A Novel Feature Selection Method in the Categorization of Imbalanced Textual Data

Jafar Pouramini[1], Behrouze Minaei-Bidgoli[2], Mahdi Esmaeili[3]
[1]A PhD. Student in Department of Computer and Information Technology Engineering, Faculty of Engineering, University of Qom, Qom, Iran
[e-mail:j_pouramini@pnu.ac.ir]
[2]Faculty of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
[e_mail:b_minaei@iust.ac.ir]
[3]Faculty of Computer Engineering, Kashan Islamic Azad University, Kashan, Iran
[e-mail:m.esmaeili@iaukashan.ac.ir]
*Corresponding author: Behrouze Minaei-Bidgoli

Received July 24, 2017; revised November 19, 2017; revised December 17, 2017; accepted February 17, 2018; published August 31, 2018

Abstract

Text data distribution is often imbalanced. Imbalanced data is one of the challenges in text classification, as it leads to the loss of performance of classifiers. Many studies have been conducted so far in this regard. The proposed solutions are divided into several general categories, include sampling-based and algorithm-based methods. In recent studies, feature selection has also been considered as one of the solutions for the imbalance problem. In this paper, a novel one-sided feature selection known as probabilistic feature selection (PFS) was presented for imbalanced text classification. The PFS is a probabilistic method that is calculated using feature distribution. Compared to the similar methods, the PFS has more parameters. In order to evaluate the performance of the proposed method, the feature selection methods including Gini, MI, FAST and DFS were implemented. To assess the proposed method, the decision tree classifications such as C4.5 and Naive Bayes were used. The results of tests on Reuters-21875 and WebKB figures per F-measure suggested that the proposed feature selection has significantly improved the performance of the classifiers.

Keywords: Feature selection, Imbalanced class, High dimensionality, Text classification

1. Introduction

Imbalanced data is a data set, in which the number of samples in a class is much lower than the number of samples in other classes [1]. A class with large number of samples is called major class, whereas a class with small number of samples is called minor class. The imbalanced data can be seen in various areas including text classification, risk management, medical diagnosis/monitoring, biological data analysis, web categorization, and credit card fraud detection identification from satellite images.

Imbalance is divided into two types of intrinsic and extrinsic. When imbalance occurs due to the nature of data space, then it is called as intrinsic imbalance. For example, the imbalance of data in areas such as fraud detection, cancer diagnosis, earthquake prediction, and text classification is intrinsic. However, in some cases, due to the limitations such as high cost of sample collection, legal problems, or private issues, the data collection is not possible, which is called the extrinsic imbalance [2].

Classification algorithms are much more inclined toward the major class data, they might even collide with minor class data as outlier [3]. When the number of major class samples is much higher than the minor class number, and data is high dimensional, the over-fitting is more probable [4, 5].

Text data is one of the areas, where imbalance can be found. The volume of text information in the books, reports, and articles is rapidly increasing. The fast and accurate processes of this volume of information require efficient automated methods. One of the key tools in the text process is text classification. Text classification is the task of assigning a Boolean value to each pair $< \mathrm{d_j}, \mathrm{c_i} > \in \mathrm{D} \times \mathrm{C}$, where D is a domain of documents and $\mathrm{C} = \{c_1, c_2, \ldots, c_{|C|}\}$ is a set of predefined categories. A value of T assigned to $< d_j, c_i >$ indicates a decision to file d_j under c_i, while a value of F indicates a decision not to file d_j under c_i [6]. One of the problems in text classification is the high dimension of data, leading to impractical learning algorithms. The problem becomes larger when text data are imbalanced.

For example, when a figure with a variety of subjects is divided into two classes, in such a way that a class contains documents of a certain subject, and other class contains documents of all other topics, the case is called the one against all [7]. In order to solve the imbalanced data problem, numerous approaches have been introduced, including sampling methods, algorithmic methods, and feature selection methods [7, 8].

Sampling methods change the data balance by increasing the samples of minority, or decreasing the samples of majority; however, but these methods could have side effects [9]. Over-sampling methods that increase minority data lead to over-fitting [4, 10, 11], while methods that reduce majority samples lead to a loss of useful data [12]. The SMOTE method is presented to reduce imbalanced effects of data [13]. This method increases the number of minority samples by adding new samples to the existing samples. This method effectively avoids over-fitting. In [14], the MWMOTE method was presented. In this method, samples that were hardly learned by the algorithm were determined. Then, a weight was considered for the samples. The weight was determined based on Euclidean distance between this sample and nearest sample of the majority category. In [15], frequent studies and tests were performed on imbalanced text classification using the support vector machine classification. In this research, sampling methods such as under-sampling and over-sampling were examined, and the results were compared with another method of dealing with the imbalanced data including use of the cost for classification error.

Chen et al. presented a new method known as DCOM to produce new samples for minority class [10]. In this study, using the general semantic information, the minority class of a probabilistic topic model was formed for the minority, and then produced for new samples. The advantage of the proposed method was to reduce over-fitting probability.

BorajoIglesias et al. presented a novel over-sampling method for new documents called COS-HMM [11]. The COS-HMM model was trained using the figure. Then, the model was used as the new hybrid document engine. This new model showed better performance compared to random over-sampling (ROS) and SMOTE.

Liu et al., using a novel method for term weighting, develop a method to improve the performance of Naive Bayes and support vector machine for imbalanced text classification [16].

The algorithm level approaches create or modify the algorithms that exist, to take into account the significance of positive examples. These algorithms are designed in such a way to increase the performance of the minority class. Among these methods, the one-class learning, ensemble and cost sensitive learning methods can be mentioned [3].

In the one-class learning, the classifier is trained to identify a certain class. In this method, only the training data of the certain class are given to the classifier. Hence, the prediction performance of the class increases. This method always provides an optimal solution because only the data of a class is used for training. This point leads to not-well defined data boundaries [17]. One-class SVM has been successfully used in many areas such as Handwritten Digit Recognition, Information Retrieval, Spam Detection, and Medical Analysis [18].

In Ensemble methods, instead of a classifier, a set of classifiers is used. The objective of ensemble methodology is to improve the performance of single classifiers by inducing several classifiers and combining them to obtain a new classifier that outperforms other classifier. The idea is to construct several classifiers from the original data and then aggregate their predictions when unknown instances are presented. In most cases, a set of classifiers has a better performance than a classifier. The ensemble methods are divided into boosting and bagging categories. In bagging methods, various classifiers are made using bootstrapping. Then, using the mixed results obtained from the classifiers, the final conclusion is achieved. The IIVotes, UnderOverBagging, and Asymetric Bagging methods are among the bagging methods provided for the imbalanced data [3].

In Bootstrap (AdaBoost) methods, the entire dataset is used for training; however after each training, the focus is more on the data until it is properly classified. At first, the same weight is given to all the records. The weight of the samples that have been incorrectly classified will increased, while the weight of those samples that have been properly classified will be reduced. Then, another weighting is separately assigned to each classifier with respect to the overall accuracy, which will be used later in the test phase. SMOTEBoost, MSMOTEBoost, RUSBoost, and DataBoost-IM are among the methods that have been presented for imbalanced data [3].

Yang et al. used the combined sampling. In this method, the minority samples increased by SMOTE, and majority samples decreasd by under-sampling [2]. From the ensemble of the minority samples and all the majority samples, a large number of classifiers were formed, and the ensemble formed a classifier using the ensemble method; and the result was applied as the main classifier.

Cost-sensitive methods increase the performance of certain classifications by assigning different costs to classifiers instead of increasing the total performance of classification [19]. The MetaCost [19] and large cost-sensitive margin distribution machine (LCSDM) [20] are

among these methods.
Sampling methods and algorithmic methods cannot always have high performance in high-dimensional imbalanced data [21]. Recently, feature selection is considered as one of the methods to solve the imbalanced problem [8]. Several studies have shown the performance of the feature selection method in imbalanced data [22]. The results of sampling methods and feature selection and composition of both methods to improve the high-dimensional imbalanced text data classification have shown that the effect of feature selection methods is more than the sampling methods [23].
With regard to the features of text data, the typical feature selection methods cannot have the required performance for feature selection in text classification. Hence, feature selection methods of text data are presented. Feature selection methods in the text data are divided into two groups based on the approach [24] as follows:
Syntactic and semantic approach: In this approach, order of terms, meaning of terms, dependence of terms, relevance of concepts, role of terms are used in the sentence. Here, the pre-processes of stop words removal and word stemming are investigated. Moreover the out-of-text sources such as ontology might be also used In this approach[24-26].
Statistical approach: In this approach, statistical parameters such as number of repeated terms in a document, number of documents containing the term, number of repeated terms in a class, and term distribution in various documents are used. However, some certain specific features of the text such as order of terms, relevance of terms, and dependence of terms are neglected. In this approach, the pre-processes of stop words removal and word stemming are often investigated [27-30].
In this paper, a new feature selection method is presented with statistical approach for imbalanced text data classification. The various parts of this article are as follows: In the second section, the studies are examined. In the third section, the proposed method is introduced. In the fourth section, the studies are discussed and the results of tests are presented. In the fifth section, the results are analyzed. Finally, the conclusion is presented in the sixth section.

