Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A new feature selection method for handling redundant information in text classification*

You-wei WANG[†‡1], Li-zhou FENG[2]

[1]*School of Information, Central University of Finance and Economics, Beijing 100081, China*

[2]*School of Science and Engineering, Tianjin University of Finance and Economics, Tianjin 300222, China*

[†]E-mail: ywwang15@126.com

Received Nov. 30, 2016; Revision accepted Feb. 5, 2017; Crosschecked Feb. 8, 2018

Abstract: Feature selection is an important approach to dimensionality reduction in the field of text classification. Because of the difficulty in handling the problem that the selected features always contain redundant information, we propose a new simple feature selection method, which can effectively filter the redundant features. First, to calculate the relationship between two words, the definitions of word frequency based relevance and correlative redundancy are introduced. Furthermore, an optimal feature selection (OFS) method is chosen to obtain a feature subset FS_1. Finally, to improve the execution speed, the redundant features in FS_1 are filtered by combining a predetermined threshold, and the filtered features are memorized in the linked lists. Experiments are carried out on three datasets (WebKB, 20-Newsgroups, and Reuters-21578) where in support vector machines and naïve Bayes are used. The results show that the classification accuracy of the proposed method is generally higher than that of typical traditional methods (information gain, improved Gini index, and improved comprehensively measured feature selection) and the OFS methods. Moreover, the proposed method runs faster than typical mutual information-based methods (improved and normalized mutual information-based feature selections, and multilabel feature selection based on maximum dependency and minimum redundancy) while simultaneously ensuring classification accuracy. Statistical results validate the effectiveness of the proposed method in handling redundant information in text classification.

Key words: Feature selection; Dimensionality reduction; Text classification; Redundant features; Support vector machine; Naïve Bayes; Mutual information

https://doi.org/10.1631/FITEE.1601761 **CLC number:** TP391

1 Introduction

Text classification assigns a predefined category to an unlabeled text document. It has become a very efficient method to manage the vast volumes of digital documents available on the Internet (Apte et al., 1999; Uğuz, 2011). In recent years, many classifiers have been applied to text classification based on machine learning and statistical theory. Among these classifiers, decision trees (Apte et al., 1999), *k*-nearest neighbors (KNN) (Dallachiesa et al., 2014), neural networks (De Souza et al., 2009), naïve Bayes (NBs) (McCallum and Nigam, 2001), and support vector machines (SVMs) (Caruana et al., 2013) are the most successful and widely used methods (Chen et al., 2009).

In text classification tasks, documents are represented as vectors of features when using different classifiers. As the most popular document representation method, the vector space model (VSM) uses a bag of 'words' or 'terms' to construct a feature vector (Jing et al., 2010). However, because a document collection usually contains thousands of words, the

‡ Corresponding author

* Project supported by the Joint Funds of the National Natural Science Foundation of China (No. U1509214), the Beijing Natural Science Foundation, China (No. 4174105), the Key Projects of National Bureau of Statistics of China (No. 2017LZ05), and the Discipline Construction Foundation of the Central University of Finance and Economics, China (No. 2016XX02)

ORCID: You-wei WANG, http://orcid.org/0000-0002-3925-3422

dimension of the feature vector is usually high, causing a computational obstacle in the machine learning process. Moreover, most of the features in the original feature vectors are irrelevant or redundant, leading to a low accuracy when classifiers are used. Therefore, it is desirable to find suitable approaches to reduce the number of features and improve the qualities of the selected features.

There have been many methods for reducing dimensions of feature vectors. Among them, feature extraction and selection have been used widely. Feature extraction methods generate new features by transforming the original feature vectors into different feature spaces. Typical feature extraction methods include: latent semantic indexing (LSI) (Elghazel et al., 2016), partial least square (Tenenhaus et al., 2005), multidimensional scaling (Kruskal and Wish, 1978), latent Dirichlet allocation (LDA) (Zhang et al., 2016), and mixed graph of terms (MGT) (Napoletano et al., 2012). Feature selection methods select a feature subset from the original feature set by measuring the performance of different features or classifiers. Typical feature selection methods include information gain (IG) (Yang and Pedersen, 1997), chi-square (CHI) (Yang and Pedersen, 1997), improved Gini index (IMGI) (Shang et al., 2007), mutual information (MI) (Peng et al., 2005), comprehensively measured feature selection (CMFS) (Yang et al., 2012), orthogonal centroid feature selection (OCFS) (Yan et al., 2005), and *t*-test based feature selection (TTFS) (Wang et al., 2012). Compared with feature extraction methods, feature selection methods achieve a better interpretability and retain physical properties of the original features. Therefore, feature selection methods have been used widely in the fields of machine learning and pattern classification (Han and Ren, 2015). According to Liu and Yu (2005), wrappers and filters are two typical types of feature selection methods. Filters select features based on some agnostic criteria that measure the class discrimination abilities of different features, and they are not dedicated to any classification algorithm in their feature-selecting process. Wrappers rely on the performance of one classifier when a feature subset is used (Zhang and Zhang, 2012). Compared with filters, although wrappers can select the subsets with optimal features, they entail a high computational cost and are not suitable for text classification tasks (Han and Ren, 2015).

2 Related work

In this section, we first introduce several traditional feature selection methods, which calculate only the discrimination abilities of a word with respect to different categories. Furthermore, we introduce several MI-based feature selections that filter the redundant features based on the MI theory.

2.1 Typical traditional feature selections

2.1.1 Darmstadt indexing approach

Darmstadt indexing approach (DIA) has been used widely in automatic indexing (Sebastiani, 2002). When a document contains a word t_i, DIA evaluates the probability that the document belongs to category c_k by measuring the conditional probability. DIA is defined as

$$\mathrm{DIA}(t_i)=\sum_{c_k} p(c_k \mid t_i), \tag{1}$$

where $p(c_k|t_i)$ represents the conditional probability that a document d belongs to c_k when t_i occurs in d.

2.1.2 The IG approach

IG is a widely used measurement in the field of machine learning. This method defines the expected reduction in entropy caused by partitioning the texts according to a word (Yang and Pedersen, 1997; Zhang et al., 2011). IG is defined as

$$\mathrm{IG}(t_i)=\sum_{t\in(t_i,\bar{t}_i)}\sum_{c_k} p(t,c_k)\log_2\left(\frac{p(t,c_k)}{p(t)p(c_k)}\right), \tag{2}$$

where $p(t)$ denotes the probability that a document contains word t, $p(c_k)$ is the probability that category c_k occurs in the dataset, and $p(t,\ c_k)$ denotes the probability that t occurs in c_k.

2.1.3 Improved Gini index

Gini index is a nonpurity split method proposed by Breiman et al. (1984). To apply this method to tasks of multiclass text classification, Shang et al. (2007) improved the Gini index to the improved Gini index (IMGI):

$$\mathrm{IMGI}(t_i)=\sum_{c_k}\left[p(t_i \mid c_k)\right]^2\left[p(c_k \mid t_i)\right]^2, \tag{3}$$

where $p(t_i|c_k)$ represents the conditional probability that word t_i occurs in category c_k.

