An Empirical Evaluation Of The State Of Art Feature Selection Methods For Text Categorization

Ananya Gupta, Shahin Ara Begum

Abstract: Feature selection methods select a small subset of the relevant features from the original feature space by eliminating redundant or irrelevant features. In the process it also reduces the dimensionality of the feature space and improves the efficiency of the data mining algorithms. In this paper, sixteen state of art feature selection methods are studied that use different benchmark datasets with respect to text categorization and their performance is summarized. The past research reveals that performance of feature selection methods are dataset specific. In the present work, further experiments are carried out with the state of art feature selection methods for text categorization over a unifying framework of benchmark datasets to evaluate and compare their performance on same standards. The efficiency of the methods is evaluated on the basis of their performance with k-means clustering and KNN classification. The experiments reveal that unsupervised feature selection method of Multi-Cluster Feature Selection (MCFS) performs the best in comparison to the state of art feature selection methods studied in the present work. MCFS reduces the data dimensionality to an extent of 95% on an average on the considered datasets with acceptable results of classification and clustering.

Index Terms: Classification, Clustering, Dimensionality reduction, Feature selection, Predictive accuracy, Text categorization, Text datasets.

1. INTRODUCTION
Technological advances have led to data explosion both in perspective of dimensionality and size of samples. Various machine learning applications pertaining to text mining, biomedical field and computer vision have been confronted with the problem of high dimensionality of the data. With the rapid growth of database technologies, data mining from databases and machine learning are gaining increasing popularity. Knowledge acquisition has become very important to study these large-scale datasets. The main challenge in handling these large-scale datasets is their problem of high dimensionality. The problem of dimensionality has imposed a very big challenge towards the efficiency of the machine learning algorithms. The machine learning algorithms cannot handle these high dimensional data which in turn makes the machine learning tasks intractable. Thus, it becomes necessary to reduce the dimensionality of the data. Feature selection is such a method of dimensionality reduction, wherein small subsets of features that are relevant are chosen. It not only removes the irrelevant and redundant features but also reduces the computational cost and improves predictive capability. Feature selection is broadly guided by two aspects: 1) label information 2) search strategy (Fig. 1). Feature selection methods can be categorized into three types on the basis of labeled information. They are: supervised feature selection [1], [2], [3], [4] semi-supervised feature selection [5], [6], [7], and unsupervised feature selection [8], [9], [10], [11], [12]. Supervised methods are those that are guided by the presence of labeled information. Studies on supervised methods can be found in [3], [13]. Semi-supervised methods are employed when a small portion of the data is labeled. Most of the semi-supervised methods are graph based learning methods that rely on similarity matrix [5], [14]. Unsupervised methods are used when datasets are devoid of labels. The absence of labels in unsupervised methods makes feature selection a much harder task [9]. Based on search strategy, feature selection can be of three types – the filter method, wrapper method and the hybrid method. The filter method is based on rank and scores of the features based on certain statistical criterion. The features with top score are considered to be the potential features for the target concept. Filter methods are fast, but they lack in robustness in terms of multi-way feature interactions. Another crucial aspect of filter methods is the selection of the cutoff points while selecting the discriminative features (i.e. the value of the cut off score of ranking). Frequently used filter methods include t-test [15], chi-square test [16], Wilcoxon Mann–Whitney test [17], mutual information [18], and principal component analysis [19]. Filter components do not involve any mining algorithm. In the wrapper approach, a pre-determined mining algorithm is used to evaluate the quality of feature subset. It searches for features that are suitable for the mining algorithm. The learning algorithm is applied on the subset of features and tested on a hold-out set or the test data and its prediction accuracy is used to determine the quality of the feature subset. Usually, wrappers are more effective than filter methods, although they are computationally more expensive than the filter methods [20], [21]. In hybrid methods, initially the filter approach is applied to select a feature pool and then the wrapper method is implemented on the feature pool to select an optimal subset of features.
The rest of the paper is outlined as follows: Section 2 describes feature selection process, Section 3 describes state of art feature selection methods, Section 4 describes experimental setup to evaluate the state of art methods over three benchmark datasets to evaluate the feature selection methods on the same standards of the considered datasets and Section 5 concludes the paper.

2 FEATURE SELECTION PROCESS
A feature selection process typically comprises of four stages: generation of subset, evaluation of subset, stopping criterion to stop the iterative process of subset generation and validation of the results.

2.1 Subset Generation
The subset generation procedure generates the candidate feature subset. Each state in the search space is a candidate set for evaluation. Two issues that are of utmost importance in this stage are the search starting points and the search strategy. Search can be a forward search, or backward search, or can be both forward or backward simultaneously. Search may also begin with a random subset to avoid being trapped in local optima [22]. Search strategies can be complete, sequential or random depending on the cardinality of the set.