2. Related Work

Feature selection increases the speed and performance of the classifiers and reduces the over-fitting [30]. The problem of feature selection is to find a subset of the original feature of a dataset, such that an induction algorithm that is run on data containing only these fetures generates a classifier with the highest possible accuracy. Evaluating all the subsets of features for a given data becomes an NP-hard problem. Feature selection methods are divided into three classes of filter, wrapper and embedded [31].

2.1 Filter methods

Filter methods assess feature relevancies using various scoring frameworks that are independent from a machine learning algorithm. Filter techniques select top-N features attaining the highest scores. Many filter selection methods are presented so far including chi square, information gain, normalized differential measure (NDM) [27], mutual information (MI) and Gini index [32]. Due to the specific features of text data, many studies were performed to provide a method for feature selection in text data, such as distinguishing feature selector (DFS) [33], which showed better performance compared to the other methods in text data. Moreover, several studies were performed on imbalanced text data. The results showed the significane of feature selection over the learning algorithm in imbalanced text data is also

more than the learning algorithm [34].
Li and Zhu examined imbalanced text data [35] and investigated effective factors on performance of imbalanced text data classification. Effective factors were reported as follows:
Data distribution: The ratio of small to large class samples is one of the major factors in performance of imbalanced data classification. This ratio is different in various cases. According to the studies, this ratio is 1:10 in some cases, while it is 1:35 in other cases. When the ratio exceeds this limit, performance of classification is largely reduced.
Class overlapping: In case of high overlapping between classes, imbalanced distribution further reduces the performance of classifiers; however, by increasing overlapping, susceptibility of linear classifiers to data distribution is increases.
Training data volume: In case of constant ratio of small and large class samples, volume of data can affect performance of classification, so that smaller number of samples further reduces performance will be reduced further.
Subclasses: Subclasses cause complex training of classifiers.
According to these studies, when number of minority samples are very low, performance is highly reduced in the minority class. Zheng et al. proposed a framework for feature selection in imbalanced text classification [36]. In this framework, features are divided into two categories: positive feature, and negative Feature. Positive feature in a document indicates that the document belongs to a certain class, while negative feature indicates that the document does not belong to a certain class. Feature selection methods are divided into two categories of one-sided, and two-sided [36]. One-sided methods only choose positive feature, while two-sided methods select a composition of negative feature and positive feature. In [36], a multi-step approach was used for feature selection. First, positive features, and then, negative features were selected. In this research, it was assumed that L was the number of required features determined by the user. First, L1 of t feature was selected with highest f(t,c). The function f(t,c) showed the representativeness of the feature t for the category c. The maximum value of the function f (t, c) was at the time that the occurrence t indicated the category c. Then, L2 = L-L1 of a feature with the lowest f(t,$\bar{c}$) was selected. The L1/L2 ratio was an important parameter in this solution. Based on the results of this study, the feature selection significantly increased the performance of imbalanced text data classification. The results of this research properly showed the importance of feature selection for imbalanced text data classification.Furthermore, the impact of the composition of positive and negative features on imbalanced text data classification was assessed [7].
In this research, the effect of explicit and implicit composition of positive and negative features was compared. One-sided methods were used for explicit control on the ratio of number of positive and negative features. The composition rate was changed from 0.1 to 0.9. The composition rate was calculated using Equation 1:

$$SR \equiv \frac{n_+}{N} = \frac{n_+}{n_+ + n_-} \tag{1}$$

where N is the number of features, n_+ is the number of positive features, and n_- is the number of negative features. The results of this study showed that the best performance is obtained in the explicit composition of positive and negative features. However, the highest performance requires to determine the appropriate ratio of the number of positive and negative features based on number of required features.
The Performance of the proposed method was compared with the various methods having better performance in the studies. These methods are briefly described as follows. The

following definitions are used to express the calculation method.
$P(c)$: The probability that a document x belong to category c
$P(\bar{c})$: The probability that a document x does not belong to category c
$P(c|t)$: The probability that a document x belong to category c, under the condition that it contains term t.
Some other probabilities such as $P(t|c)$ are similarity defined. As mentioned in previous sections, different methods were presented for filter feature selection to choose the features in literature classification. The effectiveness of the proposed method was assessed with the Gini, DFS, FAST, and MI methods. The following subsections introduce each of these feature selection methods.

2.1.1 Gini index

Gini index is used for measuring the degree of impurity to create a decision tree. Equation 2 depicts the calculation of the Gini Index:

$$Gini(S) = 1 - \sum_{i=1}^{m} P_i^2 \quad (2)$$

where m shows the number of classes and P_i shows the probability that each case belongs to class C_i, calculated from the equation s_i/s in which s_i represents the number of cases belonging to class c_i, and s shows the total number of cases. The minimum value of this equation is 0 which occurs when all cases belong to the same class and do not exist in other classes. The value of 0 indicates the best condition. If cases are distributed equally among all classes, the maximum of this equation, i.e. 1-(1/m), would be achieved, in which c shows the number of classes. [32] proposed a modified version of the Gini Index for categorization of texts:

$$Gini(t) = \sum P(t|C_i)^2 \, P(C_i|t)^2 \quad (3)$$

where t is the word under study, and $P(C_i|t)$ is the probability that the class C_i exists on the condition that word t occurs. $P(t|C_i)$ is the probability of occurrence of the word t on the condition that the class C_i exists. The best condition is when a word exists in only one class and not in other classes in which case the equation would yield 1. Equation 3 was used in this study.

2.1.2 Distinguishing feature selector (DFS)

The DFS method is specific to text categorization. Equation 4 depicts the scoring of the word t: here, t is the word, C_i is the class i, and M is the number of classes. The results of the experiments showed that the DFS method is more effective than the Gain and chi-square in text classification.

$$DFS(t) = \sum_{i=1}^{M} \frac{P(C_i|t)}{P(\bar{t}|C_i) + P(t|\bar{C}_i) + 1} \quad (4)$$

2.1.3 Mutual information (MI)

The MI method uses the equation 5 or 6 to score t. Here, N is the total number of documents, A is the number of times t and c co-occurred, B is the number of times t occurred without C, c is the number of times c occurred without t, and D is the number of times neither t nor c occurred.

$$I(t,C) = log\frac{A*N}{(A+C)(A+B)} \quad (5)$$

$$I(t,c) = log\frac{P(c|t)}{P(t)} \quad (6)$$

2.1.4 Feature assessment by sliding thresholds (FAST)

A method called FAST is introduced for feature selection of imbalanced data. This method measures AUC by sliding the threshold for a certain classifier with single feature. Based on this study, FAST performed better than RELIEF and correlation coefficient methods [21].

2.2 Wrapper methods

In wrapper methods, features are selected based on a specific learning algorithm. In the wrapper method, various subsets of features are selected. The set of selected features are evaluated using a classification algorithm. This process is repeated until stop condition is met. In a comparison, wrapper methods find more appropriate features for considering the classifier. Moreover, filter methods are faster than wrapper methods for not repeating the classification. Two dominant methods in the wrapper method are sequential forward selection (SFS) [37] and sequential backward elimination (SBE). The SFS began from an empty set, and at each step, a feature is selected with highest association among the remaining features. In the SBE, it is began from the complete set of features, and at each step, the least significant feature is eliminiated. Instead of testing all subsets, a method is selected so that only a few of the subsets are examined to perform the algorithm. Innovative methods such as PSO [38], genetic algorithm [39], and harmony search [40] are among the most important tools in these methods. Kok et al. presented a method called SYMON for feature selection using the wrapper method for imbalanced data by the Harmony Search [40].

2.3 Embedded methods

In embedded methods, feature selection is integrated into the learning process of an algorithm. Decision tree learning algorithms can be considered among these methods, because in each step, a feature is selected for a tree. L1-SVM is also an embedded method [41]. Embedded methods are faster than wrapper methods.

3. Proposed Method

Different methods have been so far proposed for feature selection in text data classification; however, but no method is presented for feature selection of imbalanced text data. One of the problems with feature selection found in imbalanced text data classification is the composition of positive feature and negative feature. The present feature selection methods consider the same weight for these two types of features indicating two different classes, while imbalanced text data classification has an emphasis on performance of minority class [42]. For example, the optimal orthogonal centroid feature selection (OCFS) assigns larger weight to a more general class instead of a smaller class. Hence, this method has higher performance for general classes than for small classes and thus, it is not appropriate for imbalanced data [43]. For this reason, the present methods do not have the required performance in imbalanced situation. In this section, a novel method is proposed for feature

selection in imbalanced text data. This paper focused on two-class imbalanced data.
In order to examine the performance of classifiers, the true positive (TP), true negative (TN), false Positive (FP), and false negative (FN) parameters were used. Using the parameters, to review and evaluate the results of classifiers, different measures were presented [43]. Equations 7-11 show the calculation of gauges.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FN} + \text{FP}} \quad (7)$$

$$\text{ErrorRate=1-Accuracy} \quad (8)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (9)$$

$$\text{Recall} = \frac{TP}{\text{TP} + \text{FN}} \quad (10)$$

$$\text{F1\_measure} = \frac{2 * \text{presion} * \text{recall}}{\text{precision} + \text{recall}} \quad (11)$$

When data is imbalanced, Accuracy and ErrorRate are inappropriate to evaluate the classification performance since such metrics overly emphasize the majority class and neglect the rare class which is usually more important in real world applications. The measures such as Precision, Recall, and F-Measure are suitable to evaluate the performance of imbalanced data classification [1, 44].