2.1.4 Improved comprehensively measured feature selection

As the traditional feature selection methods cannot achieve the best performance due to the adverse effect of an imbalanced dataset, Yang et al. (2014) improved CMFS and proposed a new method called 'the improved comprehensively measured feature selection (CMFSX)'. CMFSX can weaken the adverse effect caused by the imbalance factor in the dataset. CMFSX is defined as

$$\begin{cases} \mathrm{CMFSX}(t_i)=\max\limits_{c_k}\{\mathrm{CMFS}(t_i,c_k)\}, \\ \mathrm{CMFS}(t_i,c_k)=\dfrac{p(t_i\mid c_k)p(c_k\mid t_i)}{p(c_k)}. \end{cases} \tag{4}$$

2.2 Typical mutual information based feature selections

Traditional feature selections do not consider the relevance of different words when selecting the features. Napoletano et al. (2012) considered a hierarchical structure, named 'a mixed graph of terms (MGT)', which was constructed automatically from a set of documents through the probabilistic topic model. In this method, the relevant words are represented by pairs of related words, and a set of parameters are changed until the effectiveness is maximized. This method shows better performance than that using a list of weighted words. However, as all the relations of the words are calculated by LDA, the time complexity is high when calculating the relevance of different words. The MI-based feature selection methods are the most widely used methods for calculating the relevance of different words. MI measures the dependence information of a word t_i and a category c_k (Peng et al., 2005). The formula is given as

$$\begin{cases} \mathrm{MI}(t_i)=\sum\limits_{c_k}\mathrm{MI}(t_i,c_k), \\ \mathrm{MI}(t_i,c_k)=p(c_k,t_i)\log_2\left(\dfrac{p(c_k,t_i)}{p(c_k)p(t_i)}\right), \end{cases} \tag{5}$$

where $p(t_i)$ is the occurrence probability of word t_i and $p(c_k, t_i)$ is the probability that a document in category c_k contains t_i.

The MI-based methods obtain the feature subsets by balancing the correlations between candidate features and all categories, and the correlations between candidate features and all selected features. Herein, we give the details of four typical MI-based feature selections.

2.2.1 Mutual information feature selection

Battiti (1994) introduced the balance parameter and proposed an MI-based feature selection method based on maximal relevance. The objective function of the mutual information feature selection (MIFS) process is defined as follows:

$$\max\left\{\sum_{c_k}I(t_i,c_k)-\beta\sum_{t_j\in S}I(t_i,t_j)\right\}, \tag{6}$$

where t_i is a candidate feature, S is the set of selected features, β is a balance parameter, $I(t_i, c_k)$ denotes the MI between t_i and category c_k, and $I(t_i, t_j)$ denotes the MI between t_i and a selected feature t_j.

2.2.2 Improved mutual information feature selection

Hoque et al. (2014) introduced a greedy feature selection using MI. The method considered the MI of feature–feature and feature–class to determine an optimal set of features. The objective function of the improved MIFS (MIFS-ND) is defined as

$$\max\left\{\sum_{c_k}I(t_i,c_k)-\frac{1}{|S|}\sum_{t_j\in S}I(t_i,t_j)\right\}, \tag{7}$$

where $|S|$ is the number of features in S. Obviously, this method becomes the MIFS method when β in MIFS is equal to $1/|S|$.

2.2.3 Multilabel feature selection based on maximum dependency and minimum redundancy

In contrast to the traditional single-label learning approach, Lin et al. (2015) considered the conditional redundancy between the candidate and selected features, and thereafter presented a maximum dependency and minimum redundancy (MDMR) based feature selection method. The objective function is defined as

$$\max\left\{\sum_{c_k}I(t_i,c_k)-\frac{1}{|S|}\sum_{t_j\in S}\left(I(t_i,t_j)-\sum_{c_l}I(t_i,c_l\,|\,t_j)\right)\right\}, \tag{8}$$

where $I(t_i, c_l|t_j)$ represents the relevance between candidate feature t_i and category c_l when given the selected feature t_j.

2.2.4 Normalized mutual information based feature selection

Estevez et al. (2009) improved the MIFS measurement and proposed a normalized MI-based feature selection (NMIFS) approach to solve the problem that the entropy of a feature varied greatly. The normalization restricts the output values to the range [0, 1]. The objective function is defined as

$$\begin{cases} \max\left\{\sum_{c_k} I(t_i,c_k) - \dfrac{1}{|S|}\sum_{t_j\in S} \mathrm{NI}(t_i,t_j)\right\}, \\ \mathrm{NI}(t_i,t_j) = \dfrac{I(t_i,t_j)}{\min\left\{H(t_i),H(t_j)\right\}}, \end{cases} \tag{9}$$

where $H(t_i)$ and $H(t_j)$ are defined as

$$\begin{cases} H(t_i) = -p(t_i)\log_2 p(t_i), \\ H(t_j) = -p(t_j)\log_2 p(t_j). \end{cases} \tag{10}$$

3 Motivation

Suppose there are four categories C_1, C_2, C_3, and C_4, each of which includes 10 documents. Table 1 gives the distributions of six words ('thanks', 'look', 'forward', 'subject', 'buy', and 'http') in these categories. A value '1' in Table 1 means that the word occurs in the corresponding document of the category. Table 2 gives five selected features when three typical feature selection methods (CMFSX, MIFS-ND, and MDMR) are used. From Table 1, we know that because the distribution of 'look' in different categories is the same as that of 'forward'. 'Forward' is redundant with respect to 'look' and should be ignored when the latter one is selected. However, the CMFSX method still selects 'forward' because of the high output value of this word. Moreover, 'http', which follows feature 'buy', should be added into the feature subset as 'http' can bring extra information on category discrimination. However, 'http' is ignored by CMFSX when only five features are selected. Thus, CMFSX cannot represent the word 'http' when it occurs in a document, showing a lower accuracy than the MI-based methods, when the numbers of the selected features are equal.

Moreover, although 'forward' is filtered by MIFS-ND and MDMR, it cannot be represented by these MI-based methods when 'forward' occurs but 'look' does not in a document. In addition, denoting $|C|$ as the number of categories in a dataset, N as the number of all words in all categories, and $|S|$ as the number of selected features, Table 2 gives the time complexities of CMFSX, MIFS-ND, and MDMR.

Table 1 Distributions of six words in four categories

Word	Distributions of the words in categories			
	C_1	C_2	C_3	C_4
'Thanks'	1, 1, 1, 1, 1, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
'Look'	0, 0, 0, 0, 0, 1, 1, 1, 1, 0	1, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
'Forward'	0, 0, 0, 0, 0, 1, 1, 1, 1, 0	1, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
'Subject'	1, 1, 1, 0, 0, 0, 0, 0, 0, 0	0, 1, 1, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
'Buy'	0, 0, 0, 1, 1, 0, 0, 0, 0, 0	0, 0, 0, 1, 1, 0, 0, 0, 0, 0	1, 0, 0, 0, 0, 0, 0, 0, 0, 0	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
'http'	0, 0, 0, 0, 0, 0, 0, 0, 0, 1	0, 0, 0, 0, 0, 1, 0, 0, 0, 0	0, 1, 0, 0, 0, 0, 0, 0, 0, 0	1, 0, 0, 0, 0, 0, 0, 0, 0, 0

Table 2 Selected features and time complexities of typical feature selections

Method	Selected feature	Time complexity	MI-based method
CMFSX	'Thanks', 'look', 'forward', 'subject', 'buy'	$O(N(\|C\|+\log_2 N))$	No
MIFS-ND	'Thanks', 'look', 'subject', 'buy', 'http'	$O(\|S\|(N-\|S\|)(\|C\|+\|S\|))$	Yes
MDMR	'Thanks', 'look', 'subject', 'buy', 'http'	$O(\|S\|(N-\|S\|)(\|C\|+\|S\|(1+\|C\|)))$	Yes

CMFSX: improved comprehensively measured feature selection; MIFS-ND: improved mutual information feature selection; MDMR: maximum dependency and minimum redundancy

Obviously, because the MI-based methods (MIFS-ND and MDMR) need to compute the relevance between the candidate and selected features, their time complexities are generally higher than that of the CMFSX method.

