ISSN:2231-0711
www.ijcset.net

Candidate Cluster Extraction for Hierarchical Document Clustering

Leena H. Patil
Department of Computer Sci. & Engg
Priyadarshini Institute of Engineering and Technology, Nagpur, India
Email-id: harshleena23@rediffmail.com

Dr. Mohammad Atique
Department of Computer Sci. & Engg
Sant Gadge Baba Amravati University Amravati, India
Email-id: atique_shaikh@rediffmail.com

Abstract: **Text Document are tremendously increasing in the internet, the hierarchical document clustering has proven to be useful in grouping similar document for large applications. Still most documents suffer from problems of high dimensionality, scalability, accuracy and meaningful cluster labels. In this paper an new approach fuzzy frequent itemsets based hierarchical clustering is proposed, in which fuzzy association rule mining algorithm is used to improve the clustering accuracy. In this approach firstly the key terms are extracted from the document set and each document is preprocessed into the document representation for the further mining process. Secondly, a fuzzy association rule mining algorithm for text is discover to find the sets, highly related fuzzy frequent itemsets, in which the key terms are regarded as a labels of the candidate clusters. Referring the candidate cluster it has been experimentally evaluated based on classic 30, Classic 4, and tr11 data sets for two methods FIHC and K-means in MATLAB 2009Rb.**

Keywords- Introduction, document clustering approach, document preprocessing, candidate cluster extraction, F-measure evaluation.

I. INTRODUCTION

To browse and organize documents smoothly, hierarchical clustering techniques have proposed to cluster a collection of documents into hierarchical tree structure. Still there exist several challenges such as high dimensionality, scalability, accuracy and meaningful cluster [1].According to data mining, text mining is more complex because text data are inherently unstructured and fuzzy proposed a method namely mining term association to acquire the semantic relation between terms when applying to documents. Moreover [2] shown that the association rule mining is the first data mining technique employed in mining text collection. These focuses on analyzing the co-occurrence terms for document management. Further [3] proposed a method namely FIHC to produce hierarchical topic tree for document clustering. It resolve the challenges like dimensionality, reduction, scalability, accuracy and easy browsing with meaningful cluster labels. In this tf-idf(term frequency –inverse document frequency method is used to replace the actual frequency of a term by the weighted frequency. The main limitation of tf-idf method is that long documents tend to have higher weights than short ones.

In this approach we propose an integrating fuzzy set concept and the association rule mining to find interesting fuzzy association rules from given transactions. Using this [2] algorithm it is easily understandable and realistic for integrating linguistic terms with fuzzy sets. In the first stage the key terms are extracted from the document set and each document is preprocessed into the designated representation. In this a hybrid feature selection method is used to effectively reduce the unimportant terms for each document. Secondly a set of relevant fuzzy frequent itemsets are used to discover efficiently, where FARM algorithm is proposed for text. A frequent itemset defined [3][4], is a set of words that occur together in some minimum fraction documents in a cluster by employing predefined membership functions. In this the three fuzzy values are calculated in MATLAB i.e Low,Mid and High regions for each term based on its frequency. The derived frequent itemsets contains key terms to be regarded as candidate cluster labels and shown experimentally.

II. DOCUMENT CLUSTERING

A. Hierarchical Document Clustering

Most clustering algorithm are divided into partitioning methods and hierarchical methods, in partitioning methods, documents are exclusively partition the set of documents into a number of clusters by moving documents from one cluster to another. Commonly K-methods has been used . the purpose of hierarchical document clustering is to build a hierarchical tree of clusters whose leaf nodes represent the subset of a document collection. Besides, frequent itemsets have been applied to document clustering extensively. For example [1] proposed the HFTC method that by using frequent itemsets and minimizing the overlap of clusters in terms of shared documents. But experimentally [4] proved that HFTC is not scalable. To be scalable [4] proposed a new novel FIHC algorithm by using frequent itemsets a hierarchical topic tree for cluster can be constructed. It also proved that by using frequent itemsets it can reduce the dimension of a vector space effectively. Therefore it not only reduces dimensionality, but also offer efficient processing of high volume data, support ease of browsing and provides meaningful cluster labels. In the following we will present our approach from frequent itemset based clustering technique and implement the algorithm proposed by [8] to text processing in MATLAB to find suitable F-measure values and fuzzy frequent itemset.

