Engineering Applications of Artificial Intelligence 70 (2018) 25–37

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

A novel multivariate filter method for feature selection in text classification problems

Mahdieh Labani [a], Parham Moradi [a,*], Fardin Ahmadizar [b], Mahdi Jalili [c]

[a] Department of Computer Engineering, University of Kurdistan, Sanandaj, Iran
[b] Department of Industrial Engineering, University of Kurdistan, Sanandaj, Iran
[c] School of Engineering, RMIT University, Melbourne, Australia

ARTICLE INFO

Keywords:
Text classification
Feature selection
Dimensionality reduction
Filter approach
Multivariate analysis

ABSTRACT

With increasing number of documents in digital format, automatic text categorization has become a crucial task in pattern recognition problems. To ease the classification task, feature selection methods have been introduced to reduce the dimensionality of the feature space, and thus improve the classification performance. In this paper a novel filter method for feature selection, called Multivariate Relative Discrimination Criterion (MRDC), is proposed for text classification. The proposed method focuses on the reduction of redundant features using minimal-redundancy and maximal-relevancy concepts. To this end, the proposed method takes into account document frequencies for each term, while estimating their usefulness. The proposed method not only selects the features with maximum relevancy, but also the redundancy between them is takes into account using a correlation metric. MRDC does not employ any learning algorithm to evaluate the usefulness of the selected features, and thus it can be categorized as a filter method. In order to assess the effectiveness of the proposed method, several experiments are performed on three real-world datasets. The obtained results are compared to the state-of-the-art filter methods. The reported results show that in most cases MRDC results in better classification performance than others.

1. Introduction

With increasing growth of the Internet and information technologies, the massive volume of electronic text documents are given through web pages, the news feeds, electronic emails and digital libraries. To handle such massive information, text categorization has become a key technology to discover and classify text documents. Text categorization is defined as a task of automatically classifying unlabeled documents into predefined categories (Adeva and Atxa, 2007). It has been successfully developed in many applications such as topic detection (Zeng and Zhang, 2007), spam e-mail filtering (Guzella and Caminhas, 2009), SMS spam filtering (Idris and Selamat, 2014), author identification (Zhang et al., 2014), Bioinformatics (Saeys et al., 2007, Tabakhi et al., 2015), web page classification (Özel, 2011), document classification (Jiang et al., 2016) and sentiment analysis (Medhat et al., 2014). In the process of text categorization, documents are generally modeled as a vector space, in which each word is considered as a feature. In the vector model of a document, the value of a feature can be its corresponding word's frequency or term frequency-inverse document frequency (tf-idf). One of the most important issues in the text categorization is to deal with high dimensionality of the feature space. Excessive number of features not only increases the computational time, but also degrades the classification accuracy (Shang et al., 2013). Feature selection and extraction are two main approaches for reducing the dimensionality of the text feature space (Bharti and Singh, 2015). The feature extraction refers to the process of generating a small set of new features by combining or transforming the original ones (Agarwal and Mittal, 2014), while in the feature selection the dimension of the space is reduced by selecting the most prominent features (Saleh and El-Sonbaty, 2007).

Feature selection methods can be classified into four categories: filter, wrapper, embedded and hybrid approaches. Filter methods perform a statistical analysis over the features space to select a discriminative subset of features. In the wrapper approach, various subsets of features are first identified, and then evaluated using classifiers (Agarwal and Mittal, 2014). The hybrid approach takes advantages of both filter and wrapper approaches, and in the embedded approach the feature selection process is embedded into the training phase of the classification

[*] Corresponding author.
E-mail addresses: m.labani@eng.uok.ac.ir (M. Labani), p.moradi@uok.ac.ir (P. Moradi), f.ahmadizar@uok.ac.ir (F. Ahmadizar), mahdi.jalili@rmit.edu.au (M. Jalili).

https://doi.org/10.1016/j.engappai.2017.12.014
Received 31 May 2016; Received in revised form 23 November 2016; Accepted 28 December 2017

process (Chouaib et al., 2008). Due to the use of a learning model in the selection process of wrapper, hybrid and embedded approaches, they have an advantage of achieving higher accuracy compared to the filter approach, while being more computationally expensive. The filter approach is often fast and its results are not biased to the choice of classifiers, and thus are widely used to reduce the dimensionality, especially for large-scale feature spaces (Günal, 2012).

In general, the filter approach can be classified into univariate and multivariate methods. In the univariate methods, a specific criterion is used to evaluate the relevance of features independently. Although, these methods can effectively identify irrelevant features, they are unable to remove redundant ones. In other words, univariate filter methods only evaluate features individually, and completely ignore the redundancy between them. On the other hand, multivariate methods consider correlation between features in their process, and thus can handle both irrelevant and redundant features. Although the performance of multivariate methods is better than the univariate ones, they are computationally inefficient.

Relative discrimination criterion (RDC) is an effective univariate filter criterion, which has been recently proposed to reduce the dimensionality of text data (Rehman et al., 2015). In this method, it is assumed that the terms that frequently occur in a specific class compared to others have much higher discriminative properties, and thus are assigned high scores. Although RDC is an effective method for identifying relevant features, the correlation between features is ignored in its evaluation process, and thus it cannot identify redundant features. There often remarkable number of correlated features in the text data identifying which leads to enhance the quality of text classifiers. On the other hand, the aim of feature selection is to select a compact feature subset with maximal discriminative capability, which requires having a high relevance to class labels and low redundancy within the selected feature subset. To reach this goal, in this paper, a novel multivariate feature selection method, called Multivariate Relative Discrimination Criterion (MRDC), is proposed to consider both relevancy and redundancy concepts in its evaluation process. To this end, the proposed method first computes the relevancy of each feature using RDC measure, and then Pearson correlation is used to compute correlation values between features. This results in avoiding higher correlated features. Several experiments are performed on three real-world datasets including Reuters-21,578, 20-Newsgroups and WebKB to evaluate the performance of the proposed method. The reported results reveals that MRDC performs much better compared to state-of-the-art filter methods.

The rest of the paper is organized as follows. Section 2 gives a brief review of previous works. Section 3 presents the details of proposed MRDC method and experimental results are reported and discussed in Section 4. Finally, Section 5 concludes the paper.

2. Literature review and background

2.1. Overview of feature selection methods for text classification

Text categorization is to assign documents to one or more classes. Manual text classification is time-consuming, especially for large-scale dataset; therefore automatic text classification methods have been increasingly used in various applications (Perikos and Hatzilygeroudis, 2016). A text document is a collection of words arranged according to their corresponding language grammatical rules. Although arrangement of words is necessary for constructing meaningful sentences, a text document is usually represented as a "bag of words" for text classifiers, where the order of the words is not considered in the classification process (Badawi and Altınçay, 2014). Therefore, a document d_j is represented as a vector $d_j = \left\{tw_{1j}, tw_{2j}, \dots, tw_{vj}\right\}$ where tw_{ij} shows the weight of ith term from a vocabulary of words $T = \left\{t_1, t_2, \dots, t_v\right\}$. A general method to weight the terms in documents is $tf.idf$, where $tf(t,d)$ and $idf(t,d)$ are the term frequency and the inverse document frequency of term t in document d, respectively (Erenel and Altınçay, 2012). The term frequency tf is the term count normalized by the document size, while idf is defined as $log\left(N/df\right)$, where the document frequency df is the number of documents containing a specific term.

Text classification tasks often involve thousands of features, and the classification is indeed a high dimensional problem. Although there are tens of thousands of words in a typical text collection, most of them contain little or no information to predict the text label. The relevance of a feature indicates that the feature is always necessary for predicting the class label, and feature redundancy is usually defined in terms of some kind of correlations in the features. The goal of feature selection is to select a highly-relevant subset with minimum redundancy. To this end, dimensionality reduction approaches (such as feature extraction or selection) is curtail not only to improve the classifier's prediction performance, but also to reduce storage requirements.

Feature extraction methods can be used to reduce the size of feature vector by transforming a higher dimensional feature space to a lower dimension. There are a number of feature extraction methods to reduce the dimensionality of text documents (Agarwal and Mittal, 2014). For example, in Li and Park (2007) a Singular Value Decomposition (SVD) based method was used to learn and represent relations among large numbers of words and natural text documents including these words. This method did not take into account the semantic relationship between the terms, resulting in rather poor performance. In Kolenda et al. (2000) an Independent Component Analysis (ICA) was employed to find k components that effectively contain maximum variability of the original data. This method transforms the original high dimensional data into lower dimensional components that are maximally independent from each other. In another research, Linear Discriminant Analysis (LDA) was used to transform the original high dimensional text data into a lower dimension (Wang and Qian, 2008).

Compared to feature extraction methods, there are varieties of text feature selection methods in the literature, each being filter, wrapper, hybrid or embedded methods. Filter methods require a statistical analysis on a feature set without utilizing any learning algorithm, and are the prime choice in many cases due to much lower computational complexity than others. Filter methods can be implemented as univariate or multivariate fashions (Hu et al., 2015). Many univariate methods have been proposed in the literature. Examples include Document frequency (DF) (Liu et al., 2005), Term variance (TV) (Liu et al., 2005), Term strength (TS) (Yang, 1995), Information gain (IG) (Liu et al., 2005), Chi-square (CHI) (Li et al., 2008), Odds ratio (OR) (Mengle and Goharian, 2009), Gini index (GI) (Shang et al., 2007), Improved Gini index (GINI) (Mengle and Goharian, 2009), distinguishing feature selector (DFS) (Yang, 1995), bi-normal separation (BNS) (Forman, 2003), mutual information (Xu et al., 2007) and relative discrimination criterion (RDC) (Rehman et al., 2015). In DF the number of documents containing a specific term is considered in its evaluation process. In TV it is assumed that features with higher variance values contain valuable information (Liu et al., 2005). TS measures a term's importance based on how commonly the term is likely to appear in similar documents (Yang, 1995). The OR method evaluates the ratio of odds occurring in positive classes to its odds in negative classes (Mengle and Goharian, 2009). In DFS the contributions of terms to the class discrimination is first estimated using a probabilistic approach, and then certain importance scores are assigned to them (Yang, 1995). In BNS the occurrence of a given term in each document is modeled as a normal distribution and its corresponding area under the curve that exceeds a threshold value is considered as the importance of that term (Forman, 2003). In DP the degree of deviation from the Poisson distribution is used to evaluate the importance of the terms (Ogura et al., 2009). On the other hand, to avoid the effects of unbalanced classes, the GINI method considers the term's condition probability and combines the posterior probability and conditional probability in its evaluation process (Shang et al., 2007). All these methods are univariate methods that do not consider the dependency between features, and thus are unable to remove redundant features. While, in multivariate methods the dependencies between features are also taken into consideration in their evaluation process, thus, being able to select a set of higher quality features. mRMR (Saleh and El-Sonbaty, 2007) is a well-known multivariate method which has been successfully applied to text feature selection.