From these arguments, we conclude that the feature selections which have been proposed in recent years still have some drawbacks: (1) Traditional feature selection methods cannot filter the redundant features, showing a lower accuracy than the MI-based methods when the numbers of the selected features are equal; (2) The words filtered by the MI-based methods cannot be represented when they occur in a document, possibly missing some helpful category discrimination information; (3) Most MI-based feature selection methods calculate the correlations not only between the candidate features and all categories, but also between the candidate and selected features; thus, the time complexities of these methods are high when the numbers of all words and the selected features are both large. On this basis, a new simple feature selection method, which can efficiently improve the speed of filtering of the redundant features while ensuring simultaneously the classification accuracy, is proposed in this study.

4 The proposed method

The key point of filtering the redundant features is to measure the relevance of the words. MI is a widely used method that can calculate the statistical relationship between two variables (Battiti, 1994; Peng et al., 2005; Lin et al., 2015). According to Eq. (5), the MI between words t_i and t_j is computed as

$$\mathrm{MI}(t_i,t_j)=p(t_i,t_j)\log_2\left(\frac{p(t_i,t_j)}{p(t_i)p(t_j)}\right). \tag{11}$$

However, according to Estevez et al. (2009), the output of this method varies greatly and is not restricted to range [0, 1]. Moreover, this method focuses on the occurrence probability of the document that contains both t_i and t_j, ignoring the relevance between the frequencies of words t_i and t_j in each category. On this basis, we define a simple measurement called 'word frequency based relevance' to measure the relationship between any two words.

Definition 1 (Word frequency based relevance) Assuming that TS is a sample set that contains N_d samples, the word frequency based relevance between t_i and t_j ($i\neq j$) in TS is defined as follows:

$$\mathrm{FR}(t_i,t_j)=p(t_j\mid t_i)p(t_i\mid t_j)\mathrm{tf}_{ij}, \tag{12}$$

$$\mathrm{tf}_{ij}=\sum_c\frac{n_c}{n}\cdot\frac{\min(\mathrm{tf}_{i,c},\mathrm{tf}_{j,c})}{\max(\mathrm{tf}_{i,c},\mathrm{tf}_{j,c})}, \tag{13}$$

where $p(t_i|t_j)$ denotes the conditional probability that t_i occurs when t_j occurs in a document, $p(t_j|t_i)$ denotes the conditional probability that t_j occurs when t_i occurs in a document, tf_{ij} denotes the term frequency relevance between t_i and t_j, n_c is the number of documents in category c, n is the number of documents in the dataset, and $\mathrm{tf}_{i,c}$ and $\mathrm{tf}_{j,c}$ are the word frequencies of t_i and t_j in category c, respectively.

Obviously, from Eq. (13), we have

$$\mathrm{tf}_{ji}=\sum_c\frac{n_c}{n}\cdot\frac{\min(\mathrm{tf}_{j,c},\mathrm{tf}_{i,c})}{\max(\mathrm{tf}_{j,c},\mathrm{tf}_{i,c})}=\mathrm{tf}_{ij}. \tag{14}$$

Furthermore, by combining Eqs. (14) and (12), we have

$$\begin{aligned}\mathrm{FR}(t_j,t_i)&=p(t_i\mid t_j)p(t_j\mid t_i)\mathrm{tf}_{ji}\\&=p(t_j\mid t_i)p(t_i\mid t_j)\mathrm{tf}_{ij}\\&=\mathrm{FR}(t_i,t_j).\end{aligned} \tag{15}$$

From the definition of word frequency based relevance, we know that if t_i and t_j occur in the same document frequently and if the word frequencies of t_i and t_j in each category are similar, then t_i and t_j will have a high word frequency based relevance score. On this basis, we give the definition of correlatively redundant words as follows:

Definition 2 (Correlatively redundant words) Assuming that t_i and t_j are two words in training set TS, t_i and t_j are correlatively redundant (denoted as $t_i\equiv t_j$) if and only if the word frequency based relevance score between t_i and t_j is higher than a predetermined threshold th, namely,

$$t_i\equiv t_j\Leftrightarrow\mathrm{FR}(t_i,t_j)\geq\mathrm{th},\ \mathrm{th}\in(0,1]. \tag{16}$$

On this basis, we propose a new simple feature selection method that can efficiently improve the

speed of filtering of the redundant features while ensuring accuracy. The core idea of this method is to obtain a feature subset FS_1 by using an optimal feature selection (OFS) method and to filter the redundant features from FS_1 to form the final feature subset FS by calculating the word frequency based relevance scores of the features. Unlike the traditional feature selection methods, to represent the filtered words, we memorize the filtered features using linked lists. The details of the proposed method are shown in Algorithm 1.

Algorithm 1 The proposed feature selection method

Input: training sample set (TS), the number of temporary feature subsets (N_1), predetermined threshold th, and a temporary array $A=\{a_i\}=\{0\}$ ($0\le i\le N_1-1$).
Output: final feature subset (FS).

1: Obtain the OFS value of each word in TS using the OFS method;
2: Rank all words by their OFS values in descending order and select the top N_1 features to form a subset $FS_1=\{t_0, t_1, \ldots, t_i, t_{i+1}, \ldots, t_j, t_{j+1}, \ldots, t_{N_1-1}\}$, where $FS(t_i)\ge OFS(t_j)$ ($0\le i<j\le N_1-1$);
// Filter the redundant features in FS_1:
3: Set a temporary variable $k=0$;
4: **for** $i=0$ to N_1-1
5: **if** $a_i=1$ **then continue**;
6: **end if**
7: Create a linked list l_k, and memorize t_i into the head node of l_k;
8: $a_i\leftarrow 1$;
9: **for** $j=i+1$ to N_1-1
10: **if** $a_j=1$ **then continue**;
11: **end if**
12: **if** $t_i\equiv t_j$ **then** create a new node and add it to l_k to memorize t_j;
13: $a_j\leftarrow 1$;
14: **end if**
15: **end for**
16: k++;
17: **end for**

As shown in Algorithm 1, temporary array A is used to mark the filtered redundant features in FS_1. In linked list l_k, the head node memorizes the selected features, and the other nodes memorize the words that are redundant with respect to the selected feature. For example, according to steps 1 and 2, we select 10 words from TS and represent them as $FS_1=\{t_0, t_1, t_2, \ldots, t_9\}$. Furthermore, according to step 3, we filter the redundant features from FS_1 by combining a predetermined threshold th (assuming th=0.95). Fig. 1 shows the processes of selecting the features from FS_1 when loop control variable i changes. Moreover, with respect to Fig. 1, Table 3 gives the corresponding details of the intermediate results in each process of Algorithm 1. We can easily see from Fig. 1 and Table 3 that the final selected feature subset is $FS=\{t_0, t_3, t_5, t_9\}$.

