International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 10 (2015) pp. 26175-26190

http://www.ripublication.com

Feature Selection based on Term Frequency and Term Document Frequency for Text Clustering

Sivaram Prasad Nalluri[1] and Rajasekhara Rao Kurra[2]

[1]Department of Information Technology, Bapatla Engineering College, Bapatla 522102, Andhra Pradesh, India, becithod@gmail.com.
[2]Department of Computer Science and Engineering, Sri Prakash College of Engineering, Andhra Pradesh, India, krr_it@yahoo.co.in.

Abstract

A document when represented by redundant and noisy terms along with significant terms, as in popular Vector Space Model, leads to a poor clustering outcome. A term, used for document representation, is redundant if it is not useful to distinguish a document from other documents. Similarly a term is noisy, if it does not contribute anything to the semantics of the document in which the term is present. This paper proposes a novel unsupervised filter method for feature selection. Filter methods assign weights to terms, used for representation of documents in the collection, according to some criterion, which is different from clustering task. The proposed method assigns a score to a term based on the number of documents in which the term is present and document length normalized term frequency. Document length normalized term frequency is the ratio of the frequency of occurrence of a term in a document to the sum of the frequencies of occurrences of all terms in the document. Clustering results on two ideal text data sets TDT2 and Reuters21578 proved that the proposed method selects features with more discriminating power when compared with that selected by the existing unsupervised filter based feature section methods.

Keywords: Clustering, Document Level Term Weight, Term, Term Document Frequency, Text Document, Unsupervised Feature Selection

I. Introduction

Document Clustering aims to discover groups, or clusters, of similar documents. The similarity between documents is often determined using distance measures [1] over the

various dimensions of the documents in the collection. It is a fundamental operation used in the unsupervised document organization, automatic topic extraction and information retrieval. The motivation behind clustering a collection of documents is to find inherent structure in the collection and expose this structure as a set of coherent groups.

The most widely used document representation model is the Vector Space Model (VSM) [2]. A document in the VSM can be defined as the following mapping

$$\begin{aligned} &\phi : d \mapsto \phi(d) \\ &= (w(t_1, d), w(t_2, d), \cdots, w(t_m, d)) \in \Re^m \end{aligned} \tag{1}$$

Where $w(t_i, d)$ is the weight of term t_i in document d and m is the size of term vocabulary of the document collection. The most basic way of estimating $w(t_i, d)$ is by finding the frequency of occurrence of term t_i in document d. Popular term weighting schemes are discussed in [3]. The term vocabulary of a document collection comprises of all the terms in the pre-processed documents of the collection. Many variations of VSM have been proposed in [4] that differ in what they consider as a feature or term. The most common approach is to consider unique words that are present in the document collection after pre-processing the documents as distinct terms. The most common document pre-processing steps are stop-word elimination and stemming. Very frequent words such as articles, prepositions, conjunctions, etc., that carry little information about the content of a document are removed during stop-word elimination phase. A general stop-word list in English is given in [5]. During stemming all variants of a word are replaced with a single common stem. Porter stemming algorithm [6] is the most popular algorithm for document collection written in English. To represent the whole corpus of n documents, the Term Document matrix, D is introduced. D is a $m \times n$ matrix whose rows and columns are indexed by terms and documents, respectively. Thus VSM represents a document in term space, which causes a huge increase in the dimensionality of the matrix D. Not all the documents in a collection contain all the terms used in the representation and as a result sparseness occurs in the document vectors enormously.

Clustering algorithms struggle with high dimensional data due to the curse of dimensionality. As the number of dimensions in a document representation increases, distance measures become increasingly meaningless. Additional dimensions spread out the documents in the term space until, in very high dimensional term space, they are almost equidistant from each other. Hence dimension reduction techniques are essential not only to reduce the computational effort required for document clustering but also to improve clustering performance. Dimension reduction techniques are broadly classified into feature extraction and feature selection techniques. Feature extraction methods attempt to summarize a data set with fewer dimensions by creating combinations of the original attributes. These techniques can successfully uncover the latent structure in data sets. However, since they preserve the relative distances between objects, they are less effective when there are large numbers of irrelevant attributes that hide the clusters.