2.2 Subset Evaluation
An evaluation function measures the goodness of a subset produced using some criterion function. The value obtained by the function is compared with the previous best value. If it is found to be better than the previous one, then it replaces it. Evaluation functions can be independent or dependent on the basis of the dependency criterion of the mining algorithms [23], [24]. The former is used mainly in filter models. It utilizes the intrinsic properties of the data such as the information measure, distance measure, dependency measure, and the consistency measure [25], [26], [27], [28], [29]. The dependent criterion of evaluation is used in case of wrapper methods. If the predictive accuracy of the mining algorithm is high, then the feature subset consists of features that are better suited for the algorithm. Thus, this evaluation measure is dependent on the selected features. Classifier error rate is one such dependent evaluation measure [30].

2.3 Stopping Criterion
Stopping criterion is necessary for feature selection process, otherwise it may run exhaustively or forever through the space of subsets. Generation process and evaluation functions can influence the choice for a stopping criterion. Stopping criteria of a generation process include whether a predefined number of features are selected, and whether a predefined number of iterations are reached. Stopping criteria of an evaluation function may be attributed to two conditions 1) whether further addition (or deletion) of features does not produce a better subset than the previous and 2) whether an optimal subset is already obtained using some evaluation function.

2.4 Validation
In this stage, validation of the results is done either using the synthetic or real-world data sets. In case of synthetic datasets, prior knowledge about the relevant features aids in validation of the actual results obtained. However, in case of real-world datasets, prior knowledge is unknown and thus, some indirect measures are employed [31].

2.5 Factors affecting the choice of feature selection algorithms
The choice of feature selection method for a specific task has always been a dilemma, given a large number of available algorithms. A complete account of the factors that guide the choice of feature selection method can be found in [32]. Primarily the data mining task, namely classification or clustering needs to be ascertained. Evaluation criterion is affected by the choice of the mining task. Search strategy is purpose specific. Additional information on knowledge and data factors play key role in resolving the choice of suitable algorithms. The knowledge factor can be further categorized into purpose of feature selection, time concerned, expected output type and the ratio of relevant features to irrelevant features. The data factor comprises of class information, feature type, quality of data and the ratio between total number of features and the number of instances.

3 FEATURE SELECTION METHODS
Research over the years has led to availability of extensive feature selection methods. In this paper, we restrict the discussion on some feature selection methods for text categorization. Some state of art selection methods for text categorization is briefly stated here in the section.

(i) Information Gain (IG)
IG is a commonly adopted method used for feature selection. This criterion is used to ascertain the goodness of the term in field of machine learning [33], [34], [35]. Let the global probability of the class i be $p_i$. $p_i(t)$ is the probability of the class i considering term t is in the document. $F(t)$ is the global fraction of documents containing t. Information gain $I(t)$ for a term t is
\[ I(t) = - \sum_{i=1}^{k} P_i \log P_i + \sum_{i=1}^{k} P_i(t) \log (P_i(t)) \] 

\[ + (1-F(t)) \sum_{i=1}^{k} (1-P_i(t)) \log (1-P_i(t)) \]

Greater the value of \( I(t) \) greater is the discriminatory power of \( t \). The terms which are below a predefined threshold value of \( I(t) \) are removed.

(ii) Mutual Information (MI)
MI is derived from information theory and it gives the mutual information between classes and features. Mutual information \( M_{ij}(t) \) between term \( i \) and class \( j \) is the co-occurrence of term \( i \) and class \( j \). Mutual information is given by
\[ M_{ij}(t) = \log \frac{F(t).P_i}{F(t).P_i} \]

(2) when \( M_{ij}(t) > 0 \), \( t \) is correlated positively to \( i \) and when \( M_{ij}(t) < 0 \), \( t \) is negatively correlated to \( i \). The value of \( M_{ij}(t) = 0 \), if \( t \) and \( i \) are independent. Mutual Information suffers from the drawback that it is influenced by marginal probabilities [35].

(iii) \( \chi^2 \) statistics (CHI)
CHI[35] computes the dependence between the class \( i \) and term \( t \). Let \( N \) be the total number of documents in the corpus, \( p_i(t) \) is the conditional probability that class \( i \) contains the term \( t \) and \( F(t) \) is the global fraction of documents which contain \( t \). The \( \chi^2 \) statistics is given by
\[ \chi^2 (t) = \frac{n.F(t)^2.(p_i(t) - P_i)^2}{F(t).(1-F(t)).P_i(1-P_i)} \]

\( \chi^2 \) statistics is the normalized measure and is advantageous than mutual information. Terms belonging to the same category can be easily measured using \( \chi^2 \) statistics. However, it is not a reliable measure for low frequency terms [36].