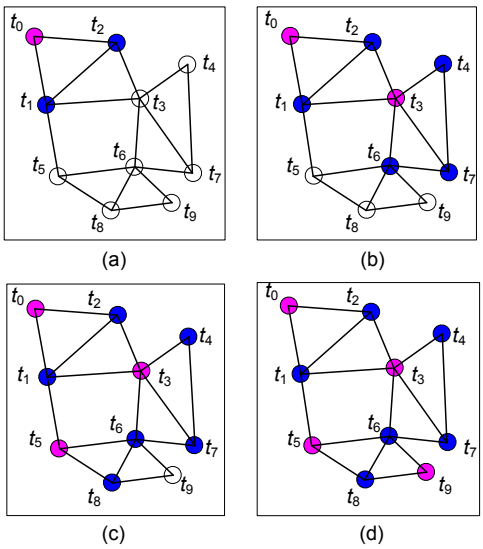

Fig. 1 Changes in selected and filtered features in Algorithm 1 (th=0.95): (a) i=0; (b) i=3; (c) i=5; (d) i=9
A solid line denotes that the two corresponding words are correlatively redundant, the pink nodes denote the selected features, and the blue nodes denote the filtered features (References to color refer to the online version of this figure)

From Algorithm 1, we know that the time complexity of the proposed method contains mainly two kinds: the time complexity of obtaining the feature subset FS_1 by the OFS method and that of filtering the redundant features in FS_1. From Yang et al. (2012), we know that the time complexity of obtaining the feature subset FS_1 is $O(|C|N+N\log_2 N)$. From Algorithm 1, we know that there are N_1^2 calls to filter the redundant features; thus, the time complexity of this step is $O(N_1^2)$. Accordingly, the time complexity of the proposed method is $O(|C|N+N\log_2 N+N_1^2)$. By combining Table 2, because the number of redundant words in FS_1 is relatively small, there usually exist $S\approx N_1$, $S\ll N$, and $\log_2 N\ll S$ in practice. Therefore, we can conclude that the time complexity of the proposed method is slightly higher than that of CMFSX but significantly lower than that of typical MI-based feature selection methods such as MIFS-ND and MDMR.

Table 3 Intermediate results in Algorithm 1

Loop control variable i	Temporary array A	Linked list	Subset of selected features	Subset of filtered features
0	{0, 1, 1, 0, 0, 0, 0, 0, 0, 0}	l_0: t_0 → t_1 → t_2	t_0	t_1, t_2
3	{0, 1, 1, 0, 1, 0, 1, 1, 0, 0}	l_0: t_0 → t_1 → t_2 l_1: t_3 → t_4 → t_6 → t_7	t_0, t_3	t_1, t_2, t_4, t_6, t_7
5	{0, 1, 1, 0, 1, 0, 1, 1, 1, 0}	l_0: t_0 → t_1 → t_2 l_1: t_3 → t_4 → t_6 → t_7 l_2: t_5 → t_8	t_0, t_3, t_5	t_1, t_2, t_4, t_6, t_7, t_8
9	{0, 1, 1, 0, 1, 0, 1, 1, 1, 0}	l_0: t_0 → t_1 → t_2 l_1: t_3 → t_4 → t_6 → t_7 l_2: t_5 → t_8 l_3: t_9	t_0, t_3, t_5, t_9	t_1, t_2, t_4, t_6, t_7, t_8

Traditional MI-based feature selection methods cannot represent the words that are redundant with respect to the selected features. In this study, we memorize the filtered features in linked lists and use these linked lists to replace the redundant words of a document in the preprocessing process before document classification. Given a document $d=\{t_0, t_1, \ldots, t_i, t_{i+1}, \ldots, t_{\mathrm{nw}-1}\}$ (nw is the number of all words in d) and the set of linked lists $\mathrm{LS}=\{l_0, l_1, \ldots, l_i, l_{i+1}, \ldots, l_{\mathrm{nl}-1}\}$ (nl is the number of linked lists obtained in Algorithm 1), the preprocessing process is described in Algorithm 2.

Algorithm 2 Preprocessing process before document classification

Input: a document $d=\{t_0, t_1, \ldots, t_i, t_{i+1}, \ldots, t_{\mathrm{nw}-1}\}$ and the set of linked lists $\mathrm{LS}=\{l_0, l_1, \ldots, l_i, l_{i+1}, \ldots, l_{\mathrm{nl}-1}\}$.
Output: the document that is preprocessed.

```
for i=0 to nw−1
  for j=0 to nl−1
    if t_i is contained in l_j and is not equal to the word
       that is contained in the head node of l_j
    then t_i is replaced by the word that is contained in
       the head node of l_j;
    end if
  end for
end for
```

5 Experimental settings

5.1 Datasets

Three benchmark datasets, WebKB (WE) (Yang et al., 2012), 20-Newsgroups (NE) (Zhang and Zhang, 2012), and Reuters-21578 (RE) (Zhang and Zhang, 2012), were used in our experiments. WebKB consists of 8282 web pages collected from the websites of different colleges and is divided into seven categories. The 20-Newsgroups dataset consists of 19 997 documents collected from Newsgroup postings and is assigned evenly to 20 different categories. The Reuters-21578 dataset consists of 21 578 stories collected from the Reuters newswire and is divided into 135 categories. For ease of computation, we considered only the top 4, 6, and 10 categories with respect to WE, NE, and RE, respectively. For the documents in each dataset, we executed the stemming process using the Porter stemming algorithm before the training and classification processes (Porter, 1997). Moreover, we used 10-fold cross-validation (CV) to compare different methods.

5.2 Classifiers

Typical classifiers such as SVMs (Joachims, 1998; Drucker et al., 1999) and NBs (Cevenini et al., 2013) have been used to compare different feature selections. In this study, these classifiers were implemented by Weka 3.6.9, which is a popular platform for machine learning and data mining. We used the multinomial event model to classify a document when the NB classifier was used (Schneider, 2003). To specify the parameters of SVM, LIBSVM provides a simple tool to check a grid of parameters by using the one-against-one method to obtain CV accuracy (Chang and Lin, 2007). According to Chang and Lin (2007), the best parameters of SVM were given as

follows: Kernel=RBF kernel, cost parameter $c=2^0=1.0$, $\gamma=2^{-7}\approx0.008$, and tolerance of the termination criterion $e=0.001$.