Also, the new features are combinations of the originals and may be very difficult to interpret the new features in the context of the domain. Feature selection methods select only the most relevant of the dimensions from a data set, to reveal groups of objects that are similar on only a subset of their attributes. Feature selection methods can be broadly divided into two categories filters [7] and wrappers [8]. The filter methods evaluate the relevance of each feature (subset) using the data set alone, regardless of the subsequent learning algorithm. On the other hand, wrapper methods invoke the learning algorithm to evaluate the quality of each feature (subset). Specifically, a clustering algorithm is run on a feature subset and the feature subset is assessed by some estimate of the clustering performance. Wrappers are usually more computationally demanding, but they can be superior in accuracy when compared to filters, which ignore the properties of the learning task at hand [8]. Overview of dimension reduction techniques can be found in [9]. Unsupervised feature selection exploits data variance and separability to evaluate feature relevance. This paper focuses on filtering strategy for its efficiency and effectiveness in handling data sets with large size and high dimensions [10], [11].

The contribution of this paper is mainly two manifolds. First, it proposes a novel feature ranking technique for dimensionality reduction. Second, experiments are carried out to evaluate the proposed method in comparison with the state of the art unsupervised feature ranking methods.

The remainder of this paper is organized as follows: Section II presents a review of unsupervised feature ranking techniques. Section III proposes the novel feature ranking technique. Section IV gives details about experimental design. Results of experimental evaluations are given in Section V. Section VI concludes the paper.

II. Existing Unsupervised Feature Ranking Methods

II.1. Collection Frequency and Inverse Document Frequency

Collection frequency and Inverse Document Frequency (CFIDF) have been proposed by the authors in their previous work [12]. CFIDF ranks terms according to its collection frequency and document frequency and can be expressed mathematically as shown in (2).

$$w(t) = \frac{tcf(t)}{df(t)} \log(df(t)) \tag{2}$$

Where, $w(t)$ is the weight assigned to the term t, $tcf(t)$ is the term collection frequency and is defined as the total number of occurrences of the term t in the document collection and $df(t)$ is the document frequency of the term t which is given by the total number of documents in which the term t appears.

II.2. Term Variance

Term Variance (TV) [13] is used to calculate the variance of the terms in the collection. It gives more weight to those terms that are present in several documents in the

collection and have a non-uniform distribution through out the collection. It can be expressed as shown in (3).

$$Var(t_i) = \sum_{j=1}^{n} (f_{ij} - \overline{f_i})^2 \tag{3}$$

Where, n is the number of documents in the collection, f_{ij} is the frequency of term t_i in $j-th$ document and $\overline{f_i}$ is the mean frequency of the term t_i in the document collection.

II.3. Term Variance Quality

Term Variance Quality - TVQ [14] is very similar to Term Variance and uses the total variance to calculate the quality of a term, as shown in (4).

$$TVQ(t_i) = \sum_{j=1}^{n} f_{ij}^2 - \frac{1}{n}\left[\sum_{j=1}^{n} f_{ij}\right]^2 \tag{4}$$

Where, n is the number of documents in the collection, f_{ij} is the frequency of term t_i in $j-th$ document.

II.4. Laplacian Score

Unsupervised feature ranking based on Laplacian score (LS) is proposed in [15]. The calculation of LS is based on a graph G that captures nearest neighbour relationships between the n documents. G is represented by a square matrix S where $S_{ij} = 0$ unless the documents d_i and d_j are neighbours, in which case:

$$S_{ij} = e^{\frac{-\left\| d_i - d_j \right\|^2}{t}} \tag{5}$$

Where, t is a bandwidth parameter and is usually set to the arithmetic mean of average dissimilarity between each document and the rest of the documents in the document collection. Document d_j is said to be a neighbour of document d_i if and only if d_j belongs to the K nearest neighbours of d_i. $L = Q - S$ is the Laplacian of this graph where Q is a degree diagonal matrix $Q_{ii} = \sum_j S_{ij}$, $Q_{ij,i\neq j} = 0$. If f_i is the feature vector that has $i-th$ feature value of each document, then the LS for the $i-th$ feature is calculated using the following equations (6) and (7).

$$\hat{f}_i = f_i - \frac{f_i^T Q U}{U^T Q U} U \tag{6}$$

Where, U is a column vector of length n with all elements equal to one.

