORMAL(7),ANDROGEN(10),HIGH(25),PATHOPHYSIOLO...	23/23=1.00
P27	D6	,NLMCATEGORY(28),ALTERATION(5),APPETITE(1),AGE(1...	45/88=0.51
P27	D15	,NLMCATEGORY(28),ADMINISTRATION(7),AGE(12),ASSO...	37/88=0.42
P27	D20	,NLMCATEGORY(28),ADMA(2),ACETATE(1),AFFECT(4),HI...	36/88=0.41
P28	D30	,DRUG(5),DAY(9),DECLINE(1),PATIENT(32),PLASMA(3),P...	27/27=1.00
P29	D24	,NLMCATEGORY(28),NONOBESE(2),AGE(12),HIGH(25),HY...	53/53=1.00
P30	D2	,ADRENAL(3),ANDROGEN(10),ACTION(4),HIGH(25),DISE...	33/74=0.45
P30	D8	,AGE(12),ANOVULATION(5),HIRSUTISM(9),HIGH(25),HU...	47/74=0.64
P31	D13	,NLMCATEGORY(28),ASSOCIATION(3),HIGH(25),HEALTH...	35/35=1.00
P32	D47	,NUMBER(10),ACNE(4),AGE(12),ANOVULATION(5),ATHE...	35/35=1.00
P33	D16	,NONOBESE(2),ACNE(4),HIRSUTISM(9),HIGH(25),HORM...	34/60=0.57
P33	D3	,ADMINISTRATION(7),DISEASE(12),DISORDER(13),DRILL...	26/60=0.43

Figure 8: Similarity Measure Value for each Document in Partitions.

$$F\ (K_i, C_j) = \frac{2 * Recall\ (K_i, C_j) * Precision\ (K_i, C_j)}{Recall\ (K_i, C_j) + Precision\ (K_i, C_j)} \quad (8)$$

where n_{ij} is the number of members of class K_i in cluster C_j. $|C_j|$ is the number of members of cluster C_j and $|K_i|$ is the number of members of class K_i. The weighted sum of all maximum F-measures for all natural classes is used to measure the quality of a clustering result C. This measure is called the overall F-measure of C, denoted $F(C)$ is calculated, as in (9):

$$F(C) = \sum_{K_i \in K} \frac{|K_i|}{|D|} max_{C_j \in C} \{F(K_i, C_j)\} \quad (9)$$

where *K* denotes all natural classes; *C* denotes all clusters at all levels; $|K_i|$ denotes the number of documents in natural class K_i; and $|D|$ denotes the total number of documents in the dataset. The range of $F(C)$ is [0,1]. A large $F(C)$ value indicates a higher accuracy of clustering.

5. EXPIREMENTS METHODOLOGYAND RESULTS

In our experiments, the experimental environment used is: CPU is 2.50 GHz Intel Core i5 processor, Memory is 6 GB RAM, Windows 7, and we chose the programming language C#.net for the implementation because it allows fast and flexible development. For experiment purpose, user query "PolyCystic Ovary Syndrome" (*PCOS*) was given as input to PubMed Search Engine. The number of documents retrieved was 11136 abstracts. After the online downloading of documents into KMDC system, it is noticed that there are some documents without abstract text. Therefore the number of documents will decrease to another actual number equals 10000 abstracts. The methodology of our experiments is as follows: first, we divided the actual number of abstracts into six document sets 100, 500, 1000, 2000, 5000 and 10000 documents, as shown in Table (1).

Table 1: Sample of Various Document Sets

# of online Documents	Size of documents	# of Weighted Terms
100	812 KB	4258
500	3.96 MB	17219
1000	7.92 MB	42998
2000	15.8 MB	65338
5000	39.6 MB	91923
10000	79.2 MB	134411

Second, since the minimum support has a critical role in the mining step, it must be properly chosen such that it is not too high where we may lose some interesting terms or too low where unimportant terms are generated. After that all strong association rules that are satisfying confidence threshold are generated. Third, applying KMDC system to cluster PubMed retrieved abstracts without need to prior knowledge about the number of clustering. Finally, investigate and evaluate the performance of KMDC system for clustering PubMed retrieved abstract in terms of the efficiency, accuracy and scalability through the evaluation measures and analyse the experiment results.