5.3 Performance measurement

According to Yang et al. (2014), precision p and recall r are two widely used methods to measure the effectivenesses of feature selection methods, denoted as

$$\begin{cases} p=\dfrac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}=\dfrac{\sum\limits_{i=1}^{|C|}\mathrm{TP}_i}{\sum\limits_{i=1}^{|C|}\mathrm{TP}_i+\sum\limits_{i=1}^{|C|}\mathrm{FP}_i}, \\ r=\dfrac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}=\dfrac{\sum\limits_{i=1}^{|C|}\mathrm{TP}_i}{\sum\limits_{i=1}^{|C|}\mathrm{TP}_i+\sum\limits_{i=1}^{|C|}\mathrm{FN}_i}, \end{cases} \tag{17}$$

where TP_i is the number of documents that are correctly classified to category C_i, FP_i is the number of documents that are misclassified to category C_i, and FN_i is the number of documents that belong to category C_i and are misclassified. In this study, we combine the effectiveness of precision and recall and use the F_1 measurement to validate the performance of different methods, defined by

$$F_1=\frac{2\cdot r\cdot p}{r+p}. \tag{18}$$

5.4 Selections of the optimal feature selection method and threshold th

As Yang et al. (2014) showed that CMFSX runs fast and is significantly superior to methods such as IG, IMGI, and DIA, CMFSX was chosen as the OFS method of the proposed method. Moreover, we used the harmony search (HS) algorithm to search for the optimal value of th and improve the trade-off between the numbers of the selected features and those of the filtered features. HS was recently developed in an analogy with the music improvisation process, whereby music players improvised the pitches of their instruments to obtain better harmony (Geem et al., 2001). Compared with those of other traditional metaheuristic algorithms, the results of the HS method do not rely on the initial settings of the decision variables, thus probably avoiding the local optima. Moreover, HS has a good global searching ability as it generates a new solution using all existing solutions, whereas the other algorithms (such as fruit fly algorithm and particle swarm optimization) consider only the global best solution (Alatas, 2010). In our previous work (Wang et al., 2014), we introduced a factor called 'the best harmony considering rate (BHCR)' and proposed a global best harmony search (GBHS) algorithm to improve the convergence characteristic of the traditional HS method. Because GBHS accelerates the convergence rate significantly and achieves a higher accuracy than typical metaheuristic algorithms, we use a GBHS to search for the optimal th in this study. The corresponding parameters of GBHS are given as

1. Objective function:

$$F(\mathrm{th})=\frac{1}{10}\sum_{n_\mathrm{s}}F_1(n_\mathrm{s},\mathrm{th}), \tag{19}$$

where n_s is the number of selected features, which ranges from 100 to 1000, with a step of 100, and $F_1(n_\mathrm{s}, \mathrm{th})$ denotes the performance of a classifier measured by the F_1 measurement when th and n_s are used.

2. Parameters in GBHS (Table 4)

Table 4 Parameters in global best harmony-oriented harmony search (GBHS)

Explanation	Parameter
Number of vectors in the harmony memory	N=1
Harmony memory size	HMS=40
Maximum harmony memory considering rate	$\mathrm{HMCR_{max}}$=0.9
Minimum harmony memory considering rate	$\mathrm{HMCR_{min}}$=0.5
Maximum best harmony considering rate	$\mathrm{BHCR_{min}}$=0.5
Maximum pitch adjusting rate	$\mathrm{PAR_{max}}$=0.99
Minimum pitch adjusting rate	$\mathrm{PAR_{min}}$=0.01
Maximum arbitrary distance bandwidth	$\mathrm{bw_{max}}$=1
Minimum arbitrary distance bandwidth	$\mathrm{bw_{min}}=0.5\times10^{-6}$
Maximum number of improvisations	NI=1000
Harmony memory	$\begin{cases}\mathbf{HM}=\left[x_1^1,x_1^2,\cdots,x_1^j,x_1^{j+1},\right.\\ \qquad\cdots x_1^{\mathrm{HMS}-1},x_1^{\mathrm{HMS}}\left.\right]^{\mathrm{T}},\\ x_1^j=\mathrm{Rand}(),\ j=1,2,\cdots,\mathrm{HMS}.\end{cases}$

6 Experimental results and analysis

6.1 Comparison of different methods in terms of execution time

We compare the proposed method with several typical feature selection methods (DIA, IG, IMGI, CMFSX, MIFS, MIFS-ND, NMIFS, and MDMR) on WE, NE, and RE datasets when the number of selected features ranges from 200 to 1000 with a step of 200. With respect to the optimal th of different classifiers, the average execution time (denoted as et, expressed in seconds) is shown in Figs. 2–4. We know from these figures that the et values of traditional feature selections (DIA, IG, IMGI, and CMFSX) are much lower than those of MI-based feature selections (MIFS, MIFS-ND, NMIFS, and MDMR), showing the disadvantage of the latter methods in terms of execution speed. Moreover, MDMR is the most time-consuming method, which may be because MDMR considers the conditional redundancies between the candidate features and the selected features. For each dataset, as the number of selected features increases, the performance of traditional feature selections slightly changes; however, the et values of MI-based feature selections increase significantly, especially when MDMR is used on the RE dataset. Furthermore, we notice that the results for the RE dataset are generally higher than those for the NE and WE datasets. It may be because RE has more categories than the other datasets, and the number of categories affects the execution speed seriously. It is noteworthy that the et values of the proposed method are slightly higher than those of DIA, IG, IMGI, and CMFSX but significantly lower than those of MIFS, MIFS-ND, NMIFS, and MDMR, illustrating the efficiency of the proposed method in filtering redundant features.

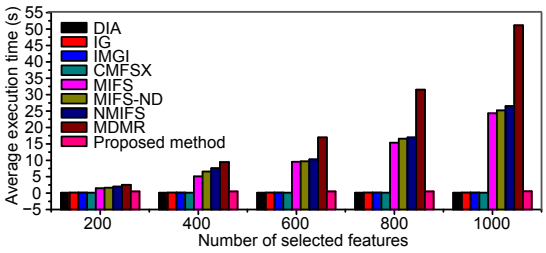

Fig. 2 Average execution time (et) of different feature selection methods on the WebKB dataset (References to color refer to the online version of this figure)

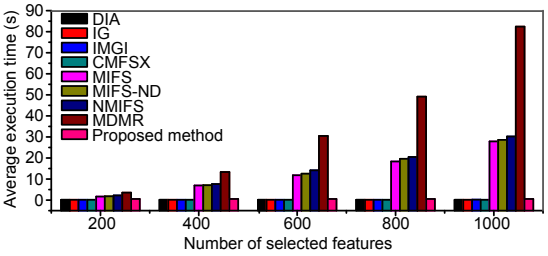

Fig. 3 Average execution time (et) of different feature selection methods on the 20-Newsgroups dataset (References to color refer to the online version of this figure)

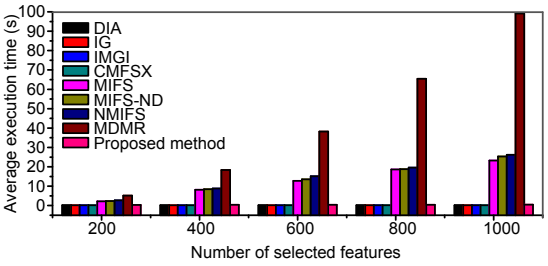

Fig. 4 Average execution time (et) of different feature selection methods on the Reuters-21578 dataset (References to color refer to the online version of this figure)

6.2 Comparison of different methods in terms of classification accuracy

6.2.1 Results for the WebKB dataset

Fig. 5 shows the F_1 values of the different feature selection methods on the WE dataset when SVM and NB are used. As the selected features of the proposed method do not contain redundant information, the F_1 values of the proposed method are generally higher than those of traditional methods such as IG, IMGI, and CMFSX. When SVM is used, the proposed method outperforms other methods when 600 or 800 features are selected. Significantly, the proposed method obtains the highest F_1 (0.885) with SVM when the number of selected features equals 800. In addition, when NB is used, although the proposed method is inferior to those of MI-based methods such as NMIFS and MDMR when more than 600 features are selected, it still outperforms other methods when 200 or 400 features are selected, illustrating the efficiency of the proposed method in classifying the web pages of the WE dataset.