The Laplacian score for the $i-th$ feature is given by (7) :

$$LS_i = \frac{\hat{f}_i^T L \hat{f}_i}{\hat{f}_i^T Q \hat{f}_i} \tag{7}$$

II.5. Term Contribution

According to [16] the contribution of a term in a data set is defined as its overall contribution to the similarity between documents. It can be mathematically expressed as

$$TC(t) = \sum_{(i,j)\wedge(i\neq j)} w(t,d_i)w(t,d_j) \tag{8}$$

where, $w(t,d)$ represents the weight of term t in document d .

III. Proposed Feature Ranking Method

In this section a new feature ranking method, based on the document length normalized term frequency and term document frequency of a term (TFTDF), for effective document clustering is proposed.

III.1. Motivation

The goal of any document clustering method is to project documents into the subspace in which the documents with different semantics can be well separated, while the documents with common semantics can be clustered. To accomplish this, documents should be represented using terms that have more discrimination power for documents. A term will have more discriminating power if it is present in a small subset of documents in the text collection. A common term that is present in almost all documents is not useful at all in identifying the category to which the document belongs to and hence has low discriminating power. Also a term whose frequency of occurrence in a document is low does not contribute much to the semantics of the document. Thus, a term which is present in a small subset of documents and whose frequency of occurrence in a document is high is considered to have more discriminating power.

III.2. Term Frequency and Term Document Frequency of a Term (TFTDF)

The contribution of a term to the semantics a document is proportional to the frequency of occurrence of the term in the document. So the weight assigned to a term in a document is given by (9).

$$w(t_i,d_j) = f(t_i,d_j) \tag{9}$$

where, $w(t_i, d_j)$ represents the weight of the term t_i in the $j-th$ document and $f(t_i, d_j)$ is the local frequency of term t_i in $j-th$ document and is given by the frequency of occurrence of the term t_i in the $j-th$ document. The problem with (9) is the length of the document in which the term occurs is not taken into consideration while calculating the weight of the term in a document. According to (9) two terms with the same frequencies of occurrence in two documents with different length is same. To overcome this the length of the document should be considered while calculating $w(t_i, d_j)$ as in (10).

$$w(t_i, d_j) = \frac{f(t_i, d_j)}{l_j} \tag{10}$$

where, l_j is the length of the $j-th$ document and is defined as the total number of terms in the document. Thus the total weight of the term t_i considering all document level weights in the collection is given by (11).

$$W_d(t_i) = \sum_{j=1}^{n} [w(t_i, d_j)]^2 \tag{11}$$

where $W_d(t_i)$ is the total weight of the term considering all document level weights.

Also a term will have more discriminating power if it is present in a small subset of documents in the text collection. A common term that is present in almost all documents is not useful at all in identifying the category to which the document belongs to and hence has low discriminating power. So, to find terms with more discriminating power the weight of a term is calculated according to its document frequency as in (12).

$$W_c(t_i) = \log\left(\frac{L}{df(t_i)}\right) \tag{12}$$

where $W_c(t_i)$ is the collection level weight of the term t_i, L is the total number of terms in the collection and $df(t_i)$ is the document frequency of the term t_i, which is the number of documents in the collection that contains the term.

Overall weight of the term t_i is given by (13)

$$W(t_i) = W_d(t_i) \times W_c(t_i) \tag{13}$$

IV. Experimental Design

Experiments are conducted to evaluate the effectiveness of proposed TFTDF feature selection technique, in comparison with existing unsupervised filter based feature selection techniques including TV, TVQ, LS, TC and CFIDF. Documents in the data set are represented using VSM, in the form of Term Document matrix D. Each element