(iv) Term Strength (TS)
TS was used by Yang and Wilbur in text categorization [35], [37]. The term importance is measured on how frequently the term is probable to appear in closely related documents. Term strength is calculated based on the conditional probability that the term occurring in the second half of the pair of related document appears in the first half. Let \( x \) and \( y \) be two arbitrary pairs of distinct but related documents. Then the term strength of the term \( x \) is given by
\[ s(t) = P_j(t \in y \mid t \in x) \]

(v) Document Frequency (DF)
DF gives the frequency of documents in which a term occurs [35]. The document frequency for each term is calculated and the terms below a predefined limit are removed considering them to be non-informative terms. It is the simplest method of feature selection and its computational complexity is approximately linear to the number of training documents.

(vi) Entropy-based Ranking
The approach of entropy-based ranking was proposed in [38]. The nature of the term is measured by reduction in entropy on removal of the term. The entropy of a term is given by:
\[ E(t) = -\sum_{i=1}^{k} \sum_{j=1}^{k} (S_{ij}.\log (S_{ij})) + (1-S_{ij}).\log (1-S_{ij}) \]

(5) where, \( S_{ij} \) is the similarity between the \( i^{th} \) and the \( j^{th} \) document and \( S_{ii} \in (0,1) \). \( S_{ij} \) is defined as
\[ S_{ij} = e^{-\alpha . \text{dist}(i,j)}, \alpha = -\ln 0.5 \]

\( \text{dist}(i,j) \) is the distance between the terms \( i \) and \( j \) and \( \text{dist} \) is the average distance in between the documents after the removal of term \( t \). The computation of \( E(t) \) for each term \( t \) requires \( O(n^2) \) operations.

(vii) Fast Clustering Based Feature Selection Algorithm (FAST)
Song, Qinbo et al. [39] proposed a feature selection algorithm FAST for high dimensional dataset. It eliminates irrelevant features by calculating the T-Relevance value of each feature, and retains the relevant features which are greater than the pre-defined threshold value. A weighted complete graph is constructed by calculating the F-Correlation value between the pair of features and setting it as the weight of the edges between the vertices (features). The complete graph represents the correlations between the target-relevant features.

(viii) Probability based Term Weighting Features Selection
The distribution of text data is often imbalanced. Classifiers corresponding to categories with fewer instances do not perform well. This method handles the data imbalance problem using probability based term weighting feature selection method for categorization of documents belonging to minor categories. This method replaces the idf factor of the tf-idf weighting scheme [40] [41] [42]. The idf term is replaced by feature value. The feature value utilizes two critical information ratios to compute the terms weight. The two ratios give the most informative information on the term’s strength to its corresponding category. The weighting scheme is formulated as:

\[ tf \cdot \log \left( 1 + \frac{A}{B \cdot C} \right) \]

(7) where, \( A \) denotes the number of documents belonging to category \( c_1 \) where the term \( t_k \) occurs at least once; \( B \) denotes the number of documents not belonging to category \( c_1 \) where the term \( t_k \) occurs at least once; \( C \) denotes the number of documents belonging to category \( c_2 \) where the term \( t_k \) does not occur. \( A/B \) and \( A/C \) are the relevance ratios of the terms. \( A/B \) gives the relevance ratio of the term \( t_k \) if it is related to category \( c_1 \) only. Given two terms, \( t_k \), \( c_1 \) and category \( c_1 \), \( A/C \) the term with a higher value of will be the better feature to represent \( c_1 \).

(ix) Orthogonal Centroid Feature Selection
The orthogonal centroid feature selection (OCFS) is a method of feature selection that optimally selects features using
objective function according to orthogonal centroid algorithm [45] [46] [47]. The centroids of the class and the training samples are calculated. Subsequently, the score of the term is calculated based on the centroid of each class and the training set. The higher the score, more is its category information. The term score of the term $t_k$ is given by:

$$OCFS(t_k) = \sum_{j=1}^{n} \frac{1}{n} (m_j^k - m^k)^2$$

(8)

where, $n_j$ is the number of documents in the category $c_j$, $n$ is the total number of documents in the training set, $m_j^k$ is the $k$th element of the vector $m_j$ of the category $c_j$, $m_k$ is the $k$th element of the centroid vector of the entire training set and $|C|$ is the total number of categories in the corpus.