6.2.2 Results on the 20-Newsgroups dataset

Fig. 6 shows the F_1 values of seven feature-selection methods on the NE dataset when SVM

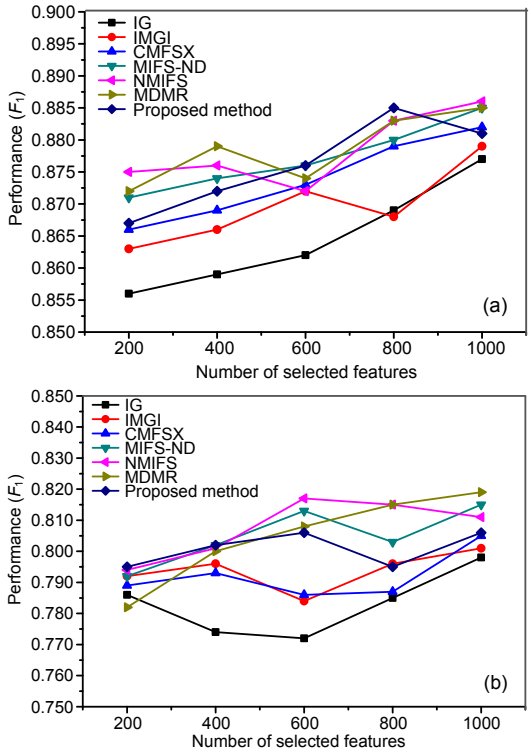

Fig. 5 Performance (values of F_1) of different feature selection methods on the WebKB dataset when support vector machine (SVM) (a) and naïve Bayes (NB) (b) are used (References to color refer to the online version of this figure)

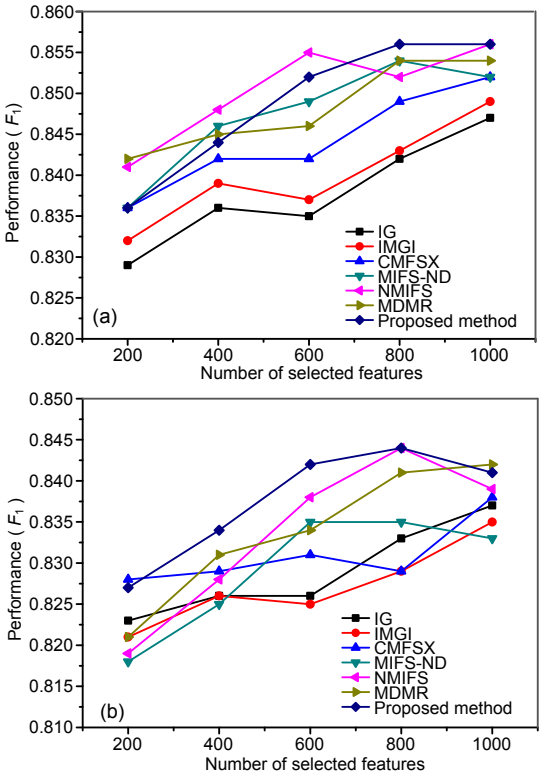

Fig. 6 Performance (values of F_1) of different feature selection methods on the 20-Newsgroups dataset when support vector machine (SVM) (a) and naïve Bayes (NB) (b) are used (References to color refer to the online version of this figure)

and NB are used. When SVM is used, IG shows the worst performance. MDMR, NMIFS, and the proposed method obtains the highest the F_1 values for once, twice, and twice, respectively. It is noteworthy that the proposed method obtains the global highest F_1 (0.856) when 800 (1000) features are selected. When NB is used, F_1 of the proposed method increases as the number of selected features ranges from 200 to 800. Moreover, the proposed method performs significantly better than other methods except when 200 or 1000 features are selected, with the highest improvement (0.015) over that of CMFSX when 800 features are selected.

6.2.3 Results on the Reuters-21578 dataset

Fig. 7 shows the F_1 values of different feature selection methods on the RE dataset when SVM and NB are used. It can be seen that, when SVM is used, the F_1 values of the proposed method are generally higher than those of the other methods, except when the number of selected features equals 400. Obviously, the proposed method obtains the global highest F_1 value of 0.862 when the number of the selected features equals 1000. When NB is used, the F_1 values of all methods gradually increase as the number of the selected features increases. The proposed method has better output results than the other methods when the number of the selected features equals 200, 600, or 1000, with the highest improvement (0.010) over CMFSX when 400 features are selected.

6.3 Comparison of the proposed method and the optimal feature selection methods

The OFS method affects the performance of the proposed method. To show the improvement of the proposed method over the OFS method, different typical traditional feature selections, such as document frequency (DF) (Yang et al., 2012), IG, DIA, and MI, were used as the OFS methods for experiments. We denote the execution time increments and the

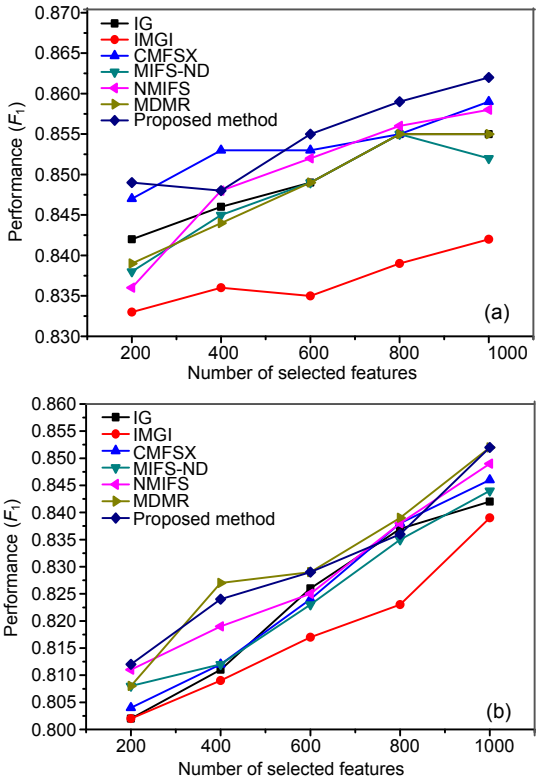

Fig. 7 Performance (values of F_1) of different feature selection methods on the Reuters-21578 dataset when support vector machine (SVM) (a) and naïve Bayes (NB) (b) are used (References to color refer to the online version of this figure)

classification accuracy improvements of the proposed method over the OFS method as T_I and F_I, respectively. When the number of the selected features ranges from 200 to 1000 with a step of 200, T_I and F_I are defined as

$$\begin{cases} T_I = \frac{1}{5}\sum_n \big(T(n) - T_{OFS}(n)\big), \\ F_I = \frac{1}{5}\sum_n \big(F(n) - F_{OFS}(n)\big), \end{cases} \tag{20}$$

where $T(n)$ and $T_{OFS}(n)$ are the execution time of the proposed method and the OFS method, respectively, when n features are selected, and $F(n)$ and $F_{OFS}(n)$ are the F_1 values of a classifier when n features are selected by the proposed method and the OFS method, respectively. On this basis, Table 5 gives the T_I values when different methods are used as OFS, and Table 6 gives the corresponding the F_I values when SVM and NB are applied on the WE, NE, and RE datasets, respectively. We can see from Table 5 that, all the T_I values are smaller than 0.5 s when different OFS methods are used, which means that the proposed method spends little time on filtering redundant features. From Table 6, we know that when SVM is used, the F_I values are larger than zero in 95.83% of the cases (23 of 24 cases), with the largest F_I of 0.024 when OFS is initialized to CHI on the RE dataset. When NB is used, the F_I values are larger than zero in 91.67% of the cases (22 of 24 cases), with the highest F_I of 0.021 when OFS is initialized to MI on the WE dataset. In evidence, the proposed method brings a high classification accuracy at the cost of slight increments of execution time, illustrating the effectiveness of the proposed method in selecting the most discriminative features.