d_{ij} of the matrix D represents the frequency of occurrence of the term t_i in $j-th$ document. In all the experiments, the terms are ranked based on the score assigned to it by the feature selection technique. The term that has the highest score is given best rank and so on. The top q terms are then used for representation of documents in the collection in the form of a modified Term Document matrix D'. Documents represented in the reduced dimension space given by the matrix D' are partitioned by the standard k-means algorithm [17]. The effectiveness of the feature selection techniques is measured using the clustering performance. For each data set clustering is performed on a document collection that contains k number of classes. For each k, 10 test runs were conducted on different randomly chosen clusters and the average performance is reported in the results tables. To evaluate the effect of number of classes k in the document collection on clustering performance in the reduced dimension space, the value of k is varied from 2 to 10 in steps of 1. To evaluate the effect of number of features q used for document representation on clustering performance, q is varied from 50 to 325, at increments of 25 for a chosen value of k. The lower limit for q is selected so as to avoid empty clusters that results from insufficient number of terms used for representing documents in the collection. For each combination of k and q clustering is performed using K-Means algorithm. The K-Means algorithm available in MATLAB is used in this paper. As the algorithm randomly chooses initial cluster representatives, for the purpose of reproducing results given in this paper the random number generator algorithm in MATLAB is seeded with the following parameters: twister and 5489. The parameters used for K-Means algorithm are as follows: "Distance" option used is "cosine", "EmptyAction" option is chosen as "singleton", "Start" option is set to "cluster" and the number of replicates is set to 10.

IV.1. Benchmark Data Sets

Two text document collections, namely TDT2 and Reuters-21578 are considered in evaluating feature selection methods. Vector Space Model (VSM) is used to represent documents in the three collections. No term weighting measures are used for document representation. The two documents corpora have been among the ideal test sets for document clustering purpose because documents in the collections are manually clustered based on their topics and each document has been assigned one or more labels indicating which topic / topics it belongs to.

Table I provides the statistics of the two document corpora. The data sets available at http://www.zjucadcg.cn/dengcai/Data/TextData.html are used in this paper. The ratio of the size of the biggest cluster to the size of the smallest cluster in a data set is highest for Reuters21578 data set.

TABLE I CHARACTERISTICS OF TEXT DOCUMENT COLLECTIONS

Characteristic	TDT2	Reuters 21578
Number of documents	9394	8213
Number of terms	36771	18933
Sparsity of D	99.65 %	99.75 %
Number of classes	30	41
Maximum class size	1844	3713
Minimum class size	52	10
Median class size	131	37
Average class size	313	200

The class distributions of the data sets are shown in Fig. 1 and Fig. 2. The Reuters21578 data set has most uneven class distribution.

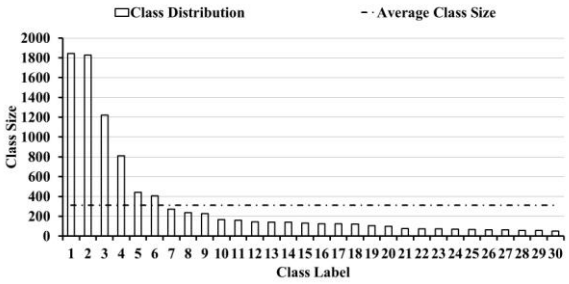

Fig. 1. Class Distribution of TDT2 data set

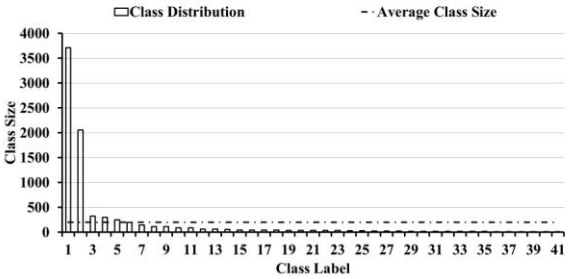

Fig. 2. Class Distribution of Reuters21578 data set

IV.2. Cluster Quality Evaluation

In this paper two external clustering quality evaluation measures namely Normalized van Dongen (NVD) proposed by [18] and Combined Bcubed measure (BCF) proposed by [19] are used. External measures compare the partitioning obtained by the clustering algorithm with a ground truth partitioning created by human annotators. Let $P = \{P_1, P_2, \cdots, P_k\}$ be a partitioning of documents obtained by a clustering algorithm, n

be the total number of documents in the collection, and $|P_j|$ be the number of documents in $j-th$ cluster. Let $C=\{C_1,C_2,\cdots,C_c\}$ be the ground-truth partitioning of the documents, and C_{ij} be the number of documents in $j-th$ cluster from $i-th$ class. The external measures use the contingency matrix $C=[C_{ij}]_{c\times k}$ to estimate the quality of clustering.