3.1 Probabilistic feature selection (PFS)

According to the equations 9 and 10, in order to increase the measures such as Recall and Precision, the TP should be increased. The positive feature in a document indicates that the document belongs to a positive class (minority), while the negative feature indicates that the document belongs to a negative class (majority). Positive features affect the TP, FN, whereas negative features affect the TN, and FP [36]. Since the positive feature can increase TP, as a result, an optimal positive feature can increase Recall, Precision, and finally F-measure, which is a mix of Recall and Precision. Accordingly, a novel method called the probabilistic feature selection (PFS) was presented for imbalanced text data classification with a purpose to find a positive feature.
The imbalanced data is divided into negative (majority) and positive (minority) classes. In the present methods, text feature selection in some cases is not used to calculate the distinction of a term such as t. For example, the MI and DFS does not use $P(t|c)$ to calculate the feature score. In the Gini method, features with low number of repetitions receive a low score, regardless of the distribution for parameter P(t|c) and multiplication by other parameters. This is while the data distribution is important in the imbalanced data.
In the proposed method, the emphasis was on the positive class (minority). Therefore, in the proposed method, there was an attempt to use more parameters than the similar methods for positive feature measurement. In the proposed method, the following cases are considered:

- If the term t is frequently repeated in the class c, t is a good indicator for the class c; thus, this term should have a high score for the class c, which means that the probability of the term t for class c must be high. This measure can be calculated using the P(t|c) equation.

- If most of the documents containing the term t belonge to the class c, a high score should be considered for t as the indicator of the class c. This means that the probability of observation of the class c must be high in case of observing the term t. The measure can be calculated using the P(c|t) equation.

Using both parameters noted, the dependence of the class c, t is calculated using the equation 12.

$$\mathbf{score}(\mathbf{t},\mathbf{c}) = \mathbf{p}(\mathbf{t}|\mathbf{c}) + \mathbf{p}(\mathbf{c}|\mathbf{t}) \quad (12)$$

- If most of the documents that do not contain the term t are not in the class c, a high score should be considered for the term t as representative of the class c. This means that the probability of no class c should be high in case the document does not contain the term t. The measure can be calculated using the $P(\bar{c}|\bar{t})$ equation.

- If most of the documents that are not in the class c do not contain the term t, the term t should receive a high score. This means that the probability of no term t in case of not seeing the class c should be high. The measure can be calculated by the $P(\bar{t}|\bar{c})$ equation.

Using both mentioned parameters, the dependence of the class $\bar{c}$, $\bar{t}$ is calculated by the equation 13.

$$\mathbf{score}(\bar{\mathbf{t}},\bar{\mathbf{c}}) = \mathbf{p}(\bar{\mathbf{t}}|\bar{\mathbf{c}}) + \mathbf{p}(\bar{\mathbf{c}}|\bar{\mathbf{t}}) \quad (13)$$

Since in imbalanced data classification, there is a large difference in probability of majority and minority classes, each of these probabilities is divided into probability of minority and majority. According to what was said, the equation 14 can be used to calculate the representativeness of a term for class c.

$$\mathbf{PFS}(\mathbf{t}) = \mathbf{score}(\mathbf{t},\mathbf{c})/\mathbf{p}(\mathbf{c}) + \mathbf{score}(\bar{\mathbf{t}},\bar{\mathbf{c}})/\mathbf{p}(\bar{\mathbf{c}}) \quad (14)$$

Using the PFS, the feature score is determined. After scoring, the features are arranged in descending order, and the N features with the highest score are selected.

3.2 Test by sample set

In order to assess the performance of the proposed method, **Table 1** is used as sample data. This collection shows 10 documents that have been shown with Doc1, Doc2, and Doc10, respectively. The class1 documents represent the minor class, while the class2 documents represent the major class. The number of the minor class documents is 2 and the number of the major class documents is 8, and thus the imbalance rate is 0.2. The number of terms is also considered as 4, which has been shown in Term1, Term2, Term3, and Term4, respectively. The number 1 in the row i and the column j indicates the term j in the the document i, while 0 indicates absence of the term in the document.

Table 2 shows that the Term1 is a good indicator for the class1, whereas the Term2 is not a good indicator. **Table 2** shows the scores calculated by the PFS for each feature. As shown in **Table 2**, the Term1 has the highest score, while the Term2 has the lowest score. **Table 3** shows how to calculate the scores of the Term1 and Term4, in detail. After scoring, the features are arranged in descending order, and eventually the N features with the highest score are selected.

Table 1. Sample dataset

Document	Class	Term1	Term2	Term3	Term4
Doc1	Class1	1	1	1	0
Doc2	Class1	1	1	0	1
Doc3	Class2	0	1	1	1
Doc4	Class2	0	1	1	1
Doc5	Class2	0	1	1	1
Doc6	Class2	0	1	1	0
Doc7	Class2	0	1	1	0
Doc8	Class2	0	1	1	0
Doc9	Class2	0	1	1	0
Doc10	Class2	0	1	1	0

Table 2. Scores calculated for each feature

Term1	Term2	Term3	Term4
12.5	0	3.05	5.57

Table 3. Scores calculated for Term1 and Term4

	$p(c\|t)$	$p(t\|c)$	$score(t,c)$	$p(\bar{c}\|\bar{t})$	$p(\bar{t}\|\bar{c})$	$score(\bar{t},\bar{c})$	$p(c)$	$p(\bar{c})$	$PFS(t)$
Term1	1	1	2	1	1	2	$\frac{2}{10}$	$\frac{8}{10}$	12.5
Term4	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{5}{6}$	$\frac{5}{8}$	$\frac{70}{48}$	$\frac{2}{10}$	$\frac{8}{10}$	5.57

4. Experimental Setup

In order to assess the proposed method, different tests were performed. In this section, the figures and pre-processing are described. The process and measures of tests as well as results are also presented.

4.1 Dataset used

To conduct the experiment, various corpora were used. The first corpus was Reuters-21875 consisting of 52 different and imbalanced classes, where documents can exist in more than one class [45].The second corpus was WebKB which comprises 4 classes of documents and where each document belongs to one class [46]. After extracting the necessary documents from the corpora, preprocessing activities including word extraction, stop words removal, and word stemming were conducted by the Porter method.

Because the present study focused on imbalanced data, imbalanced classes with few documents were selected. For a more comprehensive investigation, one-against-all experiments were also performed in which the minor class comprised one class with few documents, while the major class was a random document of other classes. Since documents can belong to more than one class in the Reuters-21875 corpus, documents which were not in the minor class were placed in the major class. Thus, documents from other classes were selected such that their count would be 10 times the number of minor class documents or, in other word, the imbalance ratio would be 1:10. The imbalance ratio was calculated from the equation 15. **Table 4** shows characteristics of text data sets used in the experiments.

$$\boldsymbol{rate} = \frac{\boldsymbol{n_{min}}}{\boldsymbol{n_{max}}} \tag{15}$$

Table 4. Characteristics of text data sets used in the experiments.

Corpus	Minor class	Number of minor class documents	Major class	Number of major class documents	Imbalance rate	Feature count
WebKB	Project	504	Student	1641	0.3	7211
Reuters-21875	Ship	290	Acq	2383	0.12	15350
Reuters-21875	Corn	252	All	2520	0.1	14465

4.2 Designing the Experiment

The data were divided into 5 sections, in such a way that 4 sections were used for training, and one section was used for the testing. In the first stage, the first part of the data was used for testing, and 4 remaining parts are used for training, and in the second step, the second part of the data was used as testing data, and 4 remaining sections were used as training data. These steps were performed for 5 times in the same way. The mean of the performance of 5 steps was considered as the performance of a test. For more comprehensive evaluation of the imbalance rate, the number of minority class documents was increased from 10% to 100%, and performance was evaluated for each of the rates and classes. For example, for Corn-All rate, the imbalance rate has changed from 0.01 to 0.10 in the tests. The number of features selected for classification is 5, 10, 15, 20, 50, and 80. In the classification of imbalanced data, the minority class status is important [41]. Therefore, in order to evaluate the different methods, the F-Measure of minor class is used. Moreover, the criterion of Macro-F1 and Micro-F1 have been measured and reported for more comprehensive evlauation. The classifier C4.5 and Naive Bayes were used as the classifier.