6.4 Statistical results and discussions

The nonparametric tests are used widely in statistical learning. The Wilcoxon signed-rank test is usually powerful in detecting the difference between two populations (Wilcoxon, 1945; Taheri and Hesamian, 2013). In this section, the Wilcoxon signed-rank test is used, and the null hypothesis is that the two methods are equivalent. If the null hypothesis is rejected (p-value is less than or equal to significance level α), the differences between the methods are significant. To further validate the effectiveness of the proposed method, different methods are used for comparisons from the aspects of execution time and classification accuracy.

1. To compare the execution speeds of different methods, the difference in execution time between method i and the proposed method is denoted as t_{di}, which is measured by

$$t_{di} = t_i - t_S, \tag{21}$$

where t_i and t_S are the execution time of method i and the proposed method, respectively. Moreover, the experiments were carried out 10 times when the number of selected features ranges from 100 to 1000 with a step of 100, and the average of t_{di}, (denoted as t_{da}) for each method is computed and shown in Table 7.

2. Given significance level α=0.05, the F_1 values of the proposed method and those of the other methods are compared using the Wilcoxon signed-rank test when the feature number ranges from 100 to 1000 with a step of 100. Moreover, experiments were

carried out 10 times when SVM and NB are applied, and the average of *p*-values (denoted as p_a) is shown in Table 8.

We can see from Table 7 that the execution time of the proposed method is slightly higher than that of traditional feature selections as the corresponding t_{da} values are less than zero. Moreover, as the proposed method does not need to calculate the correlations between the candidate and the selected features, it runs much faster than the MI-based methods, with the highest t_{da} of 46.374 in the case of MDMR on the RE dataset. When comparing the classification accuracy of different methods, we can see from Table 8 that the p_a values are lower than α in 26 of 36 cases, showing that the proposed method outperforms other methods in 72.2% of the cases in terms of classification accuracy. As the traditional feature selection methods (IG, IMGI, and CMFSX) cannot filter the redundant features, they perform worse than the proposed method in all cases. Moreover, the proposed method outperforms the MI-based methods (MIFS-ND, NMIFS, and MDMR) more than twofold, illustrating the effectiveness of the proposed method in ensuring classification accuracy. Overall, we conclude from Tables

Table 5 Increments in execution time (T_I) of the proposed method over different optimal feature selection methods

Dataset	T_I of the proposed method over different optimal feature selection methods (s)							
	DIA	IG	IMGI	OCFS	CHI	MI	DF	CMFSX
WE	0.352	0.362	0.349	0.356	0.362	0.361	0.359	0.355
NE	0.357	0.355	0.353	0.339	0.347	0.368	0.356	0.362
RE	0.209	0.217	0.192	0.203	0.218	0.214	0.198	0.204

WE: WebKB; NE: 20-Newsgroups; RE: Reuters-21578

Table 6 Improvements in classification accuracy (F_I) of the proposed method over different optimal feature selection methods

Classifier	Dataset	F_I of the proposed method over different optimal feature selection methods							
		DIA	IG	IMGI	OCFS	CHI	MI	DF	CMFSX
SVM	WE	0.015	0.003	0.004	0.017	0.002	0.017	0	0.015
	NE	0.023	0.013	0.021	0.012	0.011	0.017	0.008	0.005
	RE	0.019	0.017	0.003	0.009	0.024	0.003	0.003	0.012
NB	WE	0.007	0.015	0.004	0.007	0.018	0.019	0.002	0.009
	NE	0.005	0.006	0.009	0.014	0.002	0.012	0.007	0.005
	RE	0.014	0	0.008	0	0.008	0.021	0.011	0.007

SVM: support vector machine; NB: naïve Bayes; WE: WebKB; NE: 20-Newsgroups; RE: Reuters-21578

Table 7 The average of the difference in execution time (t_{da}) between the proposed method and other methods

Dataset	t_{da} between the proposed method and other methods (s)					
	IG	IMGI	CMFSX	MIFS-ND	NMIFS	MDMR
WE	−0.375	−0.492	−0.458	5.556	8.983	12.136
NE	−0.427	−0.453	−0.611	10.139	15.782	33.552
RE	−0.439	−0.517	−0.419	14.227	16.694	46.374

WE: WebKB; NE: 20-Newsgroups; RE: Reuters-21578

Table 8 The average of *p*-values (p_a) between the proposed method and other methods

Classifier	p_a between the proposed method and other methods						
	Dataset	IG	IMGI	CMFSX	MIFS-ND	NMIFS	MDMR
SVM	WE	0	0	0	0.332	0.408	0.439
	NE	0	0	0	0.025	0.195	0.387
	RE	0.002	0	0.042	0.014	0.029	0.036
NB	WE	0	0	0	0.291	0.535	0.366
	NE	0	0	0	0	0.008	0.022
	RE	0.032	0	0.037	0.008	0.078	0.268

SVM: support vector machine; NB: naïve Bayes; WE: WebKB; NE: 20-Newsgroups; RE: Reuters-21578

7 and 8 that the proposed method is more advantageous than traditional methods in terms of execution speed, and filters the redundant information of the selected features effectively.

7 Conclusions

To identify an efficient feature selection method that has a low time complexity and a high classification accuracy, we proposed a new simple feature selection method in this study. We introduced the definitions of word frequency based relevance and correlative redundancy, and thereafter used an optimal feature selection (OFS) method to select a feature subset FS_1. Moreover, we filtered the redundant features contained in FS_1 by combining a predetermined threshold and memorized the filtered features using the data structure of linked lists. By comparing the proposed method with several typical feature selection methods on three datasets (WebKB, 20-Newsgroups, and Reuters-21578), we found that the proposed method can effectively enhance the performances in terms of classification accuracy and execution speed when compared with typical traditional feature selection methods and typical MI-based feature selection methods.

References

Alatas B, 2010. Chaotic harmony search algorithms. *Appl Math Comput*, 216(9):2687-2699. https://doi.org/10.1016/j.amc.2010.03.114

Apte C, Damerau F, Weiss S, 1999. Text mining with decision trees and decision rules. Conf on Automated Learning and Discovery, p.169-198.

Battiti R, 1994. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans Neur Netw*, 5(4):537-550. https://doi.org/10.1109/72.298224

Breiman L, Friedman JH, Olshen RA, et al., 1984. Classification and Regression Trees. Wadsworth International Group, Monterey, USA.