IV.2.1. Normalized van Dongen criterion

The van Dongen criterion (VD) [20] was originally proposed for evaluating a graph clustering. VD measures the representativeness of the majority objects in each class and in each cluster. A normalized version of VD proposed by [18] (NVD) as shown in (14) is used in this paper. **The NVD value lies in the interval [0,1]. Smaller values of NVD indicate better clustering performance and vice versa.**

$$NVD=\frac{2n-\sum_i \max_j C_{ij}-\sum_j \max_i C_{ij}}{2n-\max_i|C_i|-\max_j|P_j|} \tag{14}$$

IV.2.2. Combined Bcubed Precision and Bcubed Recall

Bcubed precision (BP) [21] of an item is the proportion of items in its cluster, which have the item's category (including itself). The overall BCubed precision is the averaged precision of all items in the distribution. The metric BP associated with one item represents how many items in the same cluster belong to its category.

Bcubed recall (BR) [21] of an item is the proportion of items in its category which belong to item's cluster (including itself). The overall BCubed recall is the averaged recall of all items in the distribution. The metric BR associated with one item represents how many items from its category appear in its cluster.

Since the average is calculated over all items, it is not necessary to apply any weighting according to the size of clusters or categories. A standard way of combining metrics is Van Rijsbergen's F [22] and it is computed using (15).

$$F(BR,BP)=\frac{1}{\alpha\frac{1}{BP}+(1-\alpha)\frac{1}{BR}} \tag{15}$$

where, α and $(1-\alpha)$ are the weights of BP and BR, respectively. According to [19], this combined metric $F(BR,BP)$ when $\alpha=0.5$ satisfies all formal constraints on text clustering evaluation metrics. This paper uses the notation 'BCF' to represent the combined metric $F(BR,BP)$. **The BCF value lies in the interval** $[0,1]$ **and higher values of the measure indicates better clustering performance and vice versa.**

V. Results and Discussion

Tables II, III, IV and V show the minimum number of features required to achieve clustering performance, greater than or equal to that when all features are considered for document representation.

TABLE II MIN. NUMBER OF FEATURES REQUIRED TO ACHIEVE CLUSTERING PERFORMANCE (NVD) WITH ALL FEATURES ON TDT2

k	TV	LS	TC	CFIDF	TFTDF
2	50	150	**50**	50	75
3	**125**	425	125	175	150
4	325	325	375	175	**100**
5	525	250	625	150	**100**
6	825	>1500	**700**	875	1025
7	250	475	375	175	**150**
8	>1500	>1500	1225	**525**	900
9	550	>1500	800	925	**375**
10	350	500	1125	**250**	275

The smallest value corresponding to the value of k is highlighted in boldface. When the smallest value is same for two or more feature selection methods, the value corresponding to the feature selection method that gives rise to the best clustering performance, is highlighted in boldface.

TABLE III MIN. NUMBER OF FEATURES REQUIRED TO ACHIEVE CLUSTERING PERFORMANCE (BCF) WITH ALL FEATURES ON TDT2

k	TV	LS	TC	CFIDF	TFTDF
2	75	150	**50**	75	50
3	**100**	425	**100**	125	150
4	250	325	375	175	**100**
5	525	300	625	550	**125**
6	775	>1500	700	875	**525**
7	250	475	375	175	**175**
8	>1500	>1500	**800**	>1500	875
9	575	>1500	>1500	925	**400**
10	350	500	1150	250	**200**