4.3 Data Analysis and Evaluation

Microsoft Visual Studio 2015 was used to read the datasets and implementation of the feature selection methods. Accord.NET Machine Learning Framework was used for classifying dataset. IBM SPSS Statistics 19 was used for statistical analysis.

The experimental results are depicted in **Fig. 1** to **Fig. 6**. The x axis shows imbalance rate and the y axis shows the F-Measure of minor class. Each figure depicts the comparison of the proposed method (PFS) with other methods using either C4.5 or Naive Bayes classifier on a specific corpus. Each figure shows the performances for a certain number of features.

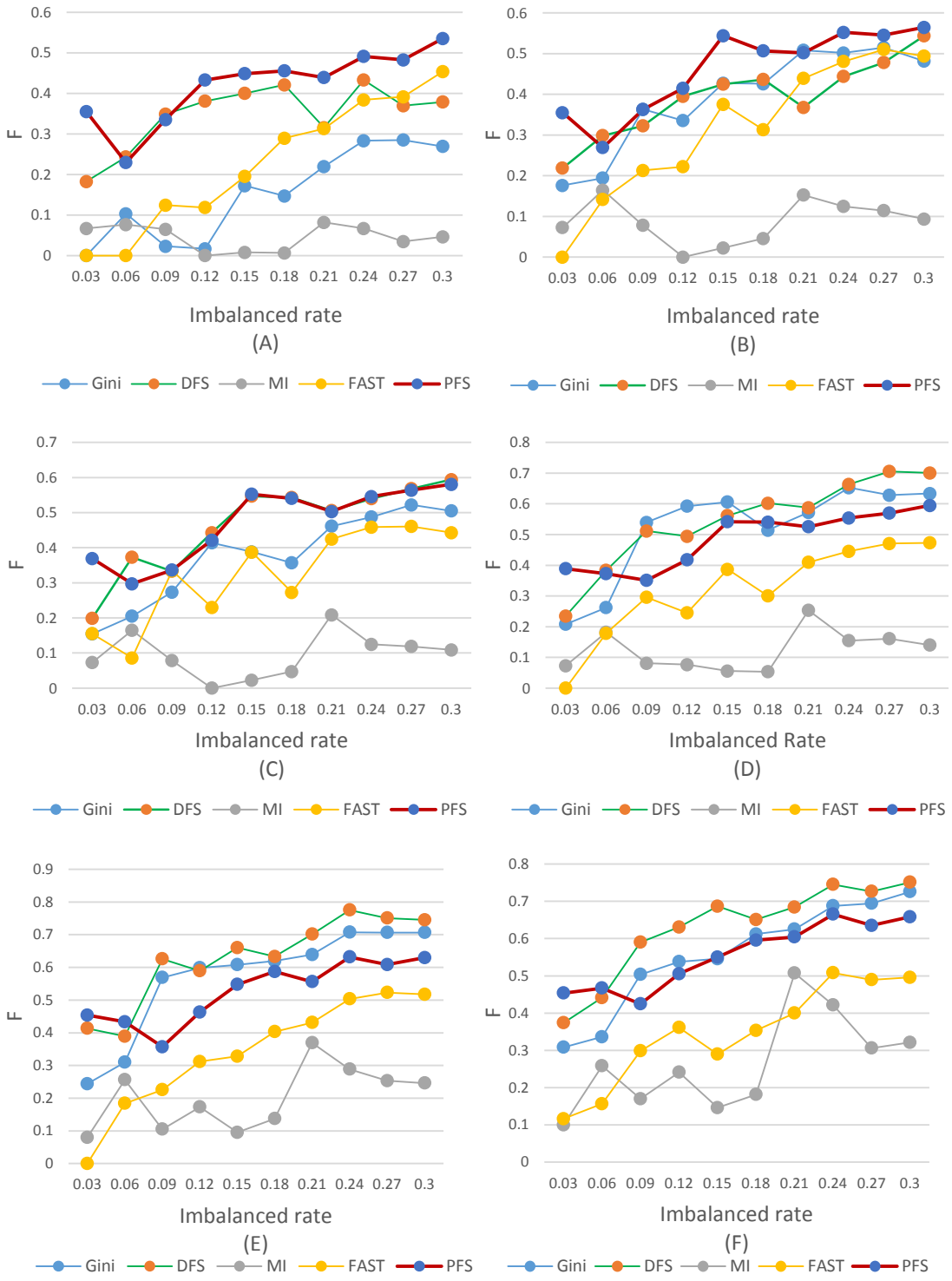

Fig. 1. Comparison of the performances C4.5 in Project and Student classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

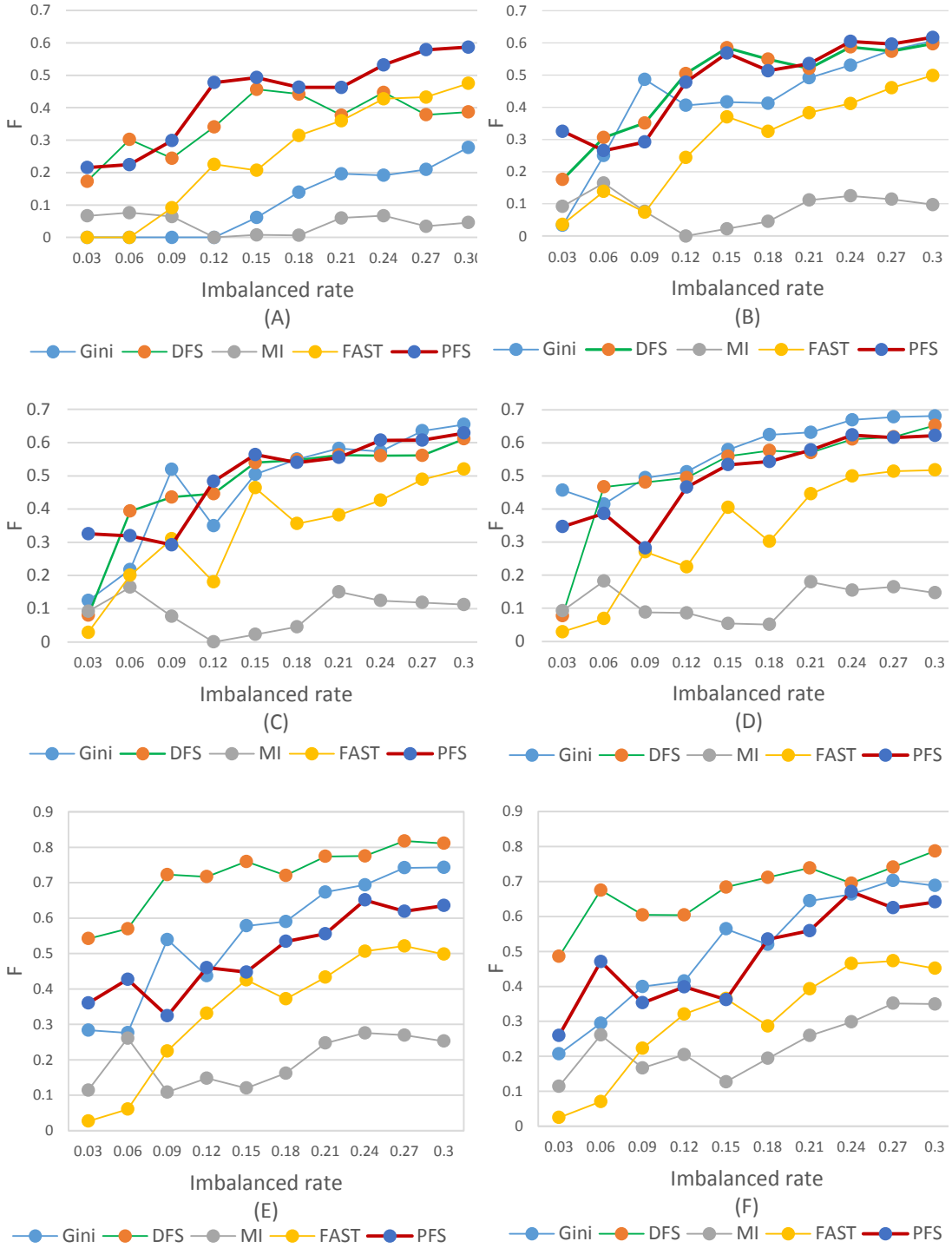

Fig. 2. Comparison of the performances of Naïve Bayes in Project and Student classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

Fig. 3. Comparison of the performances of C4.5 in Acq and Ship classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