III. FRAME WORK APPROACH

Our framework approach is divided into three stages, which are as follow:

1. Document Preprocessing: In this the frequency of each term within the document is counted.
2. Candidate cluster: In this FARM algorithm is used to find fuzzy frequent itemsets, which are then used to form the candidate cluster.
3. F-measure Evaluation.

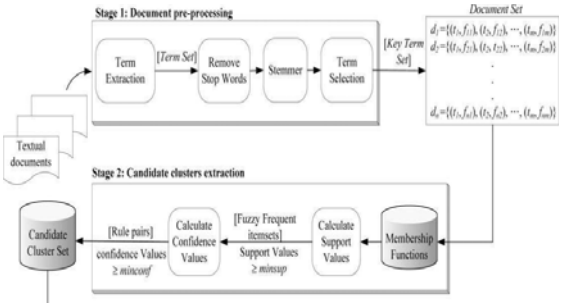


Fig1. Frame work approach

A. Document Preprocessing

The purpose of this stage is to reduce the dimensionality for high clustering accuracy[2][4], as there are thousands of document sets. Several methods like item pruning, feature clustering, feature selection technique and matrix factorization are been applied to reduce dimensionality. To solve his problem we find the terms that are significant and important to represent the content of each document. For increasing the clustering accuracy and maintaining the computing cost small we first remove the terms that are not meaningful and discriminative. The details of preprocessing are as follow:

1. Divide the sentences into terms.
2. Remove the stop word list.
3. Conduct word stemming.
4. Term selection: the term with selection metric weights all higher than the prespecified thresholds will be selected as key terms. In this three feature selection methods are defined : tf-idf, tf-df, and tf^2 are used to select representative terms for each document and these feature selection methods are defined as follow:

1. tf-idf : term frequent x inverse document frequency used for measure of the important term t_j within document d_i .
2. tf-df : term frequent x document frequency is evaluated to calculate the values by dividing the term frequency by the document frequency.
3. tf^2: it is the multiplication of $tfidf_{ij}$ and $tfdf_{ij}$.

After these weights of each term in each document have been calculated, those which have weights all higher than prespecified thresholds are retained. Subsequently, these retained terms form a set of key terms for the document set D and all are defined as follow:

Definition 1: A document di= {(t1,fi1),(t2,fi2)….(tm,fim)}, is a logical unit of text, characterized by a st of key terms tj together with their corresponding frequency fij.

Definition 2: A document set D= {d1,d2,d3…di,…dn} also called a document collection, is a set of documents where n is the total number of documents in D.

Definition 3: The term set of a document set D={d1,d2,d3…di,..dn} denoted T_D={t1,t2,t3,…tj,…ts}, is the set of terms appeared in D, where s is the total number of terms.

Definition 4: The key term set of a document set D={d1,d2,d3,…di,…dn} denoted KD= {t1,t2,t3,..tj,..tm},is a subset of the term set TD, including only meaningful key terms, which do not appear in a well defined stop word list, and satisfy the pre defined minimum tf-idf threshold α, the minimum tf-df threshold β and the minimum tf^2 threshold γ.

Based on the above definitions the representation of a document can be derived by algorithm 1. Shown in figure 1.1 .

Algorithm 1. Document preprocessing algorithm

Input:

1. A document set D={d1,d2,d3….di,..dn}
2. A well defines stop word list.
3. The minimum tf-idf threshold α.
4. The minimum tf-df threshold β.
5. The minimum tf^2 threshold γ.

Output: The key term set of D, K_D

Method:

Step 1: Extract the term set T_D={t1,t2,t3,…tj,..ts}

Step 2. Remove all stop words from TD.

Step 3. Apply word stemming for TD

Step 4. For each d_i Є D do

For each t_j Є T_D do

1. Evaluate its $tfidf_{ij}$, $tfdf_{ij}$ anf tf^2 weights.
2. Retain the term if $tfidf_{ij}$ >= α, $tfdf_{ij}$ >= β and tf^2_{ij} >= γ.