The wrapper methods are another category of feature selection methods, which use a specific search method to find a subset of features around the feature space. In these methods, feature dependencies and interaction between feature subsets are considered in the search process, resulting in often high quality classification. These methods employ a learning model and are more computationally expensive compared to filter approach. Finally, the hybrid feature selection approach is a combination of filter and wrapper methods.

2.2. Filter methods for feature selection

In this section, well-known filter methods are discussed including IG, GI, mRMR and RDC. These methods are compared with the proposed technique.

2.2.1. Information gain (IG)

Information gain (IG) of a term, measures show its entropy is reduced when documents are separated into different classes considering the presence of that term (Yang and Pedersen, 1997). IG is defined as follows:

$$IG(t) = -\sum_{i=1}^{M} P(C_i)\log P(C_i) + P(t)\sum_{i=1}^{M} P(C_i|t)\log P(C_i|t) + P(\dot{t})\sum_{i=1}^{M} P(C_i|\dot{t})\log P(C_i|\dot{t}) \tag{1}$$

where M is the number of classes, $P\left(C_i\right)$ is the probability of class C_i, $P(t)$ and $P\left(\dot{t}\right)$ are the probabilities of presence and absence of term t, respectively. $P\left(C_i|t\right)$ and $P\left(C_i|\dot{t}\right)$ are the conditional probabilities of class C_i considering presence and absence of t, respectively.

2.2.2. Gini index (GI)

Gini index (GI) is a global feature selection method for text classification, which can be considered as an improved version of the attribute selection method used in construction of decision tree (Shang et al., 2007). This measure is defined as follows:

$$GI(t) = \sum_{i=1}^{M} P(t|C_i)^2 P(C_i|t)^2 \tag{2}$$

where $P(t|C_i)$ is the probability of term t given class C_i and $P\left(C_i|t\right)$ is the probability of C_i in the presence of t.

2.2.3. Minimal-redundancy-maximal-relevance (mRMR)

Minimal-redundancy-maximal-relevance (mRMR) is a multivariate filter method, which seeks to select features with the highest dependency with the target class by using a specific relevance criterion. A criterion is used to reduce the redundancy between features, and is defined as follows:

$$mRMR(F_j) = \max_{F_j \in F\setminus S}\left[I\left(F_j;C_k\right) - \frac{1}{m-1}\sum_{F_i \epsilon S} I\left(F_j;F_i\right)\right] \tag{3}$$

where F is set of all features, $I(F_j;C_k)$ is the mutual correlation between feature X_j and class C_k and $I(F_j;F_i)$ is the mutual correlation between features F_i and F_j. S is the selected feature set and m denotes the size of S (i.e., $m = |S|$).

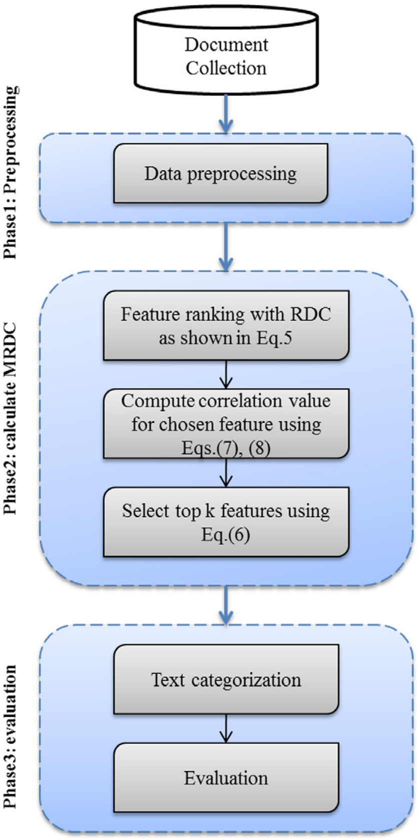

Document Collection
Phase1: Preprocessing
Data preprocessing
Phase2: calculate MRDC
Feature ranking with RDC as shown in Eq.5
Compute correlation value for chosen feature using Eqs.(7), (8)
Select top k features using Eq.(6)
Phase3: evaluation
Text categorization
Evaluation

Fig. 1. The framework of the proposed feature selection method.

2.2.4. Relative discriminative criterion (RDC)

Relative discriminative criterion (RDC) is based on difference between document frequencies related to counts of a given term in the positive and negative classes. The selection criterion used in RDC is as follows:

$$RDC(w_i,\ tc_j(w_i)) = \frac{\left(\left|df_{pos}(w_i) - df_{neg}(w_i)\right|\right)}{\min\left(df_{pos}(w_i), df_{neg}(w_i)\right) \times tc_j(w_i)} \tag{4}$$

where $df_{pos}(w_i)$ and $df_{neg}(w_i)$ are respectively the number of positive and negative documents in which term w_i is occurred. The word w_i may repeat in a specific document several times. The number of times that w_i appears is recorded in an array $tc\left(w_i\right) = \left[tc_1\left(w_i\right), tc_2\left(w_i\right), \dots, tc_m\left(w_i\right)\right]$. In this approach, instead of summing up RDC values for different term counts, the Area Under the Curve (AUC) is computed according to the following equations:

$$AUC(w_i, tc_j) = AUC(w_i, tc_{j-1}) + \frac{RDC(w_i, tc_j) + RDC(w_i, tc_{j+1})}{2} \tag{5}$$

$$RDC(w_i) = AUC(w_i, tc_m) \tag{6}$$

where $AUC(w_i, tc_j)$ is AUC of term w_i and term count tc_j. It should be noted that $AUC\left(w_i, tc_1\right) = 0$. The final RDC score for term w_i is assigned to the last AUC of tc_m.

3. Proposed method

In this section, a novel supervised feature selection method, called Multivariate Relative Discriminative Criterion (MRDC), is proposed. The proposed method is specifically designed for text classification tasks. It consists of three steps: (1) pre-processing, (2) selection and (3) evaluation. In the first step, several pre-processing methods such

Algorithm1. Multivariate Relative Discriminative Criterion (MRDC) algorithm

```
Input    k:The number of feature in the final subset
         C: Class labels of documents
         D:Vectorized input document

Output   S: Selected features

Begin algorithm
    $S \leftarrow \{\}$, $F$ = All features of D
    For each $f_s$ in $F$
        $Relevance(f_s)$=Compute RDC value using Eq. (6) (relevancy value of $f_s$)
    End for
    Sort features according to their corresponding relevancy values
    $f_{max}$ = Select the feature with maximum relevancy value
    Add $f_{max}$ to S ( $S \leftarrow S \cup \{f_{max}\}$)
    Remove $f_{max}$ from F ( $F \leftarrow F \backslash \{f_{max}\}$)
    max_index=0;
    For each $f_i$ in $F$
        sum($f_i$)=0;
    End for
    While ($|S| <= k$)
        For each $f_i$ in $F$
            For each $f_s$ in $S$
                $Correlation(f_i, f_s)$=Compute correlation value using Eqs. (8) and (9)
                $sum(f_i) = sum(f_i) + Correlation(f_i, f_s)$;
            End for
        End for
        max= - inf (e.g. a large negative value);
        For each $f_i$ in $F$
            MRDC($f_i$) = $Relevance(f_i) - sum(f_i)$ ; (Compute MRDC value using Eq. (7))
        End for
        $f_{max}$ = Select the feature with maximum MRDC value
        Add$f_{max}$ to S ( $S \leftarrow S \cup \{f_{max}\}$)
        Remove $f_{max}$ from F ( $F \leftarrow F \backslash \{f_{max}\}$)
    End while
    Return S as final selected features.
End algorithm
```

Fig. 2. The pseudo-code of the proposed algorithm.

as removing stop-words, stemming, pruning and term weighting are applied on the input documents to convert them into a suitable representation for classification algorithms. In the second step, we propose a multivariate feature ranking criterion to evaluate features for text classification problems. To consider both relevancy and redundancy of features in the evaluation process, first RDC is employed to calculate a relevancy value for each feature, and then Pearson correlation is used to compute redundancy values between features. Finally, in the third step the selected subset of features are evaluated using a supervised learning algorithm. These steps are also depicted in Fig. 1.

3.1. Preprocessing

Vector space (or bag-of-words) model is a commonly used model to represent documents in text classification problems. In this model, a document is represented as a bag of its words, without paying attention to its grammar and even word order. Also, the frequency of words is used as feature values for training a classifier. However, the high number of features generated in this model leads to use some pre-processing tasks to reduce the high dimensionality of the terms space. The most common pre-processing task for text classification is removing stop-words and stemming. In the stop-word removal process, the common words that do not have discriminative information are removed from the feature space. For example ‘a’, ‘the’, and ‘that’ are frequent words occurring almost equally in all documents and do not carry any useful information for class prediction. The stemming process leaves out root forms of the words. Thereby, different words sharing the same root due to their affixes, can be determined as the same term in the feature space. For example, the terms “computer”, “computing”, “computation” and “computes” have the same meaning as their root “compute”. In this study the Porter’s stemmer is applied to this purpose (Porter, 1980).