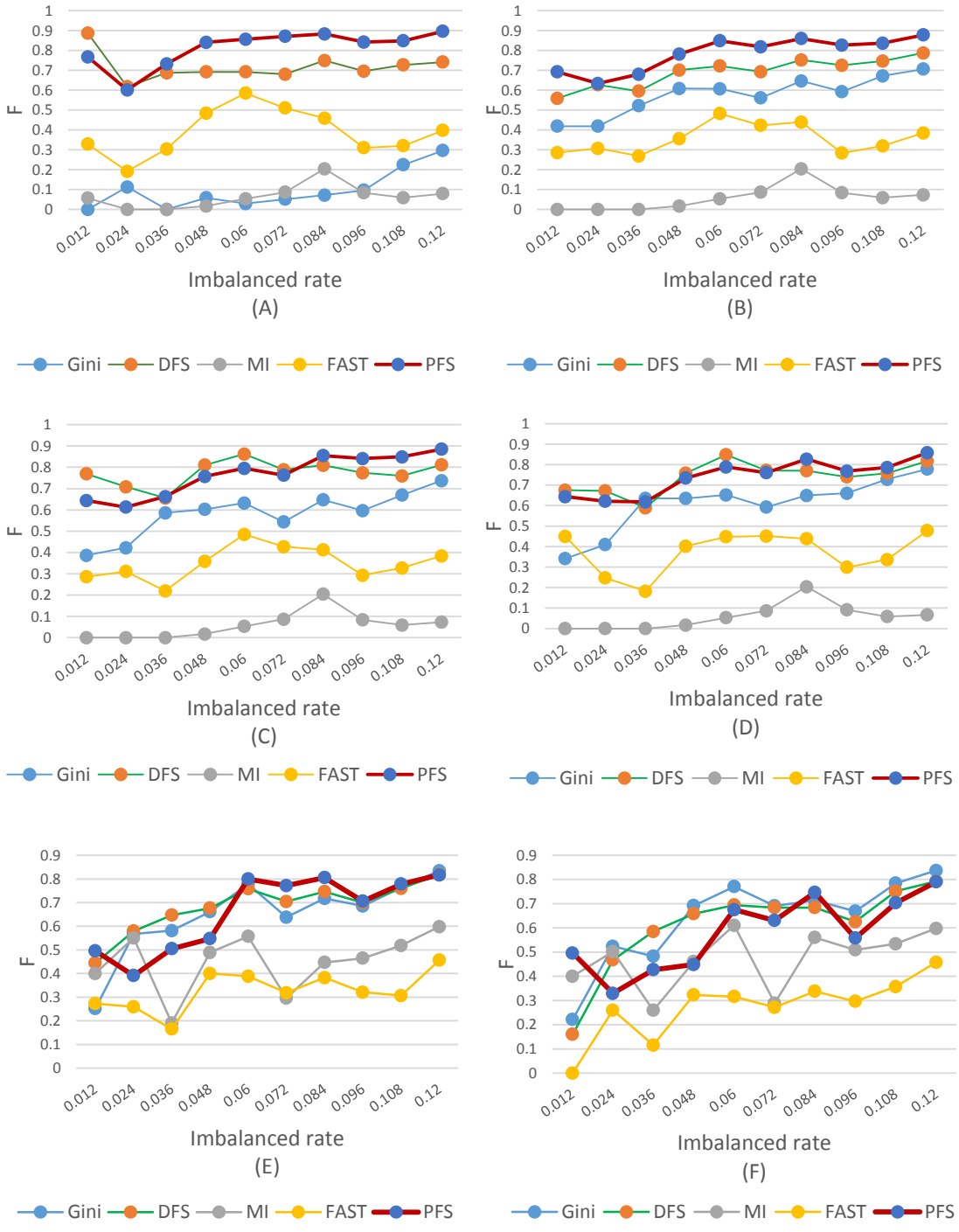

Fig. 4. Comparison of the performances of Naïve Bayes in Acq and Ship classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

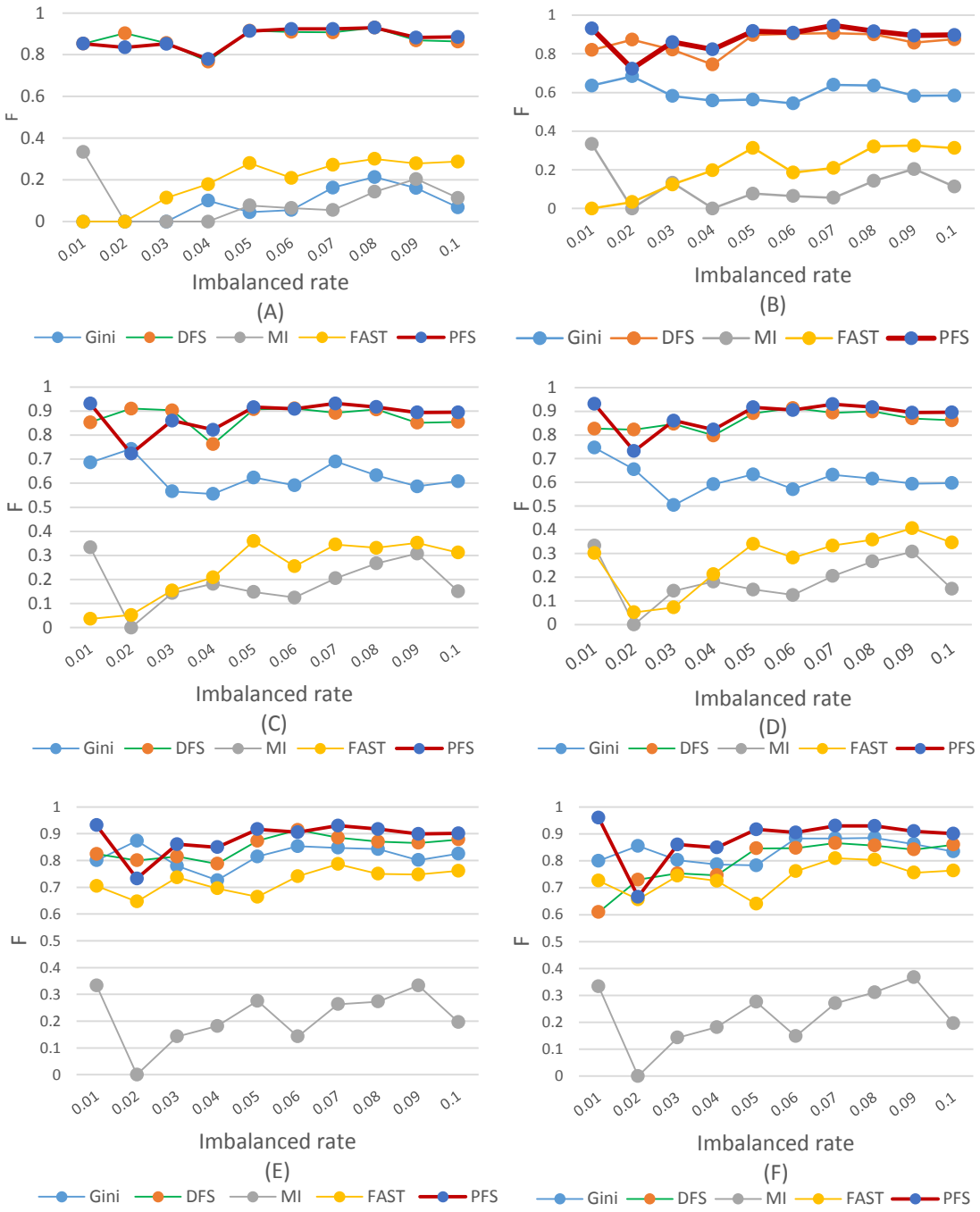

Fig. 5. Comparison of the performances of C4.5 in Corn and All classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