TABLE IV MIN. NUMBER OF FEATURES REQUIRED TO ACHIEVE CLUSTERING PERFORMANCE (NVD) WITH ALL FEATURES ON REUTERS

k	TV	LS	TC	CFIDF	TFTDF
2	50	100	50	50	**50**
3	50	50	50	**50**	50
4	**350**	>1000	>1000	>1000	>1000
5	175	600	300	**100**	125
6	100	575	225	100	**75**
7	325	925	250	175	**100**
8	525	>1000	600	500	**450**
9	600	>1000	575	475	**375**
10	400	850	375	350	**200**

TABLE V MIN. NUMBER OF FEATURES REQUIRED TO ACHIEVE CLUSTERING PERFORMANCE (BCF) WITH ALL FEATURES ON REUTERS

k	TV	LS	TC	CFIDF	TFTDF
2	225	175	125	75	**75**
3	50	50	50	**50**	50
4	375	>1000	**250**	275	>1000
5	175	775	>1000	**150**	150
6	100	575	75	100	**75**
7	325	750	250	125	**100**
8	550	>1000	675	**550**	700
9	600	>1000	600	475	**300**
10	500	>1000	450	425	**250**

The average value of clustering performance using different feature selection methods for a given data set, across a different number of classes k and a different number of features q, is compared with average clustering performance for different values of k when all features are considered (*ALLF*) in Tables VI and VII. The value corresponding to the feature selection method that gives rise to the best clustering performance, is highlighted in boldface.

TABLE VI AVERAGE CLUSTERING PERFORMANCE ON TDT2

Metric	using features from 50 to 325, at increments of 25 (X $\mathbf{10^{-2}}$)					ALLF
	TV	LS	TC	CFIDF	TFTDF	
NVD	24.28	26.38	26.80	23.64	**22.87**	21.96
BCF	82.51	81.19	80.14	83.12	**83.85**	84.86

TABLE VII AVERAGE CLUSTERING PERFORMANCE ON REUTERS

Metric	using features from 50 to 325, at increments of 25 (X $\mathbf{10^{-2}}$)					ALLF
	TV	LS	TC	CFIDF	TFTDF	
NVD	55.78	57.51	55.94	55.49	**54.95**	54.80
BCF	59.86	58.83	59.73	60.16	**60.48**	60.87

Fig. 3, Fig. 4, Fig. 5 and Fig. 6 show the effect of number of features q used for document representation on clustering performance averaged over a different number of classes from 2 to 10, for different feature selection techniques. Results for the feature selection technique Term Variance Quality (TVQ) are not shown in the tables and figures, because its performance matches exactly with that of Term Variance (TV).

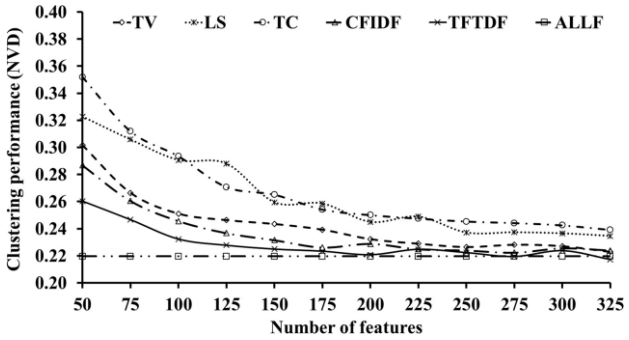

Fig. 3. Clustering performance (NVD) using different number of features for document representation on TDT2 data set

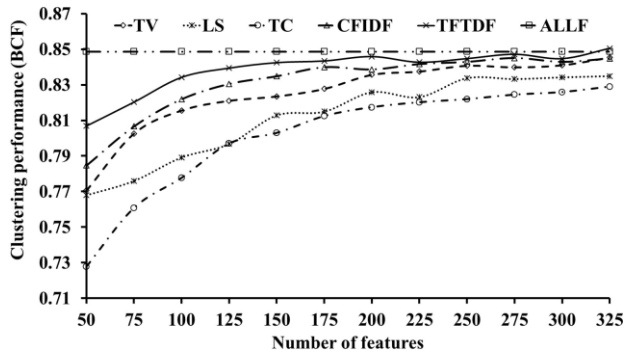

Fig. 4. Clustering performance (BCF) using different number of features for document representation on TDT2 data set