5.1 System Computational Efficiency and Scalability

Many experiments were conducted to exam the efficiency of KMDC system. All previous methods don't take into account improving the execution time during the experiments although the time is a critical factor in the clustering process especially with the text document. Using MTHFT algorithm in the mining step has significant impact for speeding up the mining and clustering steps. The experiments were performed twice at different two types of minimum support (*low and high*) to estimate the execution time at each one: first, we first chose two *high values of minimum support*; one equals to 10% for small size of documents (100 and 500) and the other equals to 3% for large size of documents (1000, 2000, 5000 and 10000). Furthermore we chose the threshold weight value *M*=80%. then compute the execution time for each step of KMDC system. Table (2) shows the computing time for each of the four phases of KMDC system operation.

Table 2: Computing Time in minutes for each step in KMDC system at high values of minimum support

# of document	Text Preprocess	Mining Step	Text Clustering	Total Time
100	~ 0.05	0.59	0.10	1.23
500	~ 0.09	1.26	0.35	2.10
1000	~ 0.15	2.47	1.49	4.51
2000	~ 0.21	5.35	2.21	10.19
5000	~0. 31	**11.08**	6.15	**17.54**
10000	~ 0.45	19.17	9.56	29.58

The first step in KMDC was online query submission and document retrieval. Since this step relies on PubMed to retrieve the abstracts, the time is not listed in Table (2). Table (2) shows the times to conduct the other three phases, text pre-processing, mining association rules and document clustering. From Table (2), we can see that the mining step took most of the time for example (11 minutes and 8 seconds out of a total 17 minutes and 54 seconds). This is due to using MTHFT algorithm since the time is consumed in building a hash table only one time.

Second, we chose two *small values of minimum support*; one equals to 1% for small size of documents (100 and 500) and the other equals to 0.2% for large size of documents (1000, 2000, 5000 and 10000) with the same threshold weight value M=80%. Then compute the execution time for each step of KMDC system. Table (3) shows the computing time for each of the three phases of KMDC system operation.

Table 3: Computing Time in minutes for each step in KMDC system at low values of minimum support

# of documents	Text Preprocess	Mining Step	Text Clustering	Total Time
100	~ 0.05	0.01	0.08	0.14
500	~ 0.09	0.02	0.15	0.26
1000	~ 0.15	0.28	1.03	1.46
2000	~ 0.21	0.39	2.07	3.07
5000	~0. 31	**0.45**	5.05	**6.21**
10000	~ 0.45	0.51	8.16	9.52

From Table (3), we can see that the execution time in the mining step decreased significantly to seconds (0.45 seconds) at 5000 documents although the minimum support is low value. The reason is due to saving the hash table into secondary media, we only begin selecting large frequent terms from the saved hash table. Consequently there is no time consuming in generating new association rules at different minimum support threshold (*small values*). The execution time is decreased to mine association rules as support decreased in compared to basic algorithms (Apriori algorithm). Improving the performance of mining process and decreasing the scanning and computational cost lead to increasing up the clustering process. As a result, KMDC system can cluster MEDLINE abstracts in a more efficient and faster manner.

To examine the scalability of KMDC system, we increase the size of downloaded documents from PubMed. Moreover, it is compared to the existing clustering algorithms like Bisecting K-means and FIHC. to ensure that the accuracy of all produced clustering are approximately the same, we use minimum support threshold 0.2% and confidence 80%. Figure 9 shows the scalability comparison of KMDC on different large sizes of PubMed abstracts. The number of documents is taken as X-axis and the time taken to find the clusters is taken as Y-axis. We concluded that KMDC system runs approximately twice faster than the two approaches FIHC and Bisecting K-means in this scaled up document set.