(x) Comprehensively Measure Feature Selection (CMFS)

Comprehensively Measure Feature Selection (CMFS) was proposed in [47]. It measures the discrepancy of a term both within the category and across the category. CMFS is defined for a term $t_k$ and category $c_i$ as follows:

$$CMFS(t_k, c_i) = \frac{tf(t_k, c_i) + 1}{tf(t_k) + |C|} (\frac{tf(t_k, c_i) + 1}{|V|}) - (\frac{tf(t_k) + 1}{|V|})$$

(9)

where, $tf(t_k, c_i)$ is the term frequency of the term $t_k$ in category $c_i$, $tf(t_k)$ is the frequency of the term $t_k$ in the whole training set, $|V|$ is the sum of the term frequencies of all terms in $c_i$, $|C|$ is the number of total categories and $|V|$ is the number of total terms in the feature space.

(xi) Distinguishing Feature Selector

Distinguishing Feature Selector[48], [49] is based on the assumptions: that term frequently occurring within a single class and not occurring in the other classes is a discriminative term and thus it must be assigned a high score, a term that rarely occurs within a class is irrelevant and therefore it is assigned a low score, a term frequently occurring in all classes is irrelevant and is assigned low score and term appearing in some of the classes is relatively distinctive and it is given a relatively high score. It is formulated as:

$$DFS(t) = \sum_{i=1}^{M} \frac{P(c_i | t)}{P(t | c_i) + P(t | c_i^-) + 1}$$

(10)

$M$ is the number of classes, $P(c_i | t)$ is the conditional probability of class $c_i$ given term $t$, $P(t | c_i)$ is the conditional probability of the absence of term $t$ given class $c_i$ and $P(t | c_i^-)$ is the conditional probability of term $t$ given the classes other than $c_i$.

(xii) Mutual information based feature selection MIFS-ND

MIFS-ND is a greedy feature selection method based on mutual information [50]. The optimal feature subset is obtained on the basis of feature-mutual information and feature-class mutual information combined.

(xiii) Gini Index

A novel Gini Index algorithm was proposed [51]. The underlying principle of this method is based on the concept of Gini-Index theory towards text feature selection. Let $p_1(t), p_2(t), ..., p_k(t)$ be the conditional probabilities that a document belongs to class $i$ considering that the term $t$ is in it. Thus, Gini Index of the term $t$, given by

$$G(t) = \sum_{i=1}^{k} p_i(t)^2$$

(11)

$G(t)$ range from $(1/k, 1)$. Greater the value of $G(t)$, better is the discriminative power of the term. The skewness present in the global class distribution at the start may interfere with accuracy in estimating the discriminative power of the underlying attributes.

(xiv) Complete Gini-Index Text (GIT) Feature Selection

Park et al. [52] reported that the feature selection method in [51] was not adequate to select the discriminative features. They proposed a new complete Gini Index Text (GIT) feature selection method that has the ability to obtain unbiased feature values. In the process it eliminates many redundant features from feature subsets while retaining the discriminative features. This new algorithm compared to the original version, demonstrates an overall noteworthy improved performance with respect to classification.

(xv) Multi-Cluster Feature Selection (MCFS)

Cai et al in [53] proposed a multi-cluster feature selection (MCFS) method which is capable to select the set of features that can cover all the possible clustering in the data. It uses the spectral analysis to measure the correlation between different features in an unsupervised domain. The top eigenvectors of graph Laplacian and spectral clustering cluster data samples. MCFS use the k-Nearest-Neighbors approach to construct the graph of the data, where $k$ is predetermined. The weighting matrix $W$ is calculated as follows:

$$W_{ij} = e^{\frac{-d_i - d_j}{\sigma}}$$

(12)

where, $d_i$ and $d_j$ are connected data points in the k nearest neighbor graph and $\sigma$ is a predefined parameter. A degree matrix $D$ is computed from which the graph Laplacian $L = D - W$ is computed. The eigen problem $Ly = \lambda Dy$ solved to capture the multi-cluster structure of the data. The relevant subset of features is obtained by minimizing the objective function

$$\min_{a_i} \left\| y_i - X^T a_i \right\|^2$$

(13)

such that $||a_k||_0 = l$. $y_k$ is the solution of the eigen problem, $a_k$ is the m-dimensional vector and $||a_k||_0$ is the number of non-zero elements in $a_k$. Then, K sparse coefficient vectors is chosen to correspond to each cluster. For each feature, $f_i$, the maximum value of $a_k$ that correspond to $f_i$ will be chosen. Finally, MCFS chooses the top L features. It is found to outperform the MaxVar method of feature selection and has
comparable performance with Laplacian Score.