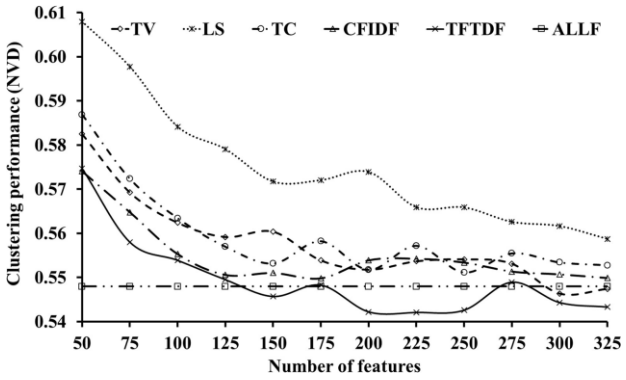

Fig. 5. Clustering performance (NVD) using different number of features for document representation on Reuters data set

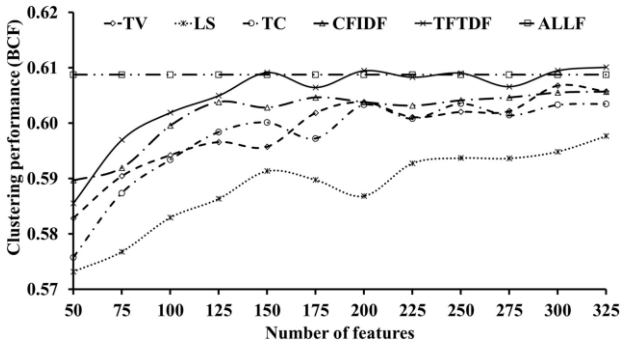

Fig. 6. Clustering performance (BCF) using different number of features for document representation on Reuters data set

The average execution time of different feature selection methods using a computer with Intel core i7-4700 MQ 2.4 GHz processor and 8 GB RAM on TDT2 data set with 10 categories is given in Table VIII. The value corresponding to the feature selection method with the least average execution time is highlighted in boldface.

TABLE VIII AVERAGE EXECUTION TIME (SEC) OF DIFFERENT FEATURE SELECTION METHODS

TV	LS	TC	CFIDF	TFTDF
3.013	2.116	2.450	0.397	**0.261**

VI. Conclusion

This paper proposes a novel filter based unsupervised feature selection method. The proposed method estimates the discriminatory power of a term based on the document level parameter and collection level parameter. The clustering performance of the TDT2 data set is more when compared to that with Reuters data set, for all feature selection methods. Relatively poor performance of feature selection methods for Reuters data set is due to large imbalance in cluster size as shown in Table I. The specific advantage of the proposed feature selection method is that the weight assigned to a term can be calculated incrementally whenever a new document is added or removed from the document collection. Empirical evaluations demonstrate that the proposed method is computationally less complex and outperforms other feature selection methods in selecting features with more discrimination power. The average clustering performance of the proposed feature selection method with just 150 top ranked features is almost equal to the clustering performance when all features are considered.

References

[1] Huang, A., 2008, "*Similarity measures for text document clustering*," Proc. 6th New Zealand Computer Science Research Student Conference, pp. 49-56.

[2] Salton, G., Wong, A., and Yang, C. S., 1975, "A Vector Space Model for Automatic Indexing," *Communications of the ACM.,* 18(11), pp. 613-620.

[3] Bai, V., and Manimegalai, D., 2013, "A Document Level Measure for Text Categorization," *International Review on Computers and Software* (IRECOS)., 8(6), pp. 1374-1381.

[4] Keikha, M., Razavian, N. S., Oroumchian, F., and Razi, H. S., 2008, "Document representation and quality of text: An Analysis," M. W. Berry, et al., eds., Survey *of Text Mining II,* Springer-Verlag, London, pp. 219-232.

[5] Fox, C., 1989, "A stop list for general text," *ACM SIGIR forum,* 24, 1-2, ACM, New York, pp. 19-21.

[6] Porter, M. F., 2006, "An algorithm for suffix stripping," *Program: electronic library and information systems,* 40(3), pp. 211-218.