Step 5. Obtain the key term set KD based on the previous steps.

Step 6. For each di Є D do

For each tj Є K_D do

1. Count its frequency in di to obtain di={(t1,f_{i1}),(t2,f_{i2}),….(tj,f_{ij}),..(tm,f_{im})}.

---

let us consider one example a document set D={d1,d2,d3,….d10} containing 10 documents. By algorithm 1, we might obtain the derived representation of D and its key term set KD (stock, record, profit, medical, treatment, health}as shown below:

Table 1 Document set.

ID	Key term set					
	Stock	Record	Profit	Medical	Treatment	Health
d1	2	1	1	0	0	0
d2	1	1	0	0	0	0
d3	1	0	2	0	0	0
d4	0	0	0	3	0	2
d5	0	0	0	11	1	1
d6	0	1	0	4	0	0
d7	0	0	0	8	1	2
d8	3	0	1	0	0	0
d9	0	1	0	3	0	0
d10	0	0	0	8	2	1

B. Candidate cluster extraction:

The objective of candidate cluster extraction is to take document set D, a set of predefined membership functions, the minimum support value θ, and the minimum confidence value as input and to output a set of candidate clusters. To achieve this goal we modified the algorithm proposed by [2]. to capture the relationship among different key terms of the document

set in MATLAB. Since each discovered fuzzy frequent itemset has an associated fuzzy count value, it can be regarded as the degree of importance that the itemset contributes to the document set. In this we define the membership function[8] and explain it with example.

1. Membership function: it is used to convert each term frequency into a fuzzy set. Therefore, we define the t-f (term frequency) fuzzy set used in this paper.

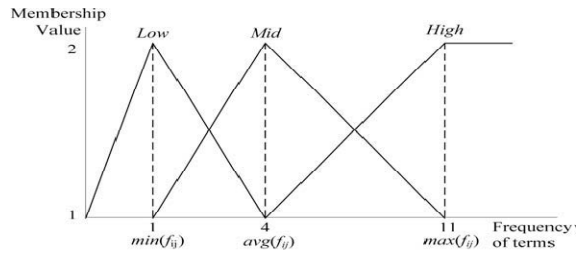
Definition 1. A t-f fuzzy set of document d_i is a pair (F_{ij}, W^r_{ij}), where Fij is a set and equals to $\{w_{ij}^{low}(F_{ij})/ t_j.\text{Low}, w^{mid}_{ij}(f_{ij})/tj.\text{Mid}, w^{High}_{ij}(f_{ij})/t_j.\text{High}\}$, w^r_{ij}:F→[0,2], and r can be Low, Mid,or High. The notation tj. r is called a fuzzy region of t_j. For each term pair (t_j, f_{ij}) of document d_i, $w^r_{ij}(f_{ij})$ is the grade of membership of tj in di with Low, Mid and High membership functions defined by formulas respectively.

$$w_{ij}^{low}(f_{ij})=\begin{cases}0, fij=0\\ 1+\frac{fij}{a1}, \quad 0<fij, a1\\ 2, fij=a1 \quad , a1=\min(fij), a2=avg(fij)\\ 1+a2-\frac{fij}{a2}-a1, \quad a1<fij<a2\\ 1, \quad fij\geq a2\end{cases}$$

$$w^{mid}_{ij}(f_{ij})=\begin{cases}0, fij=0\\ 1, fij=a1\\ 1+fij-\frac{a1}{a2}-a1, a1<fij<a2\\ 2, fij=a2, \ a1=\min(fij), a2=avg(fij), a3=\max(fij)\\ 1+a3-\frac{fij}{a3}-a2, a2<fij<a3\\ 1, \quad fij=a3\end{cases}$$

$$w^{High}_{ij}(f_{ij})=\begin{cases}0, \quad fij=0\\ 1, fij\leq a1, \quad a1=avg(fij), a2=\max(fij)\\ 1+\frac{fij}{a2}-a1, a1<fij<a2\\ 2, \quad fij=a2\end{cases}$$

where min(fij) is the minimum frequency of terms in D, max(fij) is the maximum frequency of terms in D, and avg(fij)= $\left\lceil \frac{\sum_{j-1}^{n} fij}{|k|} \right\rceil$, where fij ≠ min(fij) or max(fij) and |K| is the number of summed key terms. Based on document set, the derived membership function is shown in fig 3 below.