(xvi) Term Frequency-Inverse Document Frequency (TF-Idf)

TF-Idf [41], [54] was proposed with a heuristic perception that query terms occurring across many documents are not good discriminators and should not be considered as discriminative terms. They should be assigned less weight than those occurring across few documents. Term frequency (Tf) represents the frequency of the term occurring in the document whereas Inverse Document Frequency (Idf) gives the inverse measure of the number of documents to which the term is assigned [55]. To express the significance of textual data, it is expressed as the product of Tf and Idf. TF-Idf is given by

\[
w_{ij} = tf_{ij} \times \log \frac{N}{df_{ij}}
\]

(14)

where, \( w_{ij} \) is the weight of the term in document \( j \), \( N \) is the number of documents in the collection, \( tf_{ij} \) is the frequency of the term \( i \) in document \( j \) and \( df_{ij} \) is the frequency of the document containing the term \( i \) in the collection. TF-Idf is an estimate of the relevance of the term to the document [55]. A term occurring in many documents will have low TF-Idf values than those appearing relatively fewer across the documents.

The methods in this section are summarized in Table 1 (at the end of paper) with respect to their performance.

4 EXPERIMENTAL SETUP

Research in the past reveal that the predictive accuracy of feature selection methods are dataset driven. The results summarized in Table 1 are obtained from different text datasets. The merits and demerits of the state of art methods in Table 1 are assessed on different standard or adhoc datasets. In order to evaluate different feature selection methods on a common standard, experiments are performed on three benchmark text datasets - the Reuters-21578, 20 Newsgroups and TDT2 datasets to compare the results in a unifying framework of datasets. In the study, feature selection is evaluated in terms of the performance of k-means clustering and KNN classifier.

4.1 Data Sets

Past research works [35], reveal that text categorization performance varies with different dataset. Therefore, three different text datasets are used to evaluate text clustering and classification performance on three standard datasets: Reuters-21578, 20 Newsgroups and TDT2 datasets. In all the three datasets, multi-label documents are discarded. Dataset properties are described in Table 2 at the end of the paper.

4.2 Evaluation Metrics

The performance of clustering and classification is evaluated with standard measures of accuracy, F-Measure and Normalized Mutual Information [53].

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]

\[
\text{Precision} = \frac{TP}{TP + FP}
\]

\[
\text{Recall} = \frac{TP}{TP + FN}
\]

\[
F - \text{Measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}
\]

\[
TP, FP, TN \text{ and } FN \text{ are the number of true positives, false positives, true negatives and false negatives respectively.}
\]

Normalized Mutual Information \( MI \) is given by:

\[
MI = \frac{MI(C, C')}{\max(H(C), H(C'))}
\]

where, \( C \) and \( C' \) denote the set of clusters obtained from the ground truth and labels after clustering respectively. \( H(C) \) and \( H(C') \) are entropies of \( C \) and \( C' \) respectively. Their mutual information is given by

\[
MI(C, C') = \sum_{C_i \in C, C'_{i}} P(C_i, C'_{i}) \log \frac{P(C_i, C'_{i})}{P(C_i) \cdot P(C'_{i})}
\]

where, \( P(C_i) \) and \( P(C'_{i}) \) are probabilities that documents are selected arbitrarily from the corpus belongs to \( C_i \) and \( C'_{i} \) respectively. \( P(C_i, C'_{i}) \) are joint probabilities that the selected documents belongs to both the clusters at the same time. Normalized Mutual Information ranges from 0 to 1.

4.3 Experiment Results

The experimental results of clustering and classification by sixteen feature selection methods as listed in section 3 over the considered datasets is presented in this section. The parameters for the different methods are set as required by the different methods before the experiment. The parameter variation is studied prior to the experiments so as to conduct the experiments with best parameter value which gives best results. The study of the parameter setting is not included in the paper as the primary objective is to ascertain the predictive accuracy and dimensionality reduction after feature selection. Further, for the sake of simplicity, the methods are listed sequentially in Table 1, which are labeled as FS1 to FS16. Tables 3 to 7 (at the end of paper) report the average results of clustering and classification. It is unrealistic to obtain an ideal optimal case of feature subset, therefore the number of features is chosen in the range of 500 to 1500 empirically. The k-means clustering results with respect to accuracy, F-Measure and NMI over the considered datasets are presented in Table 3 to 5. Clustering result show that MCFS (FS11) outperforms all other methods in terms of all the measures on all the considered datasets. Classification results reveal that best performance is achieved by MCFS (FS11) method as given in tables 6 and 7 for accuracy and F-Measure respectively. TF-Idf (FS1) has a near comparable performance with MCFS (FS11) for all the datasets for both clustering and classification. Feature selection is necessary and effective as it not only increase the efficiency of machine learning algorithm but also reduces the number of features significantly. The
reason for competitive advantage of FS11-the MCFS feature selection method over other methods is its capacity to capture the correlation amongst the features, unlike the other methods that simply rank the features using scores independent of each other. As is observed FS11 has the highest predictive accuracy, the dimensionality reduction produced by it is studied herewith. Table 8 (at the end of paper) shows dimensionality reduction of 96% on Reuters-21578, 99% on 20 NG dataset and 97% on TDT2 dataset with k-means clustering. Table 9 (at the end of paper) shows that Reuters-21578 dataset has a dimensionality reduction of about 92%, while 20NG approximately has a reduction of 99% and TDT2 has dimensionality reduction of 96% with KNN classifier.