Although, both the stemming and removing stop-words processes lead to reduce dimensionality of text documents, there exist rare words that occur in only few documents. Terms with low document frequency commonly introduce noise when they are considered. In (Salton et al., 1975), the authors show that if document frequency of a term is in the range of $\left[\frac{m}{100}, \frac{m}{10}\right]$ (where m is the number of documents), it is a poor term for text classification and does not carry meaningful information for predicting class labels; otherwise it is called a discriminant term. In another research, the authors of (Miller and Newman, 1958) illustrated that word frequencies typically follow a Zipf distribution, where the frequency of each term’s occurrence is proportional to $1/k^p$, where k is its rank among terms sorted by frequency, and p is a fitting factor close to 1. They showed that about half of the total number of distinct words may occur only a single time, thus removing terms under a given threshold of occurrence yields significant savings. Thus, in this paper a pruning procedure is applied to remove such rare terms with corresponding document frequencies lower than a certain threshold value. In the experiments, words that are present in three or fewer documents are removed. Finally, after pre-processing step, the remaining words are first mapped into a vector space model, and then binary weighting method is used to assign a binary value to these terms. The binary weighting method is a simplest way to weight terms, in which if ith term appears in kth document, weight 1 is considered for this term, otherwise its corresponding weight is set to 0. This method have been frequently used by previous researches (Jiang et al., 2007; Lan et al., 2009; Nowak et al., 2006; Rehman et al., 2015) and simplifies the computation of correlations between features.

3.2. Calculating MRDC

The aim of this step is to evaluate features using the proposed MRDC algorithm. A pseudo code of the proposed algorithm is presented in Fig. 2. MRDC evaluates features in two major steps. In the first step, features are evaluated using traditional RDC criterion, and in the second step the redundancy between them is taken into account to select a final set of features. The aim of the first step is to select the maximum relevant features, while the correlation between features is takes into account in the second step.

In this step of the proposed algorithm, first features are sorted in decreasing order based on their relevance values, and then a feature with the maximum relevancy is identified and added to the final selected subset (denoted by S). Next, a feature with the lowest correlation with S is added to S. In other words, in each step, the correlation between the non-selected features with the selected ones is calculated, and the feature with the highest relevancy and lowest correlation is selected. This process is continued until the size of selected features (S) reaches k. To this end, we propose the following equation to measure this concept:

$$MRDC(f_i) = RDC\left(f_i\right) - \sum_{f_i \neq f_j, f_j \in S} correlation\left(f_i, f_j\right) \tag{7}$$

where $RDC\left(f_i\right)$ is the relevance value of feature f_i and $correlation$ (f_i, f_s) denotes the correlation between two features f_i and f_j that is defined by their similarity value. Here we use Pearson correlation coefficient to compute the correlation value:

$$correlation(f_i, f_j) = \left| \frac{\sum_{d\in|docs|}\left(f_{i,d} - \bar{f_i}\right)\left(f_{j,d} - \bar{f_j}\right)}{\sqrt{\sum_{d\in|docs|}\left(f_{i,d} - \bar{f_i}\right)^2}\ \sqrt{\sum_{d\in|docs|}\left(f_{j,d} - \bar{f_j}\right)^2}} \right| \tag{8}$$

where $\bar{f}_i$ and $\bar{f}_j$ are mean values of f_i and f_j vectors, respectively. $f_{i,d}$ and $f_{j,d}$ are respectively the values of features i and j for dth document. The value of 1 means a perfect positive correlation, while the value of -1 denotes perfect negative correlation. In some cases, the similarity values between features in datasets are negative values, which may cause some problems in the computation of the second part of MRDC criterion Eq. (7). To overcome this situation, the following equation is used to rescale the range values from [−1 1] to [0 1]:

$$normalize(x_i) = \frac{x_i - x_{i,\min}}{x_{i.\max} - x_{i,\min}} \tag{9}$$

where $x_{i,max}, x_{i,min}$ are maximum and minimum values of x_i, respectively.

3.2.1. An illustrative example

In this section an example is provided to show that the proposed MRDC measure performs better than the traditional RDC. To this end, a simple synthetic dataset is provided in Table 1. This dataset consists of twelve documents containing four terms including “Cat”, “Dog”, “Mouse” and “Fish”. Each document in this dataset belongs to the positive or negative categories. Table 2 shows the matrix form (i.e., vector model) of this dataset. First, the document frequency df metric is used to weight the terms. To show the effectiveness of MRDC measure, one of the features (f_2) is repeated again and added to the dataset as a new feature (f_5). Therefore, features f_i and f_j are completely correlated. The aim of feature selection is to select the highly relevant features with the lowest possible correlation. f_2 and f_5 contain the same information, and one of them is redundant. Criteria such as MRDC, which take into account the correlation values, assign high value to one of the features, while the other redundant features are given low values. The following computations show that RDC criterion assigns the same value for both f_2 and f_5 features; they are ranked differently by MRDC.

$$RDC\left(f_1\right) = (2+5)/2 + (5+0)/2 = 6$$
$$RDC\left(f_2\right) = RDC\left(f_5\right) = (1+0.5)/2 + (0.5+0)/2 = 1$$
$$RDC\left(f_3\right) = (0+5)/2 + (5+0)/2 = 5$$
$$RDC\left(f_4\right) = (20+5)/2 + (5+0)/2 = 15$$

These computations are repeated based on the proposed MRDC measure and the corresponding results are shown as follows, indicating that MRDC assigns different values for f_2 and f_5.

Table 1
A simple synthetic dataset.

Document	Class	Content
Doc1	Positive	Cat fish
Doc2	Positive	Cat mouse fish
Doc3	Positive	Mouse fish
Doc4	Positive	Mouse cat fish mouse fish
Doc5	Positive	Fish cat fish cat
Doc6	Positive	Fish mouse
Doc7	Negative	Dog mouse
Doc8	Negative	Dog dog
Doc9	Negative	Fish fish mouse
Doc10	Negative	Mouse
Doc11	Negative	Cat fish
Doc12	Negative	Dog fish

Table 2
Document frequency (df) values of terms in the synthetic dataset.

Document	f_1(cat)	f_2(fish)	f_3(mouse)	f_4(dog)	f_5(fish)
Doc1	1	1	0	0	1
Doc2	1	1	1	0	1
Doc3	0	1	1	0	1
Doc4	1	2	2	0	2
Doc5	2	2	0	0	2
Doc6	0	1	1	0	1
Doc7	0	0	1	1	0
Doc8	0	0	0	2	0
Doc9	0	2	1	0	2
Doc10	0	0	1	0	0
Doc11	1	1	0	0	1
Doc12	0	1	0	1	1

Table 3
Comparison of importance values assigned by RDC and MRDC methods for the terms of the synthetic dataset.

Method	f_1(cat)	f_2(fish)	f_3(mouse)	f_4(dog)	f_5(fish)
RDC	6	1	5	15	1
MRDC	5.902	−0.193	4.63	15	−0.84

$$MRDC\left(f_4\right) = 15$$
$$MRDC\left(f_1\right) = 6 - (0.0976) = 5.9024$$
$$MRDC\left(f_3\right) = 5 - (0.1339 + 0.2297) = 4.6364$$
$$MRDC\left(f_2\right) = 1 - (0.0976 + 0.7114 + 0.4824) = -0.1938$$
$$MRDC\left(f_5\right) = 1 - (0.2297 + 0.4824 + 1 + 0.1339) = -0.8460$$

According to Eq. (7), the final MRDC score of a feature is related to both relevancy (first term of Eq. (7)) and redundancy (second term of Eq. (7)) concepts. When two features f_i and f_j are very similar, their correlation would be close to 1. Therefore when f_i is selected before f_j, it is assigned with a lower redundancy value than f_j.

Table 3 compares RDC and MRDC values which have been assigned to the features. It can be seen that RDC assign the same score to both f_2 and f_5, while these features have been assigned different MRDC scores. In this example, f_2 and f_5 features are completely the same, but f_5 was assigned by lower MRDC value than f_2, whereas they are assigned by equal RDC values. Using MRDC values, the selection or rejection of f_2 and f_5 depends on the threshold k (predefined number of selected features). If $k = 3$, both f_2 and f_5 are rejected, while for $k = 4$, f_2 is selected but f_5 is rejected, and if $k = 5$, both f_2 and f_5 are selected.

3.3. Complexity analysis

The computational time of MRDC algorithm contains mainly three parts: (1) Relevancy computation with RDC algorithm, (2) Redundancy computation with Pearson correlation, and (3) Selection with MRDC. For the first part, the relevance value of each feature is computed using

Table 4
Characteristics of WebKB, 20-Newsgroup and Reuters-21578 datasets.

Dataset	Total docs	Number of classes	Min class size	Max class size
WebKB	2803	4	336	1097
20-Newsgroups	18828	20	628	998
Reuters-21578	1339	8	41	200

Table 5
Comparison results of MRDC and RDC methods in terms of precision, recall and F-measures when 200 number of features are selected and MLP classifier is used.

Dataset	Method	Precision	Recall	F-measure
WebKB	MRDC	0.530	0.530	0.530
	RDC	0.427	0.422	0.424
20 Newsgroups	MRDC	0.296	0.262	0.277
	RDC	0.290	0.260	0.274
Reuters-21,578	MRDC	0.392	0.384	0.387
	RDC	0.310	0.344	0.326

Eq. (6), which requires to obtain AUC and RDC for each feature and its different term counts using Eq. (5) and Eq. (4), respectively. The complexity of Eq. (4) is $O\left(tc_{max}\,|F|\,(|docs| + |docs|\,|F|)\right)$ that is reduced to $O(tc_{max}|F|^2\,|docs|)$ where $|F|$ is the number of features, $|docs|$ show the number of documents and tc_{max} is the size of the largest term count array. The complexity of Eq. (5) is $O(tc_{mx}\,|F|)$. Thus, the complexity of the first part of the algorithm is $O(tc_{mx}\,|F| + tc_{max}|F|^2\,|docs|)$ that is reduced to $O(tc_{max}|F|^2\,|docs|)$. The aim of the second part is to compute MRDC value for each feature using Eq. (7), which requires to obtain the correlation values between the features using Eq. (8). Thus, the complexity of this part is $O\left(|F|^2\,|docs|\right)$. In the last part, k-top features according to their corresponding MRDC value are selected as the final subset of features, with complexity of $O(k\,|F|)$. Therefore, the overall complexity of these three parts is $O\left(tc_{max}|F|^2\,|docs| + |F|^2\,|docs| + k\,|F|\right)$ that is reduced to $O(tc_{max}|F|^2\,|docs|)$.