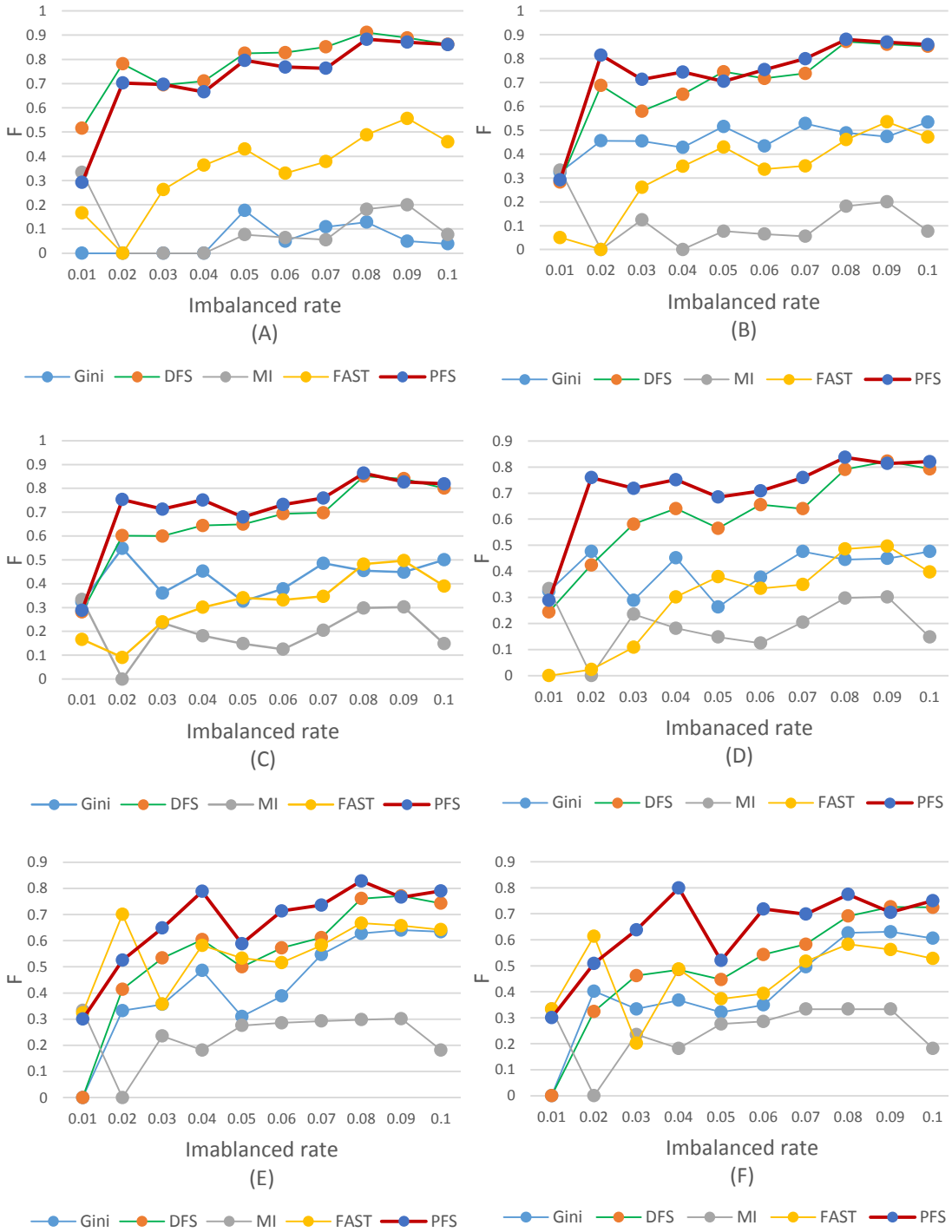

Fig. 6. Comparison of the performances of Naïve Bayes in Corn and All classes using various methods of feature selection for (a) 5 (b) 10 (c) 15 (d) 20 (e) 50 (f) 80 features

Fig. 1 and 2 indicate that PFS is the best method in the categorization of classes Project and Student belonging to the WebKB corpus using C4.5 and Naive Bayes with 5, 10, and 15 features, and is weaker than Gini and DFS with 20, 50, and 80 features. **Fig. 3 and 4** indicate that PFS is the best method in the categorization of classes Acq and Ship belonging to the Reuters-21875 corpus using C4.5 and Naive Bayes with 5, 10, 15, and 20 features, and is weaker than DFS with 50 and 80 features. **Fig. 5 and 6** indicate that PFS is the best method in the categorization of classes All and Corn using Naive Bayes and C4.5 except for one case.
Tables 5 and 6 present the comparison of the performance for various methods. Each value in these tables shows the mean of F-Measure of 10 experiments per different imbalance rates. According to these tables, the PFS method is the best method in 72 percent of methods and in other cases, it also has acceptable performance.
Table 7 presents the mean of F-Measure for all of the experiments. Results indicate that PFS is better than the other methods.

Table 5. The Performance (average value of F-Measure) of various methods using the classifier C4.5

Corpus	Class	Feature Count	Feature Selection Method				
			Gini	DFS	MI	FAST	Proposed Method
WebKB	Project-Student	5	0.15184	0.347206	0.045109	0.226889	**0.420403**
		10	0.392867	0.393111	0.086915	0.31903	**0.461748**
		15	0.376724	0.464501	0.094595	0.325191	**0.470969**
		20	0.520914	**0.544455**	0.123026	0.320478	0.485669
		50	0.570816	**0.628388**	0.200387	0.342805	0.526791
		80	0.557665	**0.628042**	0.265346	0.34684	0.555926
Reuters-21875	Acq-Ship	5	0.20017	0.72062	0.061448	0.396233	**0.85755**
		10	0.610205	0.74516	0.063044	0.411474	**0.857176**
		15	0.688268	0.852647	0.063065	0.397485	**0.85954**
		20	0.702313	0.846407	0.064693	0.401442	**0.859083**
		50	0.830412	0.832576	0.513275	0.447033	**0.872437**
		80	0.857804	0.804278	0.593403	0.462689	**0.889178**
	Corn-All	5	0.08063	0.877467	0.099047	0.192131	**0.877836**
		10	0.600869	0.860318	0.112381	0.202292	**0.882102**
		15	0.628608	0.875526	0.186159	0.240761	**0.880342**
		20	0.614345	0.86237	0.186159	0.270748	**0.880635**
		50	0.816273	0.851273	0.214202	0.72341	**0.884378**
		80	0.837651	0.795795	0.222683	0.738812	**0.882924**

Table 6. The performance (average value of F-Measure) of various methods using the classifier Naive Bayes

Corpus	Class	Feature Count	Feature Selection Method				
			Gini	DFS	MI	FAST	Proposed Method
WebKB	Project-Student	5	0.107633	0.354808	0.042917	0.253289	**0.433222**
		10	0.420841	0.474942	0.084939	0.294197	**0.479466**
		15	0.471099	0.473797	0.090775	0.33578	**0.492197**
		20	**0.57407**	0.510491	0.119926	0.327599	0.499699
		50	0.555543	**0.720785**	0.195884	0.339782	0.501289
		80	**0.509918**	0.6725	0.232658	0.30739	0.487517
Reuters-21875	Acq-Ship	5	0.094015	0.717171	0.063912	0.389309	**0.814112**
		10	0.575531	0.690923	0.057584	0.354993	**0.785658**
		15	0.582294	**0.774775**	0.057584	0.350286	0.766454
		20	0.608394	0.740102	0.057733	0.373315	**0.740782**
		50	0.647924	**0.683561**	0.451091	0.326837	0.661881
		80	**0.638236**	0.609991	0.472151	0.273494	0.580413
	Corn-All	5	0.05523	**0.786929**	0.098907	0.343707	0.730102
		10	0.463858	0.698438	0.111407	0.324375	**0.743313**
		15	0.427978	0.665936	0.197663	0.318772	**0.718751**
		20	0.402932	0.6158	0.197663	0.287782	**0.714212**
		50	0.432191	0.550948	0.238628	0.556126	**0.668438**
		80	0.459299	0.554055	0.240056	0.473223	**0.679257**

Table 7. Comparison of the performances (average value of F-Measure) of PFS and other methods

Classifier	Feature Selection Methods				
	Gini	DFS	MI	FAST	Proposed Metho
C4.5	0.55768	0.71834	0.17749	0.37587	**0.744705**
Naive Baye	0.44594	0.62755	0.16730	0.34612	**0.638709**

In order to check the significance of the differences among results of PFS and other methods, the Student T hypothesis test ($p<0.05$) was used. The following hypotheses were considered:
H0: There is no significant difference between the results of PFS and Gini methods.
H1: A significant difference exists between the results of PFS and Gini methods.
Since 5, 10, 15, 20, 50 and 80 features and 10 different rates were used in the experiments, 60 experiments have been conducted per each classifier. The same test of significance was also run for the differences between PFS and other methods. **Table 8** presents the results of significance tests.

Table 8. Results of significance tests of the differences between PFS and FAST, MI, DFS, and Gini methods

Class	Classifier		Mean	Std. Deviation	t	Sig.
Project-student	C4.5	PFS - Gini	.058446573	.138378001	3.272	**.002**
		PFS - DFS	-.014032623	.099127946	-1.097	.277
		PFS - MI	.351021379	.109194052	24.901	**.000**
		PFS - FAST	.173378847	.090958701	14.765	**.000**
	Naïve Bayes	PFS - Gini	.042380797	.165930215	1.978	.053
		PFS - DFS	-.052322085	.139683896	-2.901	**.005**
		PFS - MI	.354381863	.131099470	20.939	**.000**
		PFS - FAST	.172559058	.087586265	15.261	**.000**
Ship-Acq	C4.5	PFS - Gini	.217631985	.218082552	7.730	**.000**
		PFS - DFS	.065545782	.070003792	7.253	**.000**
		PFS - MI	.639339380	.238049066	20.804	**.000**
		PFS - FAST	.446434742	.084733029	40.811	**.000**
	Naïve Bayes	PFS - Gini	.200484518	.270019581	5.751	**.000**
		PFS - DFS	.022129672	.102529075	1.672	.100
		PFS - MI	.531541013	.286223585	14.385	**.000**
		PFS - FAST	.380177798	.104872108	28.080	**.000**
Corn-All	C4.5	PFS - Gini	.081445058	.091640927	6.884	**.000**
		PFS - DFS	.043060687	.065794140	5.070	**.000**
		PFS - MI	.666738727	.058501685	88.280	**.000**
		PFS - FAST	.287666728	.207815205	10.722	**.000**
	Naïve Bayes	PFS - Gini	.336764828	.195103300	13.370	**.000**
		PFS - DFS	.066574342	.100865795	5.113	**.000**
		PFS - MI	.520415822	.225029854	17.914	**.000**
		PFS - FAST	.321025112	.188038542	13.224	**.000**

In order to evaluate the performance of the proposed method, the Micro-F1 and Macro-F1 criteria were also calculated. **Tables 9 and 10** show the performance of the classification in terms of Micro-F1 and Macro-F. The results show that the proposed method is the best method in 84 percent of cases.