5 CONCLUSIONS
The past research reveals that the strengths and weakness of feature selection methods are dataset specific, therefore an attempt has been made to evaluate and compare the performance of sixteen state of art text feature selection methods over a unifying framework of three benchmark datasets to evaluate the performance on the same standards. The efficiency of these methods is evaluated with respect to the predictive accuracy of two learning algorithms-the k-means clustering and the KNN classifier. Although different methods perform differently, it is found that Multi-Cluster Feature Selection (MCFS) performs the best in the given framework of the experimental setup. It reduces the data dimensionality to an extent of 95% on an average on the considered datasets with acceptable results of classification and clustering. Therefore, it can be concluded that Multi-Cluster Feature Selection can be reliably used as a feature selection method for text categorization.
<table>
<thead>
<tr>
<th>No.</th>
<th>Method</th>
<th>Performance Evaluation</th>
<th>Observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>FS2</td>
<td>Document Frequency, (1995) [35]</td>
<td>kNN and Linear Least Square Fit mapping</td>
<td>Lowest computation cost can be reliably used instead of IG or CHI, effectively removes 90% terms and performs comparably with IG and CHI.</td>
</tr>
<tr>
<td>FS3</td>
<td>Term Strength, (1995) [35]</td>
<td>kNN and Linear Least Square Fit mapping</td>
<td>Comparable performance with IG, CHI and DF but its performance compromised at high vocabulary reduction levels.</td>
</tr>
<tr>
<td>FS5</td>
<td>( \chi^2 ) statistics, (1995) [35]</td>
<td>kNN and Linear Least Square Fit mapping</td>
<td>Effectively removes 90% terms without losing accuracy of classification and performs equally with IG and DF.</td>
</tr>
<tr>
<td>FS6</td>
<td>Mutual Information, (1995) [35]</td>
<td>kNN and Linear Least Square Fit mapping</td>
<td>Relatively poor performance compared to DF, TS, IG and CHI due to its bias towards rare terms and sensitivity to probability estimation errors.</td>
</tr>
<tr>
<td>FS7</td>
<td>Entropy Based Ranking, (2003) [38]</td>
<td>k-means</td>
<td>Superior performance to IG, CHI, DF, TS and Term Contribution [38].</td>
</tr>
<tr>
<td>FS8</td>
<td>Feature Selection based on Gini Index, (2007) [51]</td>
<td>SVM, KNN</td>
<td>Better performance and simpler computation than IG, CHI, expected cross entropy and weight of evidence of text [51].</td>
</tr>
<tr>
<td>FS9</td>
<td>Probability based Term Weighting Features Selection, (2007) [41]</td>
<td>SVM and Complement Naive Bayes</td>
<td>The algorithm, compared with the original version [51], demonstrates a significant overall improved performance in comparison with IG, CHI and Odds Ratio in terms of classification [52].</td>
</tr>
<tr>
<td>FS12</td>
<td>Complete Gini-Index Text Feature-Selection (GIT), (2010) [52]</td>
<td>KNN and SVM</td>
<td>The algorithm, compared with the original version [51], demonstrates a significant overall improved performance in comparison with IG, CHI and Odds Ratio in terms of classification [52].</td>
</tr>
<tr>
<td>FS14</td>
<td>Comprehensively Measure Feature Selection, (2012) [47]</td>
<td>Naive Bayes classifier and SVM.</td>
<td>Superior to IG, CHI, DF, DIA [47] and OCFS when Naive Bayes classifier is used. CMFS outperforms IG, DF, OCFS and DIA when SVM is used.</td>
</tr>
</tbody>
</table>
Table 2 Dataset Properties