4. Experiments

In this section, a set of experiments are performed to compare the performance of the proposed method(MRDC) with a number of state-of-the-art filter methods including IG, GI, MRMR and RDC. The experiments are performed on a number of datasets including Reuters, WebKB and 20-Newsgroups. All experiments are performed in windows 7 environment on a PC computer having core i5 processor and 8 GB RAM. All methods are implemented using Java programming language. We use Weka framework (Hall et al., 2009) for implementing Multinomial naïve Bayes (MNB), Decision Tree (DT) and Multilayer Perceptron (MLP) classifiers.

4.1. Datasets

The 20-Newsgroups dataset consists of 20,000 documents which have been collected from 20 different newsgroups. Reuters-21,578 corpus is a set of economic news taken from the Reuters news agency. The original version of this dataset contains 21,578 news documents which are organized in 135 categories. WebKB consists of web pages from Computer Science departments of some universities collected by the World Wide Knowledge Base project in 1997. These pages are classified into four categories as course, faculty, project and student. Additional detailed information about these datasets is given in Table 4.

Table 6
Comparison results of MRDC and RDC methods in terms of precision, recall and F-measures when 1500 number of features are selected and MLP classifier is used.

Dataset	Method	Precision	Recall	F-measure
WebKB	MRDC	0.660	0.635	0.647
	RDC	0.512	0.516	0.513
20 Newsgroups	MRDC	0.501	0.664	0.571
	RDC	0.500	0.410	0.450
Reuters-21,578	MRDC	0.770	0.740	0.754
	RDC	0.650	0.644	0.646

4.2. Evaluation measure

Often, Precision (P), Recall (R) and F-measure (F) are used to evaluate the performance of text categorization methods. Precision is the number of correct positive predictions divided by the numbers of positive predictions. Recall is the fraction of the number of documents classified correctly to the number of positive documents and F-measure is combination of the precision and recall measures. The formal definitions of these measures for a given class label i are presented in the following equations:

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{10}$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{11}$$

$$F_i = \frac{2 \times P_i \times R_i}{P_i + R_i} \tag{12}$$

where TP_i is the number of documents which are classified correctly for ith class (true positives), FP_i is the number of documents that are incorrectly labeled to ith class (false positive), and FN_i is the number of documents that belong to ith class, but are incorrectly classified to a negative class. To compute the precession, recall, and F-measure over all classes, one should obtain the average as:

$$P = \frac{\sum_{i=1}^{k} P_i}{k} \tag{13}$$

$$R = \frac{\sum_{i=1}^{k} R_i}{k} \tag{14}$$

$$F = \frac{2 \times P \times R}{P + R} \tag{15}$$

where k is the number of classes.

4.3. Classifiers

The experiments are performed using three well-known classification algorithms including Multilayer Perceptron (MLP), Multinomial naive Bayes (MNB) and Decision Tree (DT). MLP is a feed-forward artificial neural network, consisting of an input layer, one or more hidden layers and an output layer (Myllymäki and Tirri, 1993). MNB is a probabilistic classifier which assumes that the input features are independent from each other given the target class. To implement these classifiers we use Weka. DT creates a pattern that predicts the value of a target variable by training data, and learns simple decision rules deducted from data (Kim et al., 2001). The classifiers contain several adjustable parameters. For DT, the confidence factor, which is used for pruning the tree, is set to 0.25 and the minimum number of instances per leaf is set to 2. In MLP the learning rate and the momentum are set to 0.3 and 0.2, respectively. For other parameters, their corresponding values are set to the default values of Weka.

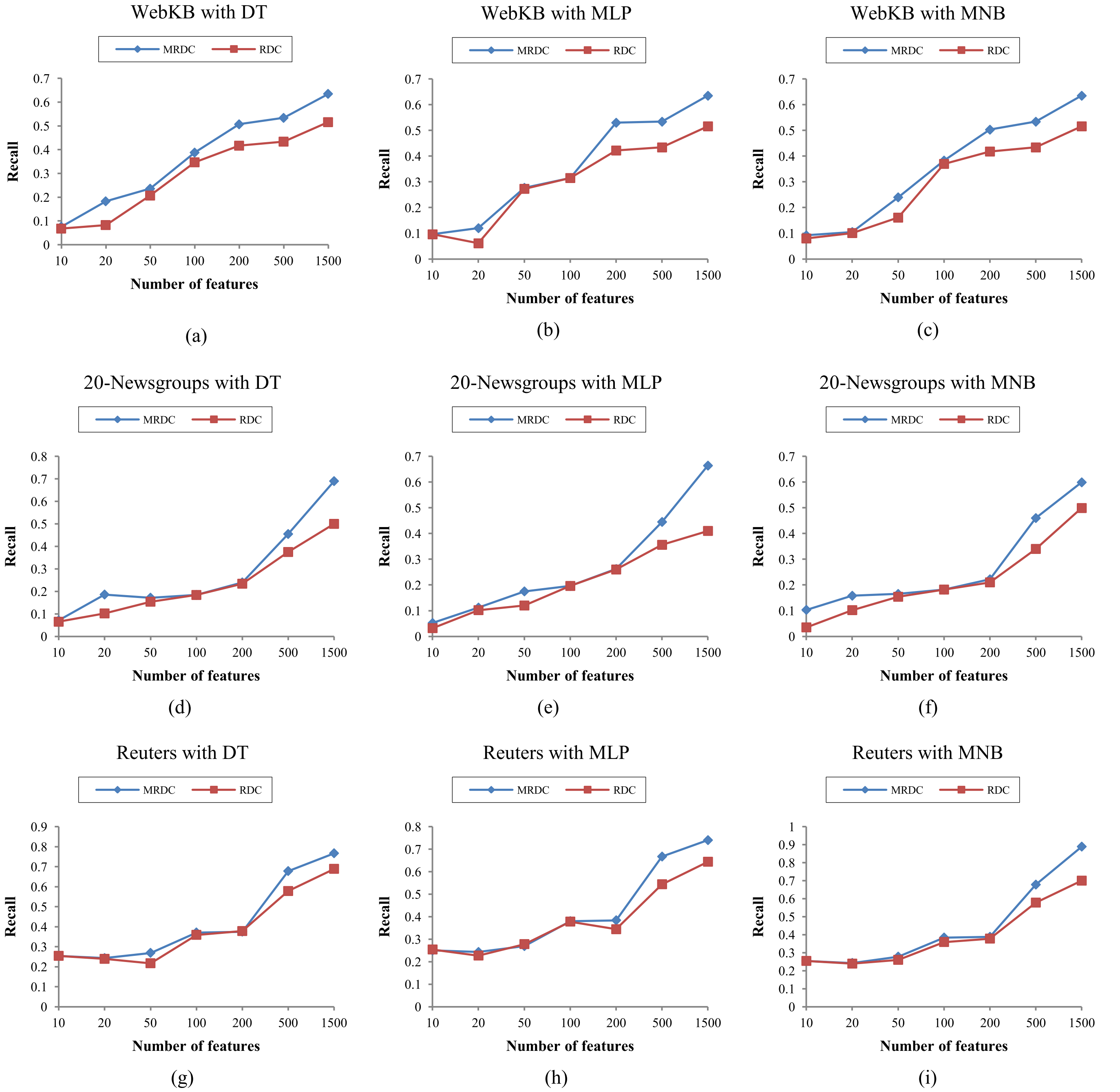


Fig. 3. Recall of MRDC and RDC methods for (a) DT classifier over WebKB, (b) MLP classifier over WebKB, (c) MNB classifier over WebKB, (d) DT classifier over 20-Newsgroups, (e) MLP classifier over 20-Newsgroups, (f) MNB classifier over 20-Newsgroups, (g) DT classifier over Reuters-21578, (h) MLP classifier over Reuters-21578, and (i) MNB classifier over Reuters-21578.

4.4. Results

4.4.1. Comparison of MRDC and RDC

The aim of this section is to compare MRDC with RDC that is a univariate method. Tables 5 and 6 report the obtained results for MRDC and RDC methods. From the results it can be seen that the overall performance of MRDC is better than RDC. For example, the precision, recall and F-measure of MRDC are 0.392, 0.384 and 0.387, respectively, while they are 0.310, 0.344 and 0.326 for RDC.

The aim of filter methods for feature selection is to assign a rank value to each feature based on its importance. To compare the performance of MRDC and RDC methods in a fair situation, one should select equal number of features. Here, we vary the number of features in the range [10 1500] for both methods and compare their performance (Fig. 3). This figure graphically compares MRDC and RDC in terms of their recall values for different classifiers and different datasets; MRDC outperforms RDC in most cases, especially when the number of features is larger than 200. Also, the results show that due to information loss, as the number of features decreases, the performance declines. On the other hand, when higher numbers of features are selected, the proposed method seeks to select the most discriminant features by ignoring redundant features, while RDC may selects redundant ones leading to poor accuracy.

Table 7
The precision values of MRDC compared to RDC, IG, GI and mRMR methods when DT classifier is used. The best results for each dataset are indicated in bold face and underlined the second best in bold face. The number of the parentheses is the rank of the classifier.

Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.059(2)**	**0.108(1)**	**0.293(1)**	**0.366(1)**	**0.510(1)**	**0.534(1)**	**0.604(1)**	1.14
	RDC	0.023(4)	0.066(3)	0.180(4)	0.214(4)	**0.417(2)**	**0.480(2)**	**0.539(2)**	3
	IG	**0.086(1)**	0.011(5)	**0.189(2)**	**0.233(2)**	0.389(3)	0.389(3)	0.332(4)	2.71
	GI	0.021(3)	0.022(4)	0.185(3)	0.218(3)	0.208(5)	0.308(5)	0.310(5)	4
	MRMR	0.021(3)	**0.106(2)**	0.106(5)	0.106(5)	0.246(4)	0.346(4)	0.390(3)	3.57
20 Newsgroups	MRDC	0.048(3)	**0.137(1)**	**0.164(2)**	**0.183(2)**	**0.289(1)**	**0.520(1)**	**0.541(1)**	1.71
	RDC	0.017(4)	**0.135(2)**	0.145(3)	0.164(4)	**0.273(2)**	**0.400(2)**	**0.500(2)**	2.71
	IG	**0.139(1)**	0.059(5)	0.081(4)	**0.199(1)**	0.108(4)	0.194(4)	0.259(3)	3.14
	GI	0.014(5)	0.106(3)	0.001(5)	0.048(5)	0.034(5)	0.097(5)	0.216(5)	4.71
	MRMR	**0.082(2)**	0.060(4)	**0.173(1)**	0.175(3)	0.225(3)	0.270(3)	0.275(4)	2.85
Reuters-21,578	MRDC	**0.254(1)**	**0.186(2)**	**0.233(1)**	0.327(3)	**0.374(2)**	**0.699(1)**	**0.780(1)**	1.57
	RDC	0.178(5)	0.162(5)	0.208(3)	**0.359(2)**	0.359(3)	0.564(3)	0.627(3)	3.42
	IG	**0.197(2)**	0.186(3)	0.190(5)	0.272(5)	0.264(5)	0.306(5)	0.406(5)	4.28
	GI	0.189(3)	0.177(4)	0.205(4)	0.302(4)	0.299(4)	0.384(4)	0.484(4)	3.85
	MRMR	0.188(4)	**0.188(1)**	**0.218(2)**	**0.388(1)**	**0.388(1)**	**0.600(2)**	**0.682(2)**	1.85

Table 8
The precision values of MRDC compared to RDC, IG, GI and mRMR methods when MLP classifier is used. Other designations are as Table 7.

Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.151(1)**	**0.167(1)**	**0.216(1)**	**0.384(1)**	**0.530(1)**	**0.534(1)**	**0.660(1)**	1
	RDC	**0.094(2)**	0.134(3)	**0.215(2)**	**0.382(2)**	**0.427(2)**	**0.434(2)**	**0.512(2)**	2.14
	IG	0.078(3)	0.045(5)	0.165(3)	0.251(4)	0.340(3)	0.317(5)	0.300(5)	4
	GI	0.031(5)	**0.140(2)**	0.130(4)	0.281(3)	0.183(5)	0.333(3)	0.354(4)	3.71
	MRMR	0.072(4)	0.068(4)	0.063(5)	0.058(5)	0.292(4)	0.322(4)	0.360(3)	4.14
20 Newsgroups	MRDC	**0.040(2)**	**0.165(1)**	**0.186(2)**	**0.196(2)**	**0.296(1)**	**0.499(1)**	**0.501(1)**	1.42
	RDC	0.039(3)	**0.155(2)**	0.184(3)	0.196(3)	**0.290(2)**	**0.400(2)**	**0.500(2)**	2.42
	IG	**0.114(1)**	0.012(5)	0.108(4)	**0.237(1)**	0.191(3)	0.269(3)	0.298(3)	2.85
	GI	0.027(4)	0.032(4)	0.001(5)	0.099(5)	0.105(4)	0.119(5)	0.216(4)	4.42
	MRMR	0.017(5)	0.15(3)	**0.189(1)**	0.161(4)	0.103(5)	0.120(4)	0.124(5)	3.85
Reuters-21,578	MRDC	**0.250(1)**	**0.213(1)**	**0.228(1)**	**0.370(2)**	**0.392(2)**	**0.721(1)**	**0.770(1)**	1.28
	RDC	0.177(5)	0.180(3)	0.214(3)	0.370(3)	0.310(3)	**0.589(2)**	**0.650(2)**	3
	IG	0.194(3)	**0.183(2)**	0.153(5)	0.263(5)	0.277(5)	0.387(5)	0.477(5)	4.28
	GI	**0.189(2)**	0.177(4)	0.197(4)	0.295(4)	0.283(4)	0.416(4)	0.508(4)	3.71
	MRMR	0.188(4)	0.118(5)	**0.218(2)**	**0.388(1)**	**0.418(1)**	0.540(3)	0.590(3)	2.71

Table 9
The precision values of MRDC compared to RDC, IG, GI and mRMR methods when MNB classifier is used. Other designations are as Table 7.

Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.102(1)**	**0.140(1)**	**0.290(1)**	**0.375(1)**	**0.503(1)**	**0.534(1)**	**0.645(1)**	1
	RDC	**0.094(2)**	**0.130(2)**	0.180(3)	**0.215(2)**	**0.487(2)**	**0.513(2)**	**0.574(2)**	2.14
	IG	0.084(3)	0.035(3)	0.106(4)	**0.215(2)**	0.387(3)	0.333(4)	0.230(5)	3.42
	GI	0.026(5)	**0.140(1)**	**0.187(2)**	0.202(3)	0.026(5)	0.211(5)	0.312(4)	3.57
	MRMR	0.072(4)	0.126(4)	0.104(5)	0.056(4)	0.259(4)	0.385(3)	0.412(3)	3.85
20 Newsgroups	MRDC	0.036(3)	**0.153(1)**	**0.172(1)**	**0.181(1)**	**0.288(1)**	**0.487(1)**	**0.501(1)**	1.28
	RDC	0.028(4)	**0.152(2)**	**0.166(2)**	**0.175(2)**	**0.278(2)**	**0.400(2)**	**0.500(2)**	2.28
	IG	**0.106(1)**	0.021(5)	0.043(4)	0.081(4)	0.276(3)	0.180(4)	0.226(4)	3.57
	GI	0.018(5)	0.034(4)	0.001(5)	0.048(5)	0.099(5)	0.107(5)	0.265(3)	4.57
	MRMR	**0.048(2)**	0.129(3)	0.190(3)	0.113(3)	0.200(4)	0.202(3)	0.221(5)	3.28
Reuters-21,578	MRDC	**0.254(1)**	**0.186(1)**	**0.238(1)**	0.348(3)	**0.441(1)**	**0.675(2)**	**0.722(1)**	1.42
	RDC	0.172(5)	0.162(3)	0.208(4)	**0.359(2)**	0.359(3)	0.538(3)	0.654(3)	3.28
	IG	**0.197(2)**	**0.186(1)**	0.213(3)	0.288(5)	0.264(4)	0.330(5)	0.392(5)	3.57
	GI	0.189(3)	**0.177(2)**	0.198(5)	0.289(4)	0.235(5)	0.481(4)	0.533(4)	3.85
	MRMR	0.188(4)	0.102(4)	**0.218(2)**	**0.388(1)**	**0.402(2)**	**0.680(1)**	**0.695(2)**	2.28

4.4.2. Comparison of MRDC with other filter methods

The aim of this section is to compare the performance of the proposed method with state-of-the-art filter methods including RDC, IG, GI and mRMR. Tables 7–9 compare the precision of the methods for DT, MLP and NN classifiers, respectively. From the results it can be seen that for WebKB dataset, the proposed method outperforms others in all cases. For 20-Newsgroup dataset, MRDC is the top-performed, while it is has the second best performance for Reuters-21,578 dataset. These results also show that for these two datasets, MRDC obtains the highest precision value when 500 and 1500 features are selected. Moreover, Table 7 shows that for WebKB and 20-Newsgroups datasets MRDC results in the highest precision values for at least 85% of the trials.

Table 10
The recall values of MRDC compared to RDC, IG, GI and mRMR methods when DT classifier is used. Other designations are as Table 7.

Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.076(1)**	**0.183(1)**	**0.236(1)**	**0.388(1)**	**0.507(1)**	**0.534(1)**	**0.635(1)**	1
	RDC	**0.068(2)**	**0.083(2)**	0.207(3)	**0.347(2)**	**0.417(2)**	**0.434(2)**	**0.516(2)**	2.14
	IG	0.011(4)	0.067(3)	**0.222(2)**	0.167(4)	0.333(3)	0.433(3)	0.222(5)	3.42
	GI	0.025(3)	0.022(5)	0.111(5)	0.111(5)	0.267(5)	0.367(4)	0.467(3)	4.28
	MRMR	0.011(4)	0.065(4)	0.152(4)	0.240(3)	0.278(4)	0.278(5)	0.360(4)	4
20 Newsgroups	MRDC	0.071(3)	**0.186(1)**	**0.172(1)**	**0.184(2)**	**0.240(1)**	**0.455(1)**	**0.690(1)**	1.42
	RDC	0.065(4)	0.102(3)	**0.154(2)**	**0.184(2)**	**0.234(2)**	**0.375(2)**	**0.500(2)**	2.42
	IG	**0.167(1)**	0.097(4)	0.097(3)	0.161(3)	0.097(4)	0.129(4)	0.439(3)	3.14
	GI	0.017(5)	0.097(4)	0.032(4)	0.065(4)	0.065(5)	0.097(5)	0.217(4)	3.85
	MRMR	**0.079(2)**	**0.129(2)**	0.032(4)	**0.194(1)**	0.161(3)	0.173(3)	0.180(5)	2.85
Reuters-21,578	MRDC	**0.254(2)**	**0.243(1)**	**0.269(1)**	**0.371(2)**	0.375(3)	**0.678(1)**	**0.767(2)**	1.57
	RDC	**0.254(2)**	**0.239(2)**	0.217(5)	0.359(3)	**0.378(1)**	0.578(4)	**0.689(1)**	2.71
	IG	0.253(3)	0.238(4)	0.261(4)	0.232(5)	0.232(4)	0.548(5)	0.548(5)	4.28
	GI	**0.266(1)**	0.227(5)	0.261(3)	0.258(4)	0.258(3)	**0.613(2)**	0.613(3)	3
	MRMR	0.200(4)	0.275(3)	**0.250(2)**	**0.373(1)**	**0.380(1)**	0.590(3)	0.595(4)	2.57

Table 11
The recall values of MRDC compared to RDC, IG, GI and mRMR methods when MLP classifier is used. Other designations are as Table 7.