5.2 System Computational Accuracy

Many experiments were conducted to exam the accuracy of KMDC system for clustering PubMed retrieved abstract compared to Bisecting K-means and FIHC. The F-measure represents the clustering accuracy. To ensure fair comparison, we use two different minimum support threshold equals to 1% for small size of documents (100 and 500) and the other equals to 0.2% for large size of documents (1000, 2000, 5000

and 10000) and threshold weight value *M*=80%. In Figure 10, the experimental results show that KMDC system is more applicable to scientific related literature, since we obtained higher recall rate and F-measure while handling the search results of PubMed compared to the other algorithms. Higher F-measure shows the higher accuracy.

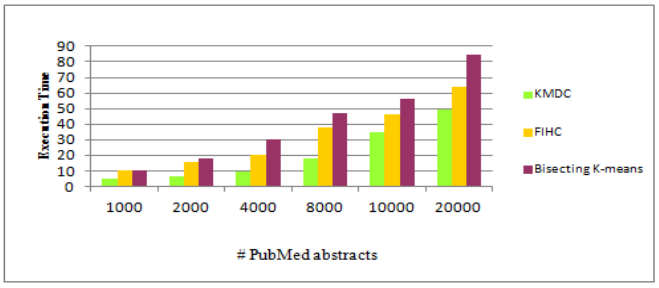


Figure 9: Scalability Comparison of ARWDC, FIHC and Bisecting K-means with Scale up Document Set.

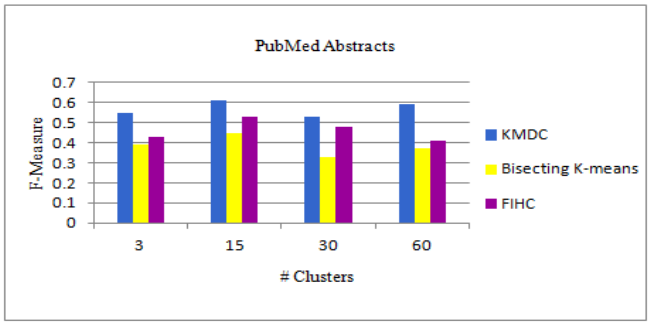


Figure 10: Accuracy Comparison with different number of Clusters for PubMed Retrieved Documents.

6. CONCLUSION

In this paper, we presented knowledge based medical documents clustering system using association rules mining. The proposed system combines domain knowledge with the features namely terms frequency, inverse document frequency and association rules in more effective way. The system showed an improvement over the existing systems with better results. From various evaluations carried out, the performance of the system found to be good comparatively to other systems in terms of F measures for clustering documents in biomedical domain. In future, we will focus on improving the efficiency and scalability of our system by using more efficient dimension reduction technique for removing less important words or by using semantic based clustering.