2. The fuzzy association rule mining algorithm for text

Following are definitions to describe the fuzzy association rule mining algorithm for text.

Definition 1. For a document set D, a candidate cluster c= (Dc, τ) is a two tuple, where Dc is a subset of the document set D, such that it includes those documents which contain all the key terms in τ={t1,t2,t3…tq} K_D, q =1, where K_D is the key term set of D and q is the number of key terms included in τ, where τ is a fuzzy frequent itemset for describing c. for example from table 1, the candidate cluster $c_{(stock)}$ =({ d1,d2,d3,d8},{stock}), as the term "stock" appeared in these documents.

Definition 2. The candidate cluster set of a document set D, denoted C_D = {c1,…cl-1,cl,….ck} is a set of candidate clusters, where k is the total number of candidate clusters. The candidate cluster set CD for a document set D can be generated by algorithm 2. Shown in figure below.

Algorithm 3.2. The fuzzy association rule mining algorithm for finding fuzzy frequent itemsets and regarded them as a candidate cluster set for output.

Input: 1. A document set $D = \{d_1, d_2, \ldots, d_n\}$, where $d_i = \{(t_1, f_{i1}), (t_2, f_{i2}), \ldots, (t_j, f_{ij}), \ldots, (t_m, f_{im})\}$;

2. A set of membership functions (as defined in Section 3.2.1);

3. The minimum support value θ;

4. The minimum confidence value λ.

Output: Generate a candidate cluster set $\tilde{C}_D$.

Method:

Step 1. For each key term, using the given membership functions to transform each term frequency f_{ij} into a *fuzzy set*

$$F_{ij} = w_{ij}^{Low}/t_j.Low + w_{ij}^{Mid}/t_j.Mid + w_{ij}^{High}/t_j.High,$$

Step 2. For D, calculate the scalar cardinalities of three fuzzy regions for each key term $t_j \in K_D$. That is,

$$count_j^{Low} = \sum_{i=1}^{n} w_{ij}^{Low}, \ count_j^{Mid} = \sum_{i=1}^{n} w_{ij}^{Mid}, \text{ and } count_j^{High} = \sum_{i=1}^{n} w_{ij}^{High}$$

Step 3. Find the region of each key term with maximum count. That is,

$max\text{-}count_j = max(count_j^{Low}, count_j^{Mid}, count_j^{High})$, for $j = 1$ to m, where m is the total number of key terms in K_D. Let $max\text{-}R_j$ be the region with $max\text{-}count_j$ for a key term t_j, which will be used to represent the fuzzy weight of t_j in the following steps.

Step 4. Find fuzzy frequent 1-itemsets L_1.

This step checks the support value of a key term t_j, denoted $support(t_j) = \frac{max\text{-}count_j}{|D|}$, $1 \leq j \leq m$. If it is larger than or equal to θ, where $|D|$ is the number of documents in D and $max\text{-}count_j$ is the count value of a region $max\text{-}R_j$, then the fuzzy frequent itemset will be put into the large 1-itemset L_1. That is,

$$L_1 = \{max\text{-}R_j \mid support(t_j) \geq \theta, \ 1 \leq j \leq m\}.$$

Step 5. Generate the candidate set C_2 from L_1.

Step 6. Find fuzzy frequent 2-itemsets L_2.

For each candidate 2-itemset τ with key terms (t_1, t_2) in C_2, there are three sub-steps to be performed:

6-1. Generate the fuzzy value of τ, $w_{i\tau} = min(w_{i1}^{max\text{-}R_1}, w_{i2}^{max\text{-}R_2})$, in each document d_i, where $w_{ij}^{max\text{-}R_j}$ is the fuzzy value of the maximum region of a key term t_j in d_i.

6-2. Calculate the scalar cardinality of τ in D as $count_\tau = \sum_{i=1}^{n} w_{i\tau}$.