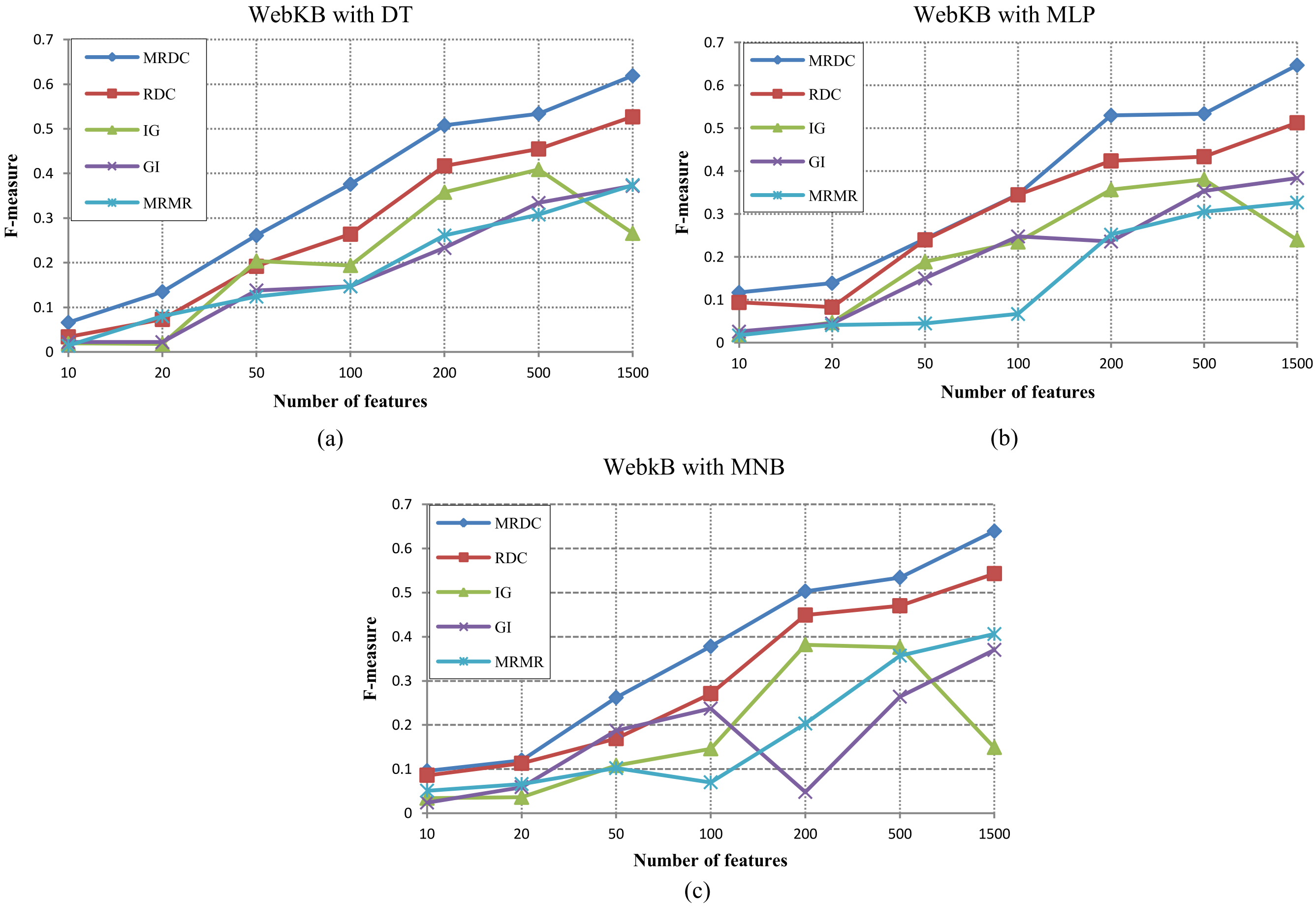
Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.096(1)**	**0.120(1)**	**0.277(1)**	**0.315(1)**	**0.530(1)**	**0.534(1)**	**0.635(1)**	1
	RDC	**0.096(1)**	**0.061(2)**	**0.273(2)**	**0.315(1)**	**0.422(2)**	0.434(3)	**0.516(2)**	1.85
	IG	0.011(3)	0.050(3)	0.222(3)	**0.222(2)**	0.378(3)	**0.478(2)**	0.200(5)	3
	GI	**0.023(2)**	0.027(5)	0.178(4)	**0.222(2)**	0.333(4)	0.378(4)	0.421(3)	3.42
	MRMR	0.010(4)	0.030(4)	0.035(5)	0.080(4)	0.222(5)	0.290(5)	0.300(4)	4.42
20 Newsgroups	MRDC	0.052(3)	**0.112(2)**	**0.175(1)**	**0.196(1)**	**0.262(1)**	**0.445(1)**	**0.664(1)**	1.42
	RDC	0.032(4)	0.102(3)	0.120(3)	**0.196(1)**	**0.260(2)**	**0.356(2)**	0.410(3)	2.57
	IG	**0.183(1)**	0.226(4)	**0.129(2)**	**0.161(2)**	0.226(3)	0.226(3)	**0.438(2)**	2.42
	GI	0.030(5)	0.032(5)	0.032(5)	0.129(3)	0.129(4)	0.129(4)	0.300(4)	4.28
	MRMR	**0.065(2)**	**0.115(1)**	0.097(4)	0.065(4)	0.097(5)	0.105(5)	0.124(5)	3.71
Reuters-21,578	MRDC	0.250(3)	**0.243(1)**	**0.269(2)**	**0.380(1)**	**0.384(1)**	**0.667(2)**	**0.740(1)**	1.57
	RDC	**0.254(2)**	**0.227(2)**	**0.278(1)**	**0.378(2)**	0.344(3)	0.544(4)	0.644(3)	2.42
	IG	0.241(4)	0.159(5)	0.195(5)	0.245(4)	0.245(5)	0.548(3)	0.516(5)	3.85
	GI	**0.266(1)**	0.205(4)	0.261(3)	0.258(3)	0.258(4)	**0.677(1)**	**0.645(2)**	2.57
	MRMR	0.202(5)	0.210(3)	0.224(4)	**0.380(1)**	**0.380(2)**	0.340(5)	0.524(4)	3.42

Table 12
The recall values of MRDC compared to RDC, IG, GI and mRMR methods when MNB classifier is used. Other designations are as Table 7.

Dataset	Method	Number of selected features							Average rank
		10	20	50	100	200	500	1500	
WebKB	MRDC	**0.092(1)**	**0.105(1)**	**0.240(1)**	**0.383(1)**	**0.503(1)**	**0.534(1)**	**0.635(1)**	1
	RDC	**0.080(2)**	**0.101(2)**	**0.161(2)**	**0.370(2)**	**0.418(2)**	**0.434(2)**	**0.516(2)**	2
	IG	0.022(5)	0.038(4)	0.111(4)	0.111(4)	0.378(3)	0.433(3)	0.111(5)	4
	GI	0.023(4)	0.038(4)	0.189(3)	0.289(3)	0.333(4)	0.356(4)	0.456(3)	3.57
	MRMR	0.040(3)	0.045(3)	0.102(5)	0.095(5)	0.167(5)	0.333(5)	0.402(4)	4.28
20 Newsgroups	MRDC	**0.103(2)**	**0.158(1)**	**0.165(1)**	**0.182(1)**	**0.222(1)**	**0.460(1)**	**0.599(1)**	1.14
	RDC	0.035(3)	**0.102(2)**	**0.154(2)**	**0.182(1)**	**0.210(2)**	**0.340(2)**	**0.499(2)**	2
	IG	**0.267(1)**	0.097(3)	0.097(3)	**0.065(2)**	0.194(3)	0.226(3)	0.406(3)	2.57
	GI	0.026(5)	0.065(4)	0.032(4)	**0.065(2)**	0.129(4)	0.161(4)	0.267(4)	3.85
	MRMR	0.032(4)	0.097(3)	0.097(3)	**0.065(2)**	0.129(4)	0.134(5)	0.173(5)	3.71
Reuters-21,578	MRDC	**0.254(2)**	**0.243(1)**	**0.278(1)**	**0.384(1)**	**0.388(1)**	**0.678(1)**	**0.889(1)**	1.28
	RDC	**0.254(2)**	0.239(3)	0.260(3)	**0.359(2)**	**0.378(2)**	0.578(4)	**0.700(2)**	2.57
	IG	0.253(3)	0.238(4)	**0.274(2)**	0.258(3)	0.258(4)	0.581(3)	0.548(4)	3.28
	GI	**0.266(1)**	0.193(5)	0.235(4)	0.232(4)	0.232(5)	**0.677(2)**	0.645(3)	3.42
	MRMR	0.250(4)	**0.240(2)**	0.260(3)	0.230(5)	0.301(3)	0.450(5)	0.490(5)	3.85

For the Reuter-21,578 dataset, MRDC obtains the second best precision values after GI, while in more than 57% of the trials MRDC has the best precision. We also find that the highest precision value (0.780) is for Reuter-21,578 dataset with 1500 features. Similar results are also reported in Tables 8 and 9.

The proposed method is also compared with others in terms of recall measure, with results shown in Tables 10–12 using DT, MLP and NB classifiers, respectively. It is seen that in most cases MRDC results in the highest recall values and outperforms others. For instance, for WebKB dataset, the proposed method has the highest recall values for DT classifier when different numbers of features are selected. Tables 11 and



(a)

(b)

(c)

Fig. 4. F-measure of the methods obtained over WebKB dataset when different numbers of features are selected for classification using (a) DT, (b) MLP, and (c) MNB classifiers.

12 also report similar results for MLP and MNB classifiers, respectively. Table 10 further shows that in 20-Newsgroup dataset, for all sizes of the feature set, except 10 and 100, the proposed method could select a subset of features leading to improve the performance of DT classifier. Although in the cases of having 10 and 100 features, MRDC could not attain the best values, the difference between its performance of and that of the best ones are only 0.008 and 0.01, respectively. The results also show that for Reuter-21,578 dataset, the proposed method obtains the best recall values in all cases when the MNB classifier is used. MRDC results in the best values in almost 60% of situations when DT and MLP classifiers are used.