REFERENCES

1. D. M. Yandell and H. W. Majoros. **Genomics and natural language processing,** *Nature Reviews Genetics*, Vol. 3, no. 8, pp. 601-610, 2002.
2. http://www.ncbi.nlm.nih.gov/pubmed
3. F. H. Saad , B. D. Iglesia , and G. D. Bell. **Comparison of Two Document Clustering Approaches for Clustering Medical Documents,** in *Proc. DMIN Conf.*, 2006, pp. 425-431.
4. M. R. David and S. Samuel. **Clustering of PubMed abstracts using nearer terms of the domain,** *Bioinformation* Vol. 8, no. 2, pp. 020-025, 2012.
5. J. S. Nelson, et al. **Relationships in medical subject headings,** in *Proc. C.A. Bean & R. Green (Eds.), Relationship in the Organization of Knowledge Conf.*, New York: Kluwer Academic Publishers, 2001, pp. 171-184.
6. O. Bodenreider. **The Unified Medical Language System (UMLS): integrating biomedical terminology,** *Nucleic Acids Res.* 32(Database issue):D267-70, Jan. 2004.
7. L. Zhiyong. **PubMed and beyond: a survey of web tools for searching biomedical literature**, *Database*, vol. ED-2011, pp. 1-13, 2011.
8. R. N. Smalheiser, W. Zhou, and I. V. Torvik. **Anne O'Tate: a tool to support user-driven summarization**, *drill-down and browsing of PubMed search results. J. Biomed. Discov. Collab.*, Vol. 3, no. 2, 2008.
9. Y. Yamamoto, and T. Takagi. **Biomedical knowledge navigation by literature clustering,** *J. Biomed. Inform.*, Vol. 40, pp. 114–130, 2007.
10. A. Doms and M. Schroeder. **GoPubMed: exploring PubMed with the Gene Ontology,** *Nucleic Acids Res.*, Vol. 33, W783, 2005.
11. ClusterMed, 2004. http://demos.vivisimo.com/clustermed
12. C. Perez-Iratxeta. **XplorMed: a tool for exploring MEDLINE abstracts,** *Trends Biochem. Sci.*, Vol. 26, pp. 573–575, 2001.
13. R. Yeh, et al. **Semantic Based Real-Time Clustering for PubMed Literatures,** in *Proc. 10th Discovery Science Int. Conf.*, Sendai, Japan, October 1-4, 2007, pp. 291-295.
14. O. E. Zamir. **Clustering Web Documents: A Phrase-Based Method for Grouping Search Engine Results**, Ph.D. dissertation, Dept. Comp. Sc. And Eng., Washington Univ., USA, 1999.
15. D. A. Smith. **Detecting and browsing events in unstructured text,** in *Proc. 25th Annu. Int. ACM SIGIR Conf. Res. and Develop. Inf. Retrieval*, 2002, pp. 73–80.
16. S. Osinski, and D. Weiss. **A Concept-Driven Algorithm for Clustering Search Results,** *IEEE Intelligent Systems*, Vol. 20, no. 3, pp. 48–54, 2005.
17. S. Osinski and D. Weiss. **Conceptual Clustering Using Lingo Algorithm: Evaluation on Open Directory Project Data,** in *Proc. Intelligent Information Processing and Web Mining Conf.*, Zakopane, Poland, Springer Physica-Verlag, 2004, pp. 369—378.
18. P. D. Turney, and P. Pantel. **From Frequency to Meaning: Vector Space Models of Semantics,** *Journal of Artificial Intelligence Research*, Vol. 37 pp. 141-188, Oct. 2010.
19. http://tartarus.org/martin/PorterStemmer
20. M. Berry. **Survey of text mining: clustering, classification, and retrieval;** New York: Springer-Verlag, 2004, ch.3.
21. N. Negm, P. Elkafrawy, M. Amin, and A. M. Salem. **Clustering web documents based on efficient multi-tire hashing algorithm for mining frequent termsets,** *Int. J. of Advanced Research in Artificial Intelligence*, Vol. 2, no. 6, pp. 6-14, 2013.

22. N. Negm, P. Elkafrawy, M. Amin, and A. M. Salem. **Investigate the Performance of Document Clustering Approach Based on Association Rules Mining,** *Int. J. of Advanced Computer Science and Applications,* Vol. 4, no. 8, pp. 142-151, 2013.
23. K. Lin, and R. Kondadadi. **A Similarity-Based Soft Clustering Algorithm for Documents**, in *Proc. Database System for Advanced Applications Conf.*, 2001, pp.40-47.
24. U. Yong, J. R. Mooney. **Text Mining with Information Extraction,** in *AAAI Symposium on Mining Answers from Texts and Knowledge Bases*, Stanford, CA, 2002.
25. Y. Zhao, G. Karypis. **Evaluation of Hierarchical Clustering Algorithms for Document Datasets**, in *Proc. Information and knowledge management Int. Conf.*, Virginia, USA, pp. 515-524, 2002.
26. Y. Zhao, G. Karypis. **Comparison of Agglomerative and Partitional Document Clustering Algorithms**, *The SIAM workshop on Clustering High-dimensional Data and Its Applications*, Washington, DC, April 2002.
27. http://en.wikipedia.org/wiki/Cosine_similarity