Caruana G, Li MZ, Liu Y, 2013. An ontology enhanced parallel SVM for scalable spam filter training. *Neurocomputing*, 108:45-57. https://doi.org/10.1016/j.neucom.2012.12.001

Cevenini G, Barbini E, Massai MR, et al., 2013. A naïve Bayes classifier for planning transfusion requirements in heart surgery. *J Eval Clin Pract*, 19(1):25-29. https://doi.org/10.1111/j.1365-2753.2011.01762.x

Chang CC, Lin CJ, 2007. LIBSVM: a library for support vector machines. *ACM Trans Intell Syst Technol*, 2(3), Article 27. https://doi.org/10.1145/1961189.1961199

Chen JN, Huang HK, Tian SF, et al., 2009. Feature selection for text classification with naïve Bayes. *Exp Syst Appl*, 36(3):5432-5435. https://doi.org/10.1016/j.eswa.2008.06.054

Dallachiesa M, Palpanas T, Ilyas IF, 2014. Top-*k* nearest neighbor search in uncertain data series. *Proc VLDB Endowm*, 8(1):13-24. https://doi.org/10.14778/2735461.2735463

De Souza AF, Pedroni F, Oliveira E, et al., 2009. Automated multi-label text categorization with VG-RAM weightless neural networks. *Neurocomputing*, 72(10-12):2209-2217. https://doi.org/10.1016/j.neucom.2008.06.028

Drucker H, Wu DH, Vapnik VN, 1999. Support vector machines for spam categorization. *IEEE Trans Neur Netw*, 10(5):1048-1054. https://doi.org/10.1109/72.788645

Elghazel H, Aussem A, Gharroudi O, et al., 2016. Ensemble multi-label text categorization based on rotation forest and latent semantic indexing. *Exp Syst Appl*, 57:1-11. https://doi.org/10.1016/j.eswa.2016.03.041

Estevez PA, Tesmer M, Perez CA, et al., 2009. Normalized mutual information feature selection. *IEEE Trans Neur Netw*, 20(2):189-201. https://doi.org/10.1109/TNN.2008.2005601

Geem ZW, Kim JH, Loganathan GV, 2001. A new heuristic optimization algorithm: harmony search. *Simulation*, 76(2): 60-68. https://doi.org/10.1177/003754970107600201

Han M, Ren WJ, 2015. Global mutual information-based feature selection approach using single-objective and multi-objective optimization. *Neurocomputing*, 168:47-54. https://doi.org/10.1016/j.neucom.2015.06.016

Hoque N, Bhattacharyya DK, Kalita JK, 2014. MIFS-ND: a mutual information-based feature selection method. *Exp Syst Appl*, 41(14):6371-6385. https://doi.org/10.1016/j.eswa.2014.04.019

Jing LP, Ng MK, Huang JZ, 2010. Knowledge-based vector space model for text clustering. *Knowl Inform Syst*, 25(1):35-55. https://doi.org/10.1007/s10115-009-0256-5

Joachims T, 1998. Text categorization with support vector machines: learning with many relevant features. Proc 10th European Conf on Machine Learning, p.137-142. https://doi.org/10.1007/BFb0026683

Kruskal JB, Wish M, 1978. Multidimensional Scaling. Sage, London, UK.

Lin YJ, Hu QH, Liu JH, et al., 2015. Multi-label feature selection based on max-dependency and min-redundancy. *Neurocomputing*, 168:92-103. https://doi.org/10.1016/j.neucom.2015.06.010

Liu H, Yu L, 2005. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans Knowl Data Eng*, 17(4):491-502. https://doi.org/10.1109/TKDE.2005.66

McCallum A, Nigam K, 2001. A comparison of event models for naive Bayes text classification. AAAI-98 Workshop on Learning for Text Categorization, p.41-48.

Napoletano P, Colace F, De Santo M, et al., 2012. Text classification using a graph of terms. 6th Int Conf on Complex,

Intelligent and Software Intensive Systems. p.1030-1035. https://doi.org/10.1109/CISIS.2012.183

Peng HC, Long FH, Ding C, 2005. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans Patt Anal Mach Intell*, 27(8):1226-1238. https://doi.org/10.1109/TPAMI.2005.159

Porter MF, 1997. An algorithm for suffix stripping. In: Jones KS, Willett P (Eds.), Readings in Information Retrieval. Morgan Kaufmann Publishers Inc., San Francisco, USA, p.313-316.

Schneider KM, 2003. A comparison of event models for naive Bayes anti-spam e-mail filtering. Proc 10th Conf on European Chapter of the Association for Computational Linguistics, p.307-314. https://doi.org/10.3115/1067807.1067848

Sebastiani F, 2002. Machine learning in automated text categorization. *ACM Comput Surv*, 34(1):1-47. https://doi.org/10.1145/505282.505283

Shang WQ, Huang HK, Zhu HB, et al., 2007. A novel feature selection algorithm for text categorization. *Exp Syst Appl*, 33(1):1-5. https://doi.org/10.1016/j.eswa.2006.04.001

Taheri SM, Hesamian G, 2013. A generalization of the Wilcoxon signed-rank test and its applications. *Stat Paper*, 54(2):457-470. https://doi.org/10.1007/s00362-012-0443-4

Tenenhaus M, Vinzi VE, Chatelin YM, et al., 2005. PLS path modeling. *Comput Stat Data Anal*, 48(1):159-205. https://doi.org/10.1016/j.csda.2004.03.005

Uğuz H, 2011. A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowl-Based Syst*, 24(7):1024-1032. https://doi.org/10.1016/j.knosys.2011.04.014

Wang DQ, Zhang H, Liu R, et al., 2012. Feature selection based on term frequency and T-test for text categorization. Proc 21st ACM Int Conf on Information and Knowledge Management, p.1482-1486. https://doi.org/10.1145/2396761.2398457

Wang YW, Liu YN, Feng LZ, et al., 2014. Novel feature selection method based on harmony search for email classification. *Knowl-Based Syst*, 73:311-323. https://doi.org/10.1016/j.knosys.2014.10.013

Wilcoxon F, 1945. Individual comparisons by ranking methods. *Biom Bull*, 1(6):80-83. https://doi.org/10.2307/3001968

Yang JM, Liu YN, Zhu XD, et al., 2012. A new feature selection based on comprehensive measurement both in inter-category and intra-category for text categorization. *Inform Process Manag*, 48(4):741-754. https://doi.org/10.1016/j.ipm.2011.12.005

Yan J, Liu N, Zhang B, et al., 2005. OCFS: optimal orthogonal centroid feature selection for text categorization. Int ACM SIGIR Conf on Research and Development in Information Retrieval, p.122-129. https://doi.org/10.1145/1076034.1076058

Yang JM, Qu ZY, Liu ZY, 2014. Improved feature-selection method considering the imbalance problem in text categorization. *Sci World J*, 2014:625342. https://doi.org/10.1155/2014/625342

Yang YM, Pedersen JO, 1997. A comparative study on feature selection in text categorization. Proc 14th Int Conf on Machine Learning, p.412-420.

Zhang W, Yoshida T, Tang XJ, 2011. A comparative study of TF*IDF, LSI and multi-words for text classification. *Exp Syst Appl*, 38(3):2758-2765. https://doi.org/10.1016/j.eswa.2010.08.066

Zhang W, Clark RAJ, Wang YY, et al., 2016. Unsupervised language identification based on latent Dirichlet Allocation. *Comput Speech Lang*, 39:47-66. https://doi.org/10.1016/j.csl.2016.02.001

Zhang YS, Zhang ZG, 2012. Feature subset selection with cumulate conditional mutual information minimization. *Exp Syst Appl*, 39(5):6078-6088. https://doi.org/10.1016/j.eswa.2011.12.003