<table>
<thead>
<tr>
<th>Datasets</th>
<th>No. Class</th>
<th>No. of Instances</th>
<th>No. of Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>REUTERS</td>
<td>80</td>
<td>10733</td>
<td>18484</td>
</tr>
<tr>
<td>20NG</td>
<td>20</td>
<td>18828</td>
<td>91652</td>
</tr>
<tr>
<td>TDT2</td>
<td>30</td>
<td>9394</td>
<td>36771</td>
</tr>
</tbody>
</table>

Table 3 Accuracy results obtained from different feature selection methods with k-means clustering

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FS1</th>
<th>FS2</th>
<th>FS3</th>
<th>FS4</th>
<th>FS5</th>
<th>FS6</th>
<th>FS7</th>
<th>FS8</th>
<th>FS9</th>
<th>FS10</th>
<th>FS11</th>
<th>FS12</th>
<th>FS13</th>
<th>FS14</th>
<th>FS15</th>
<th>FS16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>73.62</td>
<td>67.17</td>
<td>60.56</td>
<td>64.56</td>
<td>67.89</td>
<td>50.34</td>
<td>64.66</td>
<td>68.21</td>
<td>70.39</td>
<td>67.29</td>
<td>75.84</td>
<td>67.79</td>
<td>68.10</td>
<td>69.74</td>
<td>65.41</td>
<td>62.30</td>
</tr>
<tr>
<td>20NG</td>
<td>70.65</td>
<td>66.76</td>
<td>65.81</td>
<td>65.72</td>
<td>69.32</td>
<td>62.41</td>
<td>64.13</td>
<td>67.78</td>
<td>66.17</td>
<td>71.78</td>
<td>74.14</td>
<td>68.64</td>
<td>73.44</td>
<td>73.27</td>
<td>63.41</td>
<td>71.67</td>
</tr>
<tr>
<td>TDT2</td>
<td>72.77</td>
<td>68.75</td>
<td>64.13</td>
<td>67.20</td>
<td>66.47</td>
<td>67.58</td>
<td>66.79</td>
<td>60.16</td>
<td>65.75</td>
<td>64.61</td>
<td>74.77</td>
<td>67.32</td>
<td>64.55</td>
<td>65.39</td>
<td>65.01</td>
<td>64.66</td>
</tr>
</tbody>
</table>

Table 4 F-Measure results obtained from different feature selection methods with k-means clustering

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FS1</th>
<th>FS2</th>
<th>FS3</th>
<th>FS4</th>
<th>FS5</th>
<th>FS6</th>
<th>FS7</th>
<th>FS8</th>
<th>FS9</th>
<th>FS10</th>
<th>FS11</th>
<th>FS12</th>
<th>FS13</th>
<th>FS14</th>
<th>FS15</th>
<th>FS16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>79.66</td>
<td>75.77</td>
<td>62.44</td>
<td>65.87</td>
<td>75.38</td>
<td>63.29</td>
<td>66.46</td>
<td>74.78</td>
<td>74.39</td>
<td>68.33</td>
<td>70.74</td>
<td>70.84</td>
<td>74.22</td>
<td>71.56</td>
<td>67.39</td>
<td>77.81</td>
</tr>
<tr>
<td>20NG</td>
<td>78.19</td>
<td>70.45</td>
<td>65.69</td>
<td>64.57</td>
<td>76.19</td>
<td>60.54</td>
<td>60.48</td>
<td>74.62</td>
<td>73.18</td>
<td>65.76</td>
<td>82.97</td>
<td>71.20</td>
<td>75.68</td>
<td>70.86</td>
<td>68.03</td>
<td>76.52</td>
</tr>
<tr>
<td>TDT2</td>
<td>76.44</td>
<td>74.76</td>
<td>66.81</td>
<td>69.28</td>
<td>65.58</td>
<td>64.01</td>
<td>67.87</td>
<td>69.09</td>
<td>70.86</td>
<td>69.77</td>
<td>80.19</td>
<td>72.31</td>
<td>70.44</td>
<td>69.57</td>
<td>67.68</td>
<td>70.87</td>
</tr>
</tbody>
</table>

Table 5 NMI results obtained from different feature selection methods with k-means clustering