[7] Blum, A. L., and Langley, P., 1997, "Selection of Relevant Features and Examples in Machine Learning," *Artificial Intelligence,* 97(1), pp. 245-271.

[8] Kohavi, R., and John, G. H., 1997, "Wrappers for feature subset selection," *Artificial intelligence,* 97(1), pp. 273-324.

[9] Cunningham, P., 2008, "Dimension Reduction," M. Cord, et al., eds., Machine learning techniques for multimedia, Springer-Verlag, Berlin Heidelberg, pp. 91-112.

[10] Cantupaz, E., Newsam, S., and Kamath, C., 2004, "*Feature Selection in Scientific Applications*," Proc. 10th ACM SIGKDD international conference on Knowledge Discovery and Data mining, pp. 788-793.

[11] Guyon, I., and Elisseeff, A., 2003, "An introduction to variable and feature selection," *The Journal of Machine Learning Research,* 3, pp. 1157-1182.

[12] Sivaram Prasad, N., and Raja Sekhara Rao, K., 2014, "Subspace Clustering of Text Documents using Collection and Document Frequencies of Terms," *International Review on Computers and Software* (IRECOS), 9(10), pp. 1692-1699.

[13] Liu, L., Kang, J., Yu, J., and Wang, Z., 2005, "A Comparative Study on Unsupervised Feature Selection Methods for Text Clustering," Proc. IEEE International Conference on Natural Language Processing and Knowledge Engineering, pp. 597-601.

[14] Dhillon, I., Kogan, J., and Nicholas, C., 2004, "Feature selection and document clustering," M. W. Berry, eds., Survey *of text mining*, Springer-Verlag, New York, pp. 73-100.

[15] He, X., Cai, D., and Niyogi, P., 2005, "Laplacian score for feature selection," *Advances in neural information processing systems*, pp. 507-514.

[16] Liu, T., Liu, S., Chen, Z., and Ma, W. Y., 2004, "*An evaluation on feature selection for text clustering*," Proc. 20th International Conference on Machine Learning, pp. 488-495.

[17] Hartigan, J., and Wong, M., 1979, "Algorithm as 136: A k-means clustering algorithm," *Applied Statistics*, 28(1), pp. 100-108.

[18] Wu, J., Xiong, H., and Chen, J., 2009, "Adapting the right measures for k-means clustering," Proc. 15th ACM SIGKDD international conference on Knowledge Discovery and Data mining pp. 877-886.

[19] Amigó , E., Gonzalo, J., Artiles, J., and Verdejo, F., 2009, "A comparison of extrinsic clustering evaluation metrics based on formal constraints," *Information retrieval*, 12(4), pp. 461-486.

[20] Dongen, S., 2000, "Performance Criteria for Graph Clustering and Markov Cluster Experiments", Centre for Mathematics and Computer Science. Amsterdam, The Netherlands.

[21] Bagga, A., and Baldwin, B., 1998, "Entity-based cross-document co-referencing using the vector space model," Proc. 17th International Conference on Computational Linguistics, pp.79-85.

[22] Van Rijsbergen, C. J., 1974, "Foundation of evaluation," *Journal of Documentation*, 30(4), pp. 365-373.

Authors' information

Sivaram Prasad Nalluri is working as a professor in the Information Technology department of the Bapatla Engineering Collegee, Bapatla, India. He received the Master's degree (M. Tech.) in computer science and engineering from Jawaharlal Nehru Technological University at Hyderabad, India in 2002. He received the Bachelor's degree (B. Tech.) from Acharya Nagarjuna University at Guntur, India in 1995. His research interests include Data mining and Digital image processing.

Rajasekhara Rao Kurra is working as a professor in the Computer Science and Engineering department of Sri Prakash College of Engineering, Tuni, India. He received the PhD degree in computer science and engineering from Acharya Nagarjuna University at Guntur, India in 2008. He received his Master's degree MS in software systems from BITS Pilani, India in 1992. He received Bachelor's degree B.Tech in electronics and communication engineering from Acharya Nagarjuna University at Guntur, India in 1985. His research interests include Data mining and Embedded Systems. Dr. Rajasekhar is a fellow of IETE, life member of IE, ISTE, ISCA and CSI.