Table 9. The performance of various methods using the C4.5 in terms of Micro-F and Macro-F

class	Feature Count	Micro-F					Macro-F				
		Gini	DFS	MI	FAST	PFS	Gini	DFS	MI	FAST	PFS
Project-Student	5	0.84372	0.887808	0.86377	0.874586	**0.894435**	0.43654	**0.701717**	0.421282	0.523727	0.682262
	10	0.866881	0.883235	0.867903	0.873823	**0.903954**	0.573232	0.654307	0.51236	0.549635	**0.706499**
	15	0.858198	0.884056	0.869456	0.865038	**0.907489**	0.558936	0.638995	0.527838	0.537883	**0.715193**
	20	0.89544	0.899599	0.872659	0.859079	**0.910147**	0.661507	0.680045	0.642821	0.528919	**0.728468**
	50	0.916963	0.907739	0.885551	0.854615	**0.921237**	0.695815	0.736826	0.673529	0.540152	**0.752655**
	80	0.913172	0.901693	0.898145	0.849584	**0.93003**	0.686129	0.733685	0.68924	0.544117	**0.77119**
Acq-Ship	5	0.933133	0.971754	0.942852	0.955694	**0.986839**	0.460589	0.852287	0.405807	0.725686	**0.914304**
	10	0.958825	0.972205	0.94329	0.950975	**0.987149**	0.730965	0.842378	0.406151	0.656891	**0.912732**
	15	0.966242	0.984407	0.943519	0.948653	**0.987847**	0.784471	0.904808	0.406152	0.634719	**0.914248**
	20	0.958825	0.972205	0.94329	0.950975	**0.987149**	0.730965	0.842378	0.406151	0.656891	**0.912732**
	50	0.981721	0.982869	0.965185	0.940512	**0.990003**	0.885571	0.879589	0.795777	0.614869	**0.922021**
	80	0.983984	0.978194	0.969515	0.942881	**0.991673**	0.903287	0.855595	0.822461	0.626297	**0.931406**
Corn-All	5	0.932996	0.989846	0.952243	0.952839	**0.990585**	0.383508	**0.926424**	0.494575	0.480919	0.926281
	10	0.962044	0.988235	0.952436	0.951554	**0.990884**	0.716868	0.909559	0.56461	0.450211	**0.93273**
	15	0.964628	0.988566	0.955386	0.94435	**0.990657**	0.740323	0.918311	0.649264	0.482497	**0.931861**
	20	0.962199	0.98814	0.955386	0.946321	**0.990693**	0.730385	0.907927	0.649264	0.504606	**0.932808**
	50	0.982505	0.986895	0.958559	0.976159	**0.991066**	0.864555	0.895173	0.65607	0.81171	**0.934997**
	80	0.985478	0.982527	0.959476	0.977171	**0.991283**	0.885102	0.852198	0.65829	0.825519	**0.936029**

Table 10. The performance of various methods using Naïve Bayes in terms of Micro-F and Macro-F

class	Feature Count	Micro-F					Macro-F				
		Gini	DFS	MI	FAST	PFS	Gini	DFS	MI	FAST	PFS
Project-Student	5	0.821202	0.890502	0.863402	0.872238	**0.893084**	0.32475	**0.711715**	0.420854	0.535349	0.676505
	10	0.880516	0.882089	0.866964	0.871658	**0.901234**	0.617375	0.686378	0.512157	0.560389	**0.706633**
	15	0.896709	0.855042	0.867599	0.87327	**0.903698**	0.664285	0.62144	0.527258	0.575767	**0.715795**
	20	0.90459	0.864589	0.870147	0.874441	**0.905314**	**0.731874**	0.646925	0.656035	0.567897	0.725024
	50	0.918181	**0.937364**	0.877679	0.868623	0.906755	0.75113	**0.826673**	0.673214	0.560384	0.738872
	80	0.911699	**0.923499**	0.881971	0.861282	0.90775	0.739734	**0.790611**	0.681741	0.538341	0.746098
Acq-Ship	5	0.927981	0.972439	0.940926	0.951664	**0.98274**	0.389928	0.872969	0.406624	0.715504	**0.891749**
	10	0.961517	0.96356	0.940847	0.948467	**0.98033**	0.759709	0.779968	0.391729	0.682011	**0.878118**
	15	0.958767	0.973025	0.940847	0.948696	**0.978762**	0.734844	0.842096	0.391729	0.688323	**0.861499**
	20	0.962747	0.970205	0.940848	0.947429	**0.975667**	0.741541	0.816438	0.391767	0.650659	**0.849538**
	50	0.96799	0.96769	0.957965	0.931216	**0.972052**	0.780858	0.78244	0.777086	0.569603	**0.810899**
	80	**0.969797**	0.961987	0.959507	0.926683	0.965502	**0.795194**	0.731998	0.78363	0.489794	0.773852
Corn-All	5	0.934878	**0.984435**	0.951718	0.949845	0.980009	0.326364	0.872085	0.494621	0.555769	**0.823807**
	10	0.954199	0.977188	0.951911	0.949139	**0.980671**	0.668754	0.805648	0.564475	0.531975	**0.835086**
	15	0.950052	0.973444	0.954877	0.94623	**0.978203**	0.64139	0.779712	0.652247	0.552074	**0.824175**
	20	0.947609	0.968709	0.954877	0.9465	**0.977729**	0.630152	0.736546	0.652247	0.515291	**0.822646**
	50	0.955064	0.964213	0.956369	0.963626	**0.975711**	0.596539	0.689645	0.662494	0.729516	**0.802607**
	80	0.951947	0.964213	0.956369	0.963626	**0.975711**	0.596539	0.689645	0.662494	0.729516	**0.802607**

5. Discussion

According to **Table 5** and **Table 6**, the proposed method in C4.5 has been the best method except for 3 cases and the proposed method was the best method except for 7 cases, in Naive Bayes. In other words, the proposed method in C4.5 had been more successful than the Naive Bayes.

Figures of results of the tests show the performance of different feature selection methods and the proposed method of the imbalance text data. The results show the superiority of the proposed method than other methods. The results of the tests show that the feature selection, in case a proper method is selected, can greatly solve the problem of loss of performance for the data imbalance. The review of the figures also show that the more the imbalance is reduced, the overall performance is increased. Proposed method has considered more parameters than other methods in the feature selection, hence, it was more successful than the other methods. The closest method to the proposed method is DFS. Even in some cases, it was more successful than the proposed method, but in general, the proposed method is better.

6. Conclusion and Future Work

In the present study, a novel method was proposed for the feature selection of high-dimensional imbalanced textual data. The inadequacy of cases in the categorization of imbalanced data negatively affects performance. Thus, a new method called PFS was proposed for the categorization of imbalanced textual data using probabilities. PFS's feature selection was evaluated using the C4.5 decision tree and Naive Bayes classifiers. The Reuters-21875 and WebKB corpora were used for the evaluation of the new method. The results of PFS were compared with those of Gini, MI, DFS, and FAST using the F-Measure, Micro-F, and Macro-F. The number of features selected was 5, 10, 15, 20, 50, and 80. Evaluations were modified for different imbalance rates. Overall, the results of different experiments have shown that the

PFS method performs better than the other methods.

In the future work, we intend to study the use of text semantics as a way to improve feature selection. Also, the use of WordNet, as one of the most important sources, which shows the relationship between concepts will be used to reduce the feature.