6-3. Put τ in L_2 if $support(\tau) = \frac{count_\tau}{|D|}$ is larger than or equal to θ.

Step 7. Check whether L_2 is null. If L_2 is null, then exit the algorithm; otherwise, do the next step.

Step 8. Set $q = 2$, where q represents the number of key terms stored in the current fuzzy frequent q-itemsets.

Step 9. Generate the candidate set C_{q+1} from L_q.

This step is similar to the *a priori* algorithm (de Campos & Moral, 1993). First, our algorithm also joins L_q and L_q to assume that q - 1 key terms in the two itemsets are the same and the other one is different. Then, C_{q+1} itemsets that have all their sub-q-itemsets in L_q are generated.

Step 10. Find fuzzy frequent $(q + 1)$-itemsets L_{q+1}.

For each candidate $(q + 1)$-itemsets τ with key terms $(t_1, t_2, \ldots, t_{q+1})$ in C_{q+1}, there are three sub-steps to be achieved:

10-1. Generate the fuzzy value of τ, $w_{i\tau} = min\{w_{ij}^{max\text{-}R_j} \mid j = 1, 2, \ldots, q+1\}$, in each document d_i, where is the fuzzy value of the maximum region of a key term t_j in d_i.

10-2. Calculate the scalar cardinality of τ in D as $count_\tau = \sum_{i=1}^{n} w_{i\tau}$.

10-3. Put τ in L_{q+1}, if $support(\tau) = \frac{count_\tau}{|D|}$ is larger than or equal to θ.

Step 11. Check whether L_{q+1} is null or not.

If L_{q+1} is null, then exit the algorithm; otherwise, set $q = q + 1$ and repeat Steps 9-11.

Step 12. Construct the association rules.

For all the fuzzy frequent q-itemsets τ containing key terms $(t_1, t_2, \ldots, t_q)$, where $q \geq 2$. There are sub-steps to be accomplished:

12-1. Form all possible association rules. That is,

$$\tau_1 \wedge \ldots \wedge \tau_{k-1} \wedge \tau_{k+1} \wedge \ldots \wedge \tau_q \rightarrow \tau_k, \ k = 1 \text{ to } q.$$

12-2. Calculate the confidence values of all association rules using the formula:

$$\frac{\sum_{i=1}^{n} w_{i\tau}}{\sum_{i=1}^{n} (w_{i1} \wedge \ldots \wedge w_{ik-1}, w_{ik+1} \wedge \ldots \wedge w_{iq})}$$

Step 13. Hold those rules which the confidence values of the rule pair are all larger than or equal to λ.

Step 14. Output the fuzzy frequent 1-itemsets and fuzzy frequent q-itemsets derived in Step 13 to form the candidate cluster set $\tilde{C}_D$.

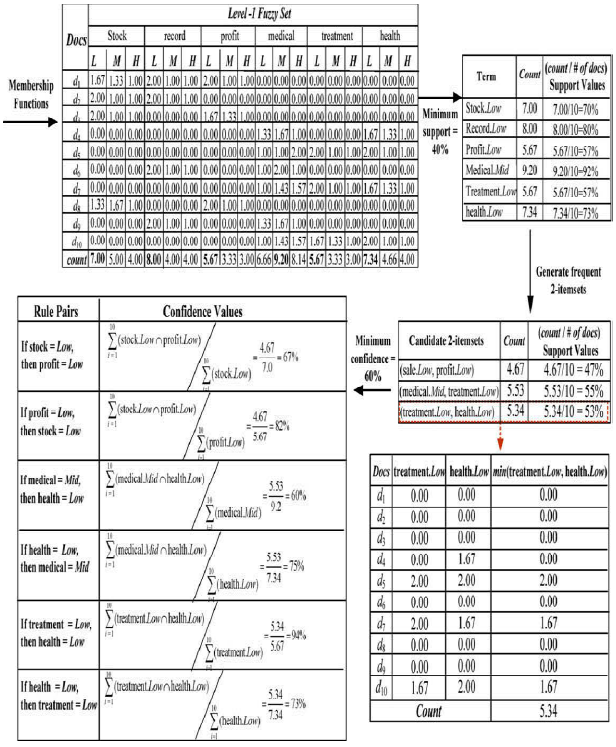
3. An illustrative example:

Consider using the document set D in table. 1 , the membership functions defined in fig 3, the minimum support value 40% and the minimum confidence value 60 % as inputs. The fuzzy frequent itemsets discovery procedure is illustrated in fig below.