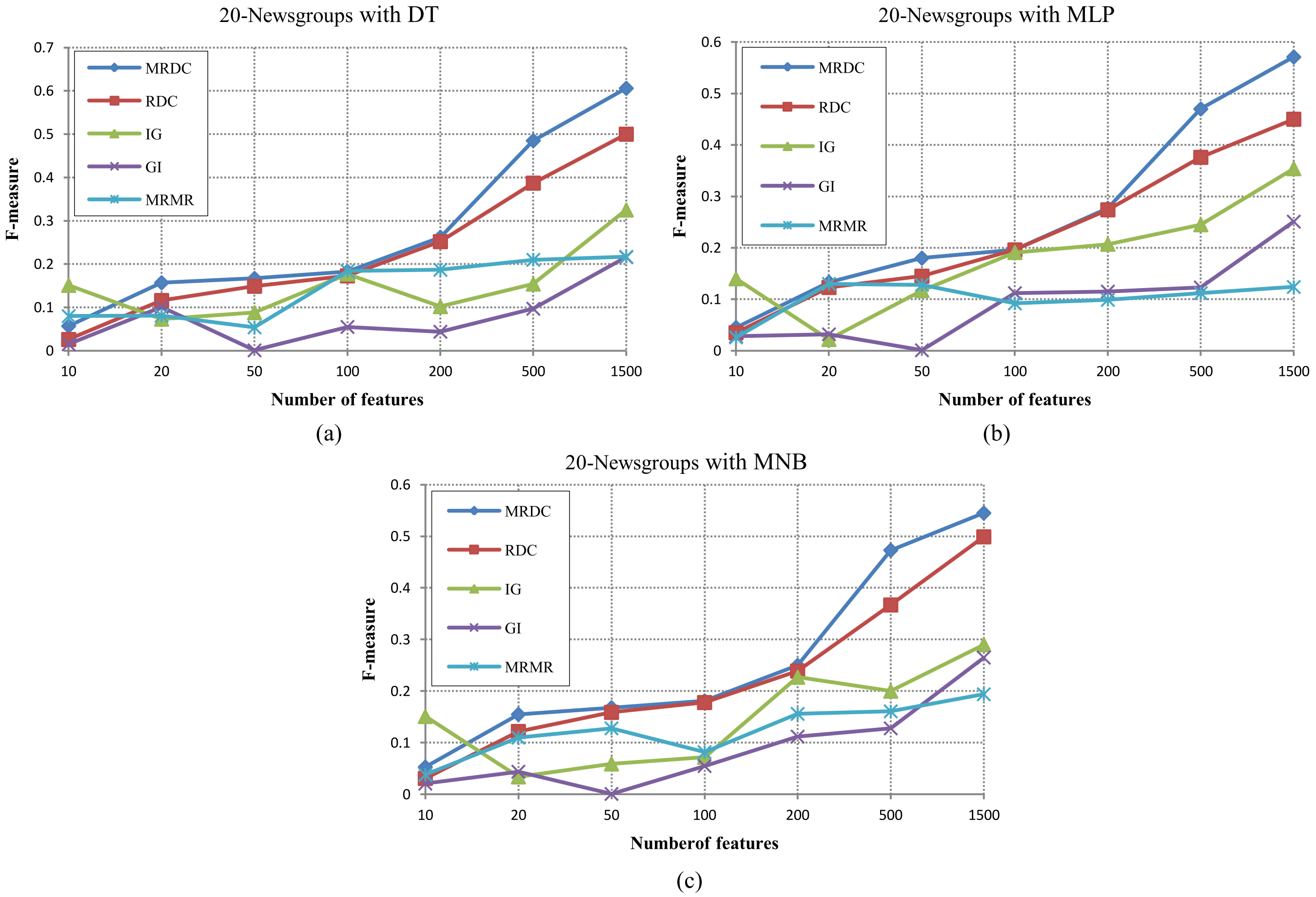
Figs. 4–6 graphically compare F-measure for the methods. Again, the proposed method has the best performance among them. For instance, Fig. 4(a) and 4(b) show that when large (higher than 200) number of features is used for classification, MRDC results in the best performance. It can also be observed that in most cases the proposed method achieves the highest F-measure when MLP classifier is used (Fig. 5).

Generally, the superiority of the proposed method is more pronounced when large number of features is used for classification, indicating that the number of selected features is a critical factor in the performance of text classification methods. In the WebKB dataset, when the size of the feature set is set to a small value such as 10, all methods result in their worst performance. This is due to the fact that small number of features does not carry enough predictive information to allow an effective performance. The following conclusions can be made from the conducted experiments:

- The results show that MRDC performs better than the traditional univariate feature selection methods such as IG, GI and RDC. This is due to the fact that these methods do not consider the similarity between features, while MRDC considers redundancy between features.
- Although mRMR and MRDC are multivariate feature selection methods, the results show that the proposed method significantly outperforms mRMR. This is because MRDC is an extension of RDC which has been specifically designed for text categorization methods, while mRMR is a general multivariate feature selection method and not specific for text classification.
- The results reveal that the proposed method selects more informative features compared to the traditional methods, and consequently improves the performance of the classifiers. Moreover, the results show that the performance of MRDC gradually improves as we move from a less refined feature subspace to more refined one.

4.5. Statistical analysis

In this section, the Friedman test is applied on the results to show that the obtained results are statistically significant. The Friedman test (Friedman, 1937) is a non-parametric test which offers a powerful procedure to measure the statistical differences between a number of methods over several datasets. To apply this test, the methods are ranked based on the results of classifier accuracy metrics (i.e., precision, recall and F-measure). In case some methods have the same performance value, the average rank is assigned to them. The Friedman test has been frequently used in the literature to statistically analyze feature selection methods (Moradi and Gholampour, 2016; Tabakhi et al., 2015; Yang et al., 2011). It compares the average ranks of M methods on N datasets.



(a)

(b)

(c)

Fig. 5. F-measure of the methods obtained over 20-Newsgroup dataset when different numbers of features are selected for classification using (a) DT, (b) MLP, and (c) MNB classifiers.

This test follows a Fisher distribution with $M-1$ and $(M-1)(N-1)$ degrees of freedom, and is defined as:

$$F_F = \frac{(N-1)\chi_F^2}{N(M-1)-\chi_F^2} \tag{16}$$

where

$$\chi_F^2 = \frac{12N}{M(M+1)}\left[\sum_{j=1}^{K} R_j^2 - \frac{M(M+1)^2}{4}\right] \tag{17}$$

and R_j is the mean rank of the jth method over N datasets ($R_j = \frac{1}{N}\sum_{i=1\ldots N} r_i^j$). If the value of F_F is lower than a critical value α, the null hypothesis is accepted, meaning that all methods perform equivalently at the significance level α; otherwise it is rejected. In this paper the Friedman test is applied on both Precision (Tables 7–9) and Recall (Tables 10–12) obtained by the methods using different classifiers, and the results are respectively reported in Tables 13 and 14. In the experiments, the number of datasets is $N = 3$, the number of methods is $M = 5$ and the critical value of Fisher distribution with $M-1$ $(5-1=4)$ degrees of freedom is $F(4,8) = 3.838$ for $\alpha = 0.05$. We used IBM SPSS V22 to obtain the Friedman test results. The results show that for all cases the value of F_F is greater than the critical value 3.838. Thus, the null hypothesis is rejected for all classifiers over both Precision and Recall results. Therefore, it can be concluded that the reported accuracy results are statistically significant.

Table 13
The results of Friedman test on precision results.

Classifier	X_F^2	F_F	$F(4,8)$	Significant
DT	9.034	6.091	3.838	+
MLP	8.267	4.429	3.838	+
MNB	9.153	6.429	3.838	+

Table 14
The results of Friedman test on recall results.

Classifier	X_F^2	F_F	$F(4,8)$	Significant
DT	9.867	9.251	3.838	+
MLP	8.800	5.5	3.838	+
MNB	9.153	6.429	3.838	+

5. Conclusion

Selection of an informative subset of features is one of the main challenges in mining text data due to computational complexity and accuracy considerations. High dimensionality of a feature space, especially in text classification tasks which consist of a large number of words, leads to increased computational cost and reduced classification performance. Feature selection methods should be able to identify and remove as many of the irrelevant and redundant features as possible. Most of feature selection methods can effectively remove irrelevant features, but fail to handle the redundant ones. In this paper, a novel multivariate filter method for feature selection, called MRDC, was introduced for text classification tasks. The aim of the proposed method, which is a multivariate extension to RDC method, is to remove both redundant and irrelevant features from the feature space. We compared