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FS1</th>
<th>FS2</th>
<th>FS3</th>
<th>FS4</th>
<th>FS5</th>
<th>FS6</th>
<th>FS7</th>
<th>FS8</th>
<th>FS9</th>
<th>FS10</th>
<th>FS11</th>
<th>FS12</th>
<th>FS13</th>
<th>FS14</th>
<th>FS15</th>
<th>FS16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>78.39</td>
<td>77.21</td>
<td>64.42</td>
<td>67.68</td>
<td>73.52</td>
<td>62.26</td>
<td>68.29</td>
<td>75.62</td>
<td>75.22</td>
<td>69.78</td>
<td>80.47</td>
<td>72.28</td>
<td>75.23</td>
<td>72.19</td>
<td>70.34</td>
<td>75.66</td>
</tr>
<tr>
<td>20NG</td>
<td>75.38</td>
<td>71.11</td>
<td>67.78</td>
<td>68.87</td>
<td>75.20</td>
<td>64.63</td>
<td>66.76</td>
<td>75.29</td>
<td>75.73</td>
<td>68.61</td>
<td>80.49</td>
<td>73.58</td>
<td>78.36</td>
<td>72.43</td>
<td>71.59</td>
<td>76.22</td>
</tr>
<tr>
<td>TDT2</td>
<td>74.38</td>
<td>72.40</td>
<td>68.84</td>
<td>70.09</td>
<td>67.68</td>
<td>64.33</td>
<td>68.54</td>
<td>70.81</td>
<td>72.08</td>
<td>72.36</td>
<td>78.54</td>
<td>70.04</td>
<td>71.77</td>
<td>70.61</td>
<td>72.60</td>
<td>72.24</td>
</tr>
</tbody>
</table>

Table 6 Accuracy results obtained from different feature selection methods with KNN classifier

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FS1</th>
<th>FS2</th>
<th>FS3</th>
<th>FS4</th>
<th>FS5</th>
<th>FS6</th>
<th>FS7</th>
<th>FS8</th>
<th>FS9</th>
<th>FS10</th>
<th>FS11</th>
<th>FS12</th>
<th>FS13</th>
<th>FS14</th>
<th>FS15</th>
<th>FS16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>500</td>
<td>600</td>
<td>700</td>
<td>800</td>
<td>900</td>
<td>1000</td>
<td>1100</td>
<td>1200</td>
<td>1300</td>
<td>1400</td>
<td>1500</td>
<td>65.51</td>
<td>70.22</td>
<td>77.71</td>
<td>81.29</td>
<td>81.18</td>
</tr>
<tr>
<td>20NG</td>
<td>70.45</td>
<td>75.71</td>
<td>82.20</td>
<td>82.31</td>
<td>82.02</td>
<td>82.67</td>
<td>82.43</td>
<td>82.39</td>
<td>82.19</td>
<td>82.35</td>
<td>82.66</td>
<td>68.18</td>
<td>70.56</td>
<td>71.21</td>
<td>71.67</td>
<td>72.09</td>
</tr>
</tbody>
</table>

Table 7 F-Measure results obtained from different feature selection methods with KNN classifier

<table>
<thead>
<tr>
<th>Dataset</th>
<th>FS1</th>
<th>FS2</th>
<th>FS3</th>
<th>FS4</th>
<th>FS5</th>
<th>FS6</th>
<th>FS7</th>
<th>FS8</th>
<th>FS9</th>
<th>FS10</th>
<th>FS11</th>
<th>FS12</th>
<th>FS13</th>
<th>FS14</th>
<th>FS15</th>
<th>FS16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reuters</td>
<td>500</td>
<td>600</td>
<td>700</td>
<td>800</td>
<td>900</td>
<td>1000</td>
<td>1100</td>
<td>1200</td>
<td>1300</td>
<td>1400</td>
<td>1500</td>
<td>66.87</td>
<td>72.45</td>
<td>74.87</td>
<td>78.79</td>
<td>82.76</td>
</tr>
<tr>
<td>20NG</td>
<td>67.13</td>
<td>70.67</td>
<td>72.54</td>
<td>75.78</td>
<td>76.66</td>
<td>77.22</td>
<td>78.31</td>
<td>82.63</td>
<td>82.74</td>
<td>82.96</td>
<td>83.02</td>
<td>66.87</td>
<td>72.45</td>
<td>74.87</td>
<td>78.79</td>
<td>82.76</td>
</tr>
<tr>
<td>TDT2</td>
<td>70.09</td>
<td>72.87</td>
<td>75.71</td>
<td>76.54</td>
<td>77.68</td>
<td>78.05</td>
<td>79.64</td>
<td>80.18</td>
<td>80.29</td>
<td>80.46</td>
<td>80.69</td>
<td>70.09</td>
<td>72.87</td>
<td>75.71</td>
<td>76.54</td>
<td>77.68</td>
</tr>
</tbody>
</table>

REFERENCES


Boston, MA, USA: Addison-Wesley.

[41] G. Salton and C. Buckley, “Term weighting approaches in automatic