In the proposed algorithm[6][7][8], we estimate the strength of association among key terms in the document set by using confidence values. There is useful information when co-occurring keywords have been shown. This is because highly co-occurring terms are used together. Thus the algorithm computes the confidence values of a rule pair to check the strong association of key terms (t1,t2,t3,….tq) of the fuzzy frequent q-item sets. Take the candidate cluster $c_{(stock,profit)}$ as an example .Since its confidence value of the rule pair "If stock=low then profit =Low" and "if profit=low then stock =low" are both larger than the minimum confidence value 60%, $c_{(stock,\ profit)}$ is put in the candidate cluster set C_D. Finally the candidate cluster set

C_D={c1(stock),c1(record),c1(profit),c1(medical),c1(treatment), c1(health), c(stock,profit), c(medical, health), c(treatment, health)} will be output.

Fig 5



C. Experimental Results

In this we experimentally evaluate the performance of the proposed algorithm by comparing with FIHC method. All the experiments have been performed on P4 3.2 ghz machine.

1. Datasets

The three standard datasets employed by the FIHC experiments. These datasets are widely adopted as standard benchmark for the text categorization task. To find key terms, stop words, were removed and stemming was performed. Documents then were represented as TF(term frequency) vectors, unimportant terms were discarded. This process implies a significant dimensionality reduction without loss of clustering performance. The statistics of these datasets, after the document preprocessing are summarized below.

Table 3.
Statistics for our test datasets.

Datasets	No.of Documents	No. of Natural clusters	Class size			The length of documents
	Total	Total	Max	Avg	Min	Average
Classic 4	7094	4	3203	1774	1033	43
Tr11	414	9	032	46	6	46

2. Evaluation of cluster quality : overall F-measures

The F-measures is often employed to evaluate the accuracy of the generated clustering results. It is a standard evaluation method for both flat and hierarchical clustering structure. More importantly, this measure balances the cluster precision and cluster recall. Hence we define a set of document clusters generated from the clustering result, denoted C and another set denoted L, consisting of natural classes, such as each document is pre-classified into a single class. Both sets are derived from the same document set D.

This measure is called the *overall F-measure* of C denoted F(C) and is defined as follow.

$F(C) = \sum_{lj \in L} \frac{|lj|}{|D|} \max_{ci \epsilon C}\{F\}$ where $F = \frac{2PR}{P+R}$, $P = \frac{|c\,i \cap lj|}{|ci|}$

and $R = \frac{|c\,i \cap lj|}{|lj|}$

in general, the higher the F(C) values, the better the clustering solution is.

3. The effect of feature selection

In document clustering, feature selection is essential to make the clustering task efficient and more accurate. The most important goal of feature selection is to extract topic related terms, which could present the content of each document.

4. Evaluation results

Experiments are conducted to compare the accuracy of the algorithm with other methods and further evaluate the accuracy of the algorithm with respect to minimum support ranging from 2% to 9 %. Also the efficiency of the algorithm is measured and shown in section below.

1. Accuracy comparison

Overall f-measures values for FIHC, k-means, bisecting k-means algorithm compared with four different numbers of clusters 3, 6, 10, 15. We use the same minimum support, ranging from 3 % to 6 % in each data set.

Table 6. Comparison of the overall f-measures.