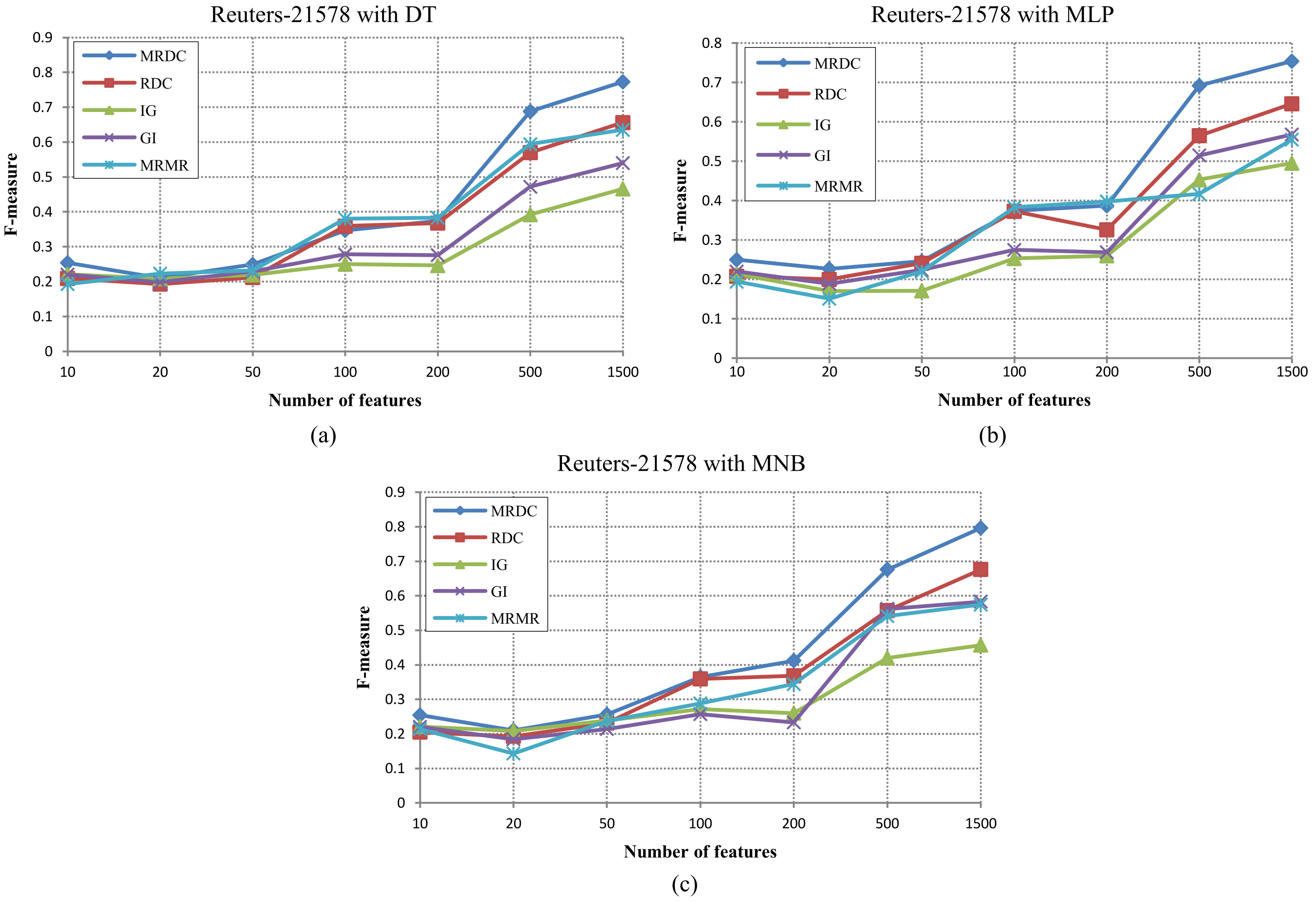


Fig. 6. *F*-measure of the methods obtained over Reuters-21,578 when different numbers of features are selected for classification using (a) DT, (b) MLP, and (c) MNB classifiers.

the performance of the proposed method with a number of state-of-the-art method by applying them on three datasets (Reuters-21,578, 20-Newsgroups, and WebKB) and using DT, MNB and MLP as classifier. The results showed the best performance for the proposed method in terms of Precision, Recall and *F*-measure.

Acknowledgment

Mahdi Jalili is supported by Australian Research Council through Grant No. DE140100620.

References

Adeva, J.J.G., Atxa, J.M.P., 2007. Intrusion detection in web applications using text mining. Eng. Appl. Artif. Intell. 20, 555–566.

Agarwal, B., Mittal, N., 2014. Text classification using machine learning methods-a survey. In: Proceedings of the Second International Conference on Soft Computing for Problem Solving. SocProS 2012, December 28–30, 2012. Springer, pp. 701–709.

Badawi, D., Altınçay, H., 2014. A novel framework for termset selection and weighting in binary text classification. Eng. Appl. Artif. Intell. 35, 38–53.

Bharti, K.K., Singh, P.K., 2015. Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. Expert Syst. Appl. 42, 3105–3114.

Chouaib, H., Terrades, O.R., Tabbone, S., Cloppet, F., Vincent, N., 2008. Feature selection combining genetic algorithm and adaboost classifiers. In: Pattern Recognition, 2008. ICPR 2008. 19th International Conference on. IEEE, pp. 1–4.

Erenel, Z., Altınçay, H., 2012. Nonlinear transformation of term frequencies for term weighting in text categorization. Eng. Appl. Artif. Intell. 25, 1505–1514.

Forman, G., 2003. An extensive empirical study of feature selection metrics for text classification. J. Mach. Learn. Res. 3, 1289–1305.

Friedman, M., 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J. Amer. Statist. Assoc. 32, 675–701.

Günal, S., 2012. Hybrid feature selection for text classification. Turkish J. Electr. Eng. Comput. Sci. 20, 1296–1311.

Guzella, T.S., Caminhas, W.M., 2009. A review of machine learning approaches to spam filtering. Expert Syst. Appl. 36, 10206–10222.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten,, I.H., 2009. The WEKA data mining software: an update. SIGKDD Explor. Newsl. 11, 10–18.

Hu, Z., Bao, Y., Xiong, T., Chiong, R., 2015. Hybrid filter–wrapper feature selection for short-term load forecasting. Eng. Appl. Artif. Intel. 40, 17–27.

Idris, I., Selamat, A., 2014. Improved email spam detection model with negative selection algorithm and particle swarm optimization. Appl. Soft Comput. 22, 11–27.

Jiang, L., Li, C., Wang, S., Zhang, L., 2016. Deep feature weighting for naive Bayes and its application to text classification. Eng. Appl. Artif. Intel. 52, 26–39.

Jiang, Y.-G., Ngo, C.-W., Yang, J., 2007. Towards optimal bag-of-features for object categorization and semantic video retrieval. In: Proceedings of the 6th ACM International Conference on Image and Video Retrieval. ACM, pp. 494–501.

Kim, J.W., Lee, B.H., Shaw, M.J., Chang, H.-L., Nelson, M., 2001. Application of decision-tree induction techniques to personalized advertisements on internet storefronts. Int. J. Electron. Commerce 5, 45–62.

Kolenda, T., Hansen, L.K., Sigurdsson, S., 2000. Independent Components in Text, Advances in Independent Component Analysis. Springer, pp. 235–256.

Lan, M., Tan, C.L., Su, J., Lu, Y., 2009. Supervised and traditional term weighting methods for automatic text categorization. IEEE Trans. Pattern Anal. Mach. Intel. 31, 721–735.

Li, Y., Luo, C., Chung, S.M., 2008. Text clustering with feature selection by using statistical data. IEEE Trans. Knowl. Data Eng. 20, 641–652.

Li, C.H., Park, S.C., 2007. Neural network for text classification based on singular value decomposition. In: Computer and Information Technology, 2007. CIT 2007. 7th IEEE International Conference on. IEEE, pp. 47–52.

Liu, L., Kang, J., Yu, J., Wang, Z., 2005. A comparative study on unsupervised feature selection methods for text clustering. In: Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on. IEEE, pp. 597–601.

Medhat, W., Hassan, A., Korashy, H., 2014. Sentiment analysis algorithms and applications: A survey. Ain Shams Eng. J. 5, 1093–1113.

Mengle, S.S., Goharian, N., 2009. Ambiguity measure feature-selection algorithm. J. Am. Soc. Inf. Sci. Technol. 60, 1037–1050.

Miller, G.A., Newman, E.B., 1958. Tests of a statistical explanation of the rank-frequency relation for words in written english. Am. J. Psychol. 71, 209–218.

Moradi, P., Gholampour, M., 2016. A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. Appl. Soft Comput. 43, 117–130.

Myllymäki, P., Tirri, H., 1993. Bayesian case-based reasoning with neural networks. In: Neural Networks, 1993. IEEE International Conference on. IEEE, pp. 422–427.

Nowak, E., Jurie, F., Triggs, B., 2006. Sampling Strategies for Bag-of-Features Image Classification, European Conference on Computer Vision. Springer, pp. 490–503.

Ogura, H., Amano, H., Kondo, M., 2009. Feature selection with a measure of deviations from poisson in text categorization. Expert Syst. Appl. 36, 6826–6832.

Özel, S.A., 2011. A web page classification system based on a genetic algorithm using tagged-terms as features. Expert Syst. Appl. 38, 3407–3415.

Perikos, I., Hatzilygeroudis, I., 2016. Recognizing emotions in text using ensemble of classifiers. Eng. Appl. Artif. Intel. 51, 191–201.

Porter, M.F., 1980. An algorithm for suffix stripping. Program 14 (3), 130–137.

Rehman, A., Javed, K., Babri, H.A., Saeed, M., 2015. Relative discrimination criterion–A novel feature ranking method for text data. Expert Syst. Appl. 42, 3670–3681.

Saeys, Y., Inza, I., Larrañaga, P., 2007. A review of feature selection techniques in bioinformatics. Bioinformatics 23, 2507–2517.

Saleh, S.N., El-Sonbaty, Y., 2007. A feature selection algorithm with redundancy reduction for text classification. In: Computer and Information Sciences, 2007. Iscis 2007. 22nd International Symposium on. IEEE, pp. 1–6.

Salton, G., Wong, A., Yang, C.S., 1975. A vector space model for automatic indexing. Commun. ACM 18, 613–620.

Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., Wang, Z., 2007. A novel feature selection algorithm for text categorization. Expert Syst. Appl. 33, 1–5.

Shang, C., Li, M., Feng, S., Jiang, Q., Fan, J., 2013. Feature selection via maximizing global information gain for text classification. Knowl.-Based Syst. 54, 298–309.

Tabakhi, S., Najafi, A., Ranjbar, R., Moradi, P., 2015. Gene selection for microarray data classification using a novel ant colony optimization. Neurocomputing 168, 1024–1036.

Wang, Z., Qian, X., 2008. Text categorization based on LDA and SVM. In: Computer Science and Software Engineering, 2008 International Conference on. IEEE, pp. 674–677.

Xu, Y., Jones, G.J., Li, J., Wang, B., Sun, C., 2007. A study on mutual information-based feature selection for text categorization. J. Comput. Inf. Syst. 3, 1007–1012.

Yang, Y., 1995. Noise reduction in a statistical approach to text categorization. In: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, pp. 256–263.

Yang, J., Liu, Y., Liu, Z., Zhu, X., Zhang, X., 2011. A new feature selection algorithm based on binomial hypothesis testing for spam filtering. Knowl.-Based Syst. 24, 904–914.

Yang, Y., Pedersen, J.O., 1997. A comparative study on feature selection in text categorization. In: ICML. pp. 412–420.

Zeng, J., Zhang, S., 2007. Variable space hidden Markov model for topic detection and analysis. Knowl.-Based Syst. 20, 607–613.

Zhang, C., Wu, X., Niu, Z., Ding, W., 2014. Authorship identification from unstructured texts. Knowl.-Based Syst. 66, 99–111.