Dataset	No. of clusters	FIHC	K-means
Classic 4	3	0.666667	0.125
	6	0.571429	0.181818
	10	0.666667	0.285714
	15	0.857143	0.5
TR11	3	0.452226	0.25896
	6	0.358962	0.15824
	10	0.452589	0.24875
	15	0.589452	0.3

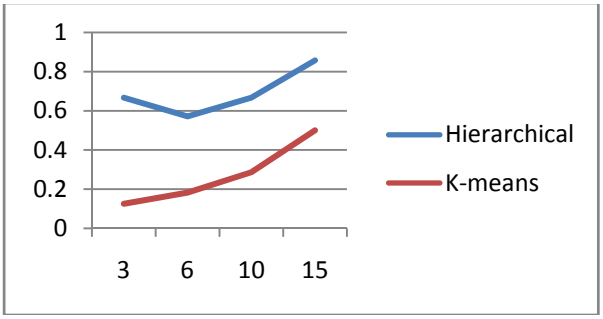


Fig 6. For classic 30 data set

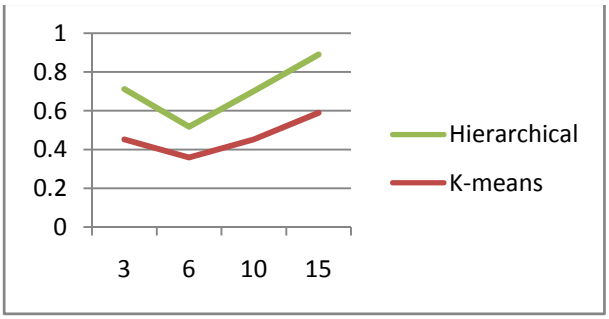


Fig7. For tr11 data set

5. *Conclusion*

Although numerous document clustering methods have been extensively studied for many years, the high Computation complexity and space need still make the clustering methods inefficient. Hence, reducing the heavy computational load and increasing the precision of the unsupervised clustering of documents are important issues. Therefore, we derived a fuzzy-based hierarchical document clustering approach, based on the fuzzy association rule mining, for alleviating these problems satisfactorily. In our approach, we start with the document pre-processing stage; then employ by using the fuzzy association data mining method in second stage; which generate a candidate cluster set, and merge the high similar clusters. Our experiments show that the accuracy of our algorithm which has implemented in matlab is higher than that of FIHC method, UPGMA, and Bisecting k-means when compared on the three standard datasets. Moreover, the experiment results show that the use of FARM discovery important candidate clusters for document clustering to increase the accuracy quality of document clustering. Therefore, it is worthy extending in reality for concentrating on huge text documents management.

Our future work focuses on the following two aspects:
1. For improving the performance of the document clustering algorithm, the soft computing approach i.e rough set approach can be applied. Further the algorithm can be improved for higher accuracy by using domain knowledge like WordNet
2. An efficient incremental clustering algorithm can be applied for assigning new document to the most similar existing cluster be proposed as the future direction.

REFERENCES

[1] Beil, F., Ester, M., & Xu, X. "*Frequent term-based text clustering"*. In Proceedings of the 8th ACM SIGKDD int'l conf. on knowledge discovery and data mining(pp. 436–442), 2002.

[2] Chen, C. L., Tseng, Frank S. C., & Liang, T. "*Hierarchical document clustering using fuzzy association rule mining"*. In Proceedings of the 3rd international Conference of innovative computing information and control (ICICIC2008) (pp. 326–330), 2008.

[3] Fung, B. C. M., Wang, K., & Ester, M. "*Hierarchical document clustering using frequent itemsets*". Master thesis, Simon Fraser University, 2002.

[4] Fung, B. C. M., Wang, K., & Ester, M. "*Hierarchical document clustering using frequent itemsets"*. In Proceedings of the 3th SIAM int'l conf. on data mining(pp. 59–70), 2003.

[5] Shahnaz, F., Berry, M. W., Pauca, V. P., & Plemmons, R. J. "*Document clustering using nonnegative matrix factorization"*. Information Processing and Management, 42(2), 373–386, 2006.

[6] Shihab, K. "*Improving clustering performance by using feature selection and extraction techniques"*. Journal of Intelligent Systems, 13(3), 135–161, 2004.

[7] Steinbach, M., Karypis, G., & Kumar, V. "*A comparison of document clustering techniques*". In Proceedings of the KDD workshop on text mining, 2000.

[8] Hong, T. P., Lin, K. Y., & Wang, S. L. "*Fuzzy data mining for interesting generalized association rules*". Fuzzy Sets and Systems, 138(2), 255–269,2003.