References

[1] H. He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, pp. 1263-1284, 2009. Article (CrossRef Link)

[2] P. Yang, W. Liu, B. B. Zhou, S. Chawla, and A. Y. Zomaya, "Ensemble-based wrapper methods for feature," *springer,Advances in Knowledge Discovery and Data Mining*, vol. 7818, pp. 544-555, 2013. Article (CrossRef Link)

[3] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, pp. 463-484, 2012. Article (CrossRef Link)

[4] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: special issue on learning from imbalanced data sets," *SIGKDD Explor. Newsl.*, vol. 6, pp. 1-6, 2004. Article (CrossRef Link)

[5] J. V. Hulse, T. M. Khoshgoftaar, and A. Napolitano, "Experimental perspectives on learning from imbalanced data," in *Proc. of presented at the Proceedings of the 24th international conference on Machine learning*, Corvalis, Oregon, USA, 2007. Article (CrossRef Link)

[6] F. Sebastiani, "Machine learning in automated text categorization," *ACM computing surveys (CSUR)*, vol. 34, pp. 1-47, 2002. Article (CrossRef Link)

[7] H. Ogura, H. Amano, and M. Kondo, "Comparison of metrics for feature selection in imbalanced text classification," *Expert Systems with Applications*, vol. 38, pp. 4978-4989, 2011. Article (CrossRef Link)

[8] S. Maldonadoa, R. Weberb, and F. Famili, "Feature selection for high-dimensional class-imbalanced data sets using Support Vector Machines," *National Research Council of Canada, Ottawa, Canada Information Sciences*, vol. 286, pp. 228–246, 2014. Article (CrossRef Link)

[9] J. Pouramini and B. Minaei-Bidgoli, "A New Synthetic Oversampling Method Using Ontology and Feature Selection in Order to Improve Imbalanced Textual Data Classification in Persian Texts," *Bulletin de la Société Royale des Sciences de Liège*, vol. 85, pp. 358-375, 2016. Article (CrossRef Link)

[10]E. Chen, Y. Lin, H. Xiong, Q. Luo, and H. Ma, "Exploiting probabilistic topic models to improve text categorization under class imbalance," *Information Processing & Management*, vol. 47, pp. 202-214, 2011. Article (CrossRef Link)

[11]E. L. Iglesias, A. Seara Vieira, and L. Borrajo, "An HMM-based over-sampling technique to improve text classification," *Expert Systems with Applications*, vol. 40, pp. 7184-7192, 2013. Article (CrossRef Link)

[12]R. Barandela, R. M. Valdovinos, J .S. Sánchez, and F. J. Ferri, "The imbalanced training sample problem: Under or over sampling?," in *Proc. of Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pp.814-806, 2004. Article (CrossRef Link)

[13]N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321-357, 2002. Article (CrossRef Link)

[14]S. Barua, M. M. Islam, X. Yao, and K. Murase," MWMOTE--majority weighted minority oversampling technique for imbalanced data set learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 26, pp. 405-425, 2014. Article (CrossRef Link)

[15] A. Sun, E.-P. Lim, and Y. Liu, "On strategies for imbalanced text classification using SVM: A comparative study," *Decision Support Systems*, vol. 48, pp. 191-201, 2009. Article (CrossRef Link)

[16]Y. Liu, H. T. Loh, and A. Sun, "Imbalanced text classification: A term weighting approach," *Expert Systems with Applications*, vol. 36, pp. 690-701, 2009. Article (CrossRef Link)

[17]C. Sanchez-Hernandez, D. S. Boyd, and G. M. Foody, "One-class classification for mapping a specific land-cover class: SVDD classification of fenland," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, pp. 1061-1073, 2007. Article (CrossRef Link)

[18]S. S. Khan and M. G. Madden, "A survey of recent trends in one class classification," in *Proc. of Irish conference on Artificial Intelligence and Cognitive Science*, pp. 188-197, 2009. Article (CrossRef Link)

[19]K. M. Ting, "A comparative study of cost-sensitive boosting algorithms," in *Proc. of Proceedings of the 17th International Conference on Machine Learning*, 2000. Article (CrossRef Link)

[20]F. Cheng, J. Zhang, C. Wen, Z. Liu, and Z. Li, "Large cost-sensitive margin distribution machine for imbalanced data classification," *Neurocomputing*, vol. 224, pp. 45-57, 2017. Article (CrossRef Link)

[21]X.-w. Chen and M. Wasikowski, "FAST: a roc-based feature selection metric for small samples and imbalanced data classification problems," in *Proc. of presented at the Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, Las Vegas, Nevada, USA, 2008. Article (CrossRef Link)

[22]Y. Xu, "A Comparative Study on Feature Selection in Unbalance Text Classification," in *Proc. of presented at the Proceedings of the 2012 Fourth International Symposium on Information Science and Engineering*, 2012. Article (CrossRef Link)

[23]H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on knowledge and data engineering*, vol. 17, pp. 491-502, 2005. Article (CrossRef Link)
[24]S. Chua and N. Kulathuramaiyer, "Feature selection semantic based," *Springer Netherlands,Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*, pp. 471-476, 2008. Article (CrossRef Link)
[25]A. Khan, B. Baharudin, and K. Khan, "Efficient Feature Selection and Domain Relevance Term Weighting Method for Document Classification," *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, vol. 2, pp. 398-403, 2010. Article (CrossRef Link)
[26]W. Zong, F. Wu, L.-K. Chu, and D. Sculli, "A discriminative and semantic feature selection method for text categorization," *International Journal of Production Economics*, vol. 165, pp. 215-222, 2015. Article (CrossRef Link)
[27]A. Rehman, K. Javed, and H. A. Babri, "Feature selection based on a normalized difference measure for text classification," *Information Processing & Management*, vol. 53, pp. 473-489, 2017. Article (CrossRef Link)
[28]A. Rehman, K. Javed, H. A. Babri, and M. Saeed, "Relative discrimination criterion – A novel feature ranking method for text data," *Expert Systems with Applications*, vol. 42, pp. 3670-3681, 2015. Article (CrossRef Link)
[29]Y. Wang, Y. Liu, L. Feng, and X. Zhu, "Novel feature selection method based on harmony search for email classification," *Knowledge-Based Systems*, vol. 73, pp. 311-323, 2015. Article (CrossRef Link)
[30]R. K. Roul, A. Bhalla, and A. Srivastava, "Commonality-Rarity Score Computation: A novel Feature Selection Technique using Extended Feature Space of ELM for Text Classification," in *Proc. of presented at the Proceedings of the 8th annual meeting of the Forum on Information Retrieval Evaluation*, Kolkata, India, 2016. Article (CrossRef Link)
[31]M. Wasikowski and X.-w. Chen, "Combating the small sample class imbalance problem using feature selection," *IEEE Transactions on knowledge and data engineering*, vol. 22, pp. 1388-1400, 2010. Article (CrossRef Link)
[32]W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," *Expert Systems with Applications*, vol. 33, pp. 1-5, 2007. Article (CrossRef Link)
[33]A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems*, vol. 36, pp. 226-235, 2012. Article (CrossRef Link)
[34]G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of machine learning research*, vol. 3, pp. 1289-1305, 2003. Article (CrossRef Link)
[35]G. S. Yanling Li and Y. Zhu, "Data imbalance problem in text classification," in *Proc. of IEEE ,Third International Symposium on Information Processing*, 2010. Article (CrossRef Link)
[36]Z. Zheng and R. S. X Wu, "Feature Selection for Text Categorization on Imbalanced Data," *ACM SIGKDD Explorations Newsletter*, 2004. Article (CrossRef Link)
[37] I. Guyon, S. Gunn, M. Nikravesh, and L. A. Zadeh, *Feature extraction: foundations and applications* vol. 207: Springer, 2008. Article (CrossRef Link)
[38]Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang, "An Improved Particle Swarm Optimization for Feature Selection," *Journal of Bionic Engineering*, vol. 8, pp. 191-200, 2011. Article (CrossRef Link)
[39]A. K. Uysal and S. Gunal" ,Text classification using genetic algorithm oriented latent semantic features," *Expert Systems with Applications*, vol. 41, pp. 5938-5947, 2014. Article (CrossRef Link)
[40]A. Moayedikia, K.-L. Ong, Y. L. Boo, W. G. S. Yeoh, and R. Jensen, "Feature selection for high dimensional imbalanced class data using harmony search," *Engineering Applications of Artificial Intelligence*, vol. 57, pp. 38-49, 2017. Article (CrossRef Link)
[41] A. Y. Ng, "Feature selection, L1 vs. L2 regularization, and rotational invariance," in *Proc. of presented at the Proceedings of the twenty-first international conference on Machine learning, Banff, Alberta*, Canada, 2004. Article (CrossRef Link)

[42]M. Alibeigi, S. Hashemi, and A. Hamzeh, "DBFS: An effective Density Based Feature Selection scheme for small sample size and high dimensional imbalanced data sets," *Data & Knowledge Engineering*, vol. 81-82, pp. 67-103, 2012. Article (CrossRef Link)

[43]H. Jing, B. Wang, Y. Yang, and Y. Xu, "A General Framework of Feature Selection for Text Categorization," in *Proc. of Machine Learning and Data Mining in Pattern Recognition: 6th International Conference, MLDM 2009, Leipzig ,Germany, July 23-25, 2009. Proceedings*, P. Perner, Ed., ed Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 647-66, 2009. Article (CrossRef Link)

[44]Y. Sun, M. S. Kamel, A. K. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition*, vol. 40, pp. 3358-3378, 2007. Article (CrossRef Link)

[45]K. Bache and M. Lichman, "UCI machine learning repository," ed, 2013. Article (CrossRef Link)

[46]M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, *et al.*, "Learning to extract symbolic knowledge from the World Wide Web," in *Proc. of presented at the Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, Madison, Wisconsin, USA, 1998. Article (CrossRef Link)

Jafar Pouramini received a B.S. degree in computer engineering from the Amir Kabir University of Tehran, Iran, in 1999 and an M.Sc. degree in software engineering from the Amir Kabir University, in 2002. He is currently studing a Phd degree in College of Computer & Information Technology Engineering in University of Qom, Qom, Iran. His current research interests include Machine learning and Data mining.

Behrooz Minaei-Bidgoli received a B.S. degree in mathematics for computer science from Qom University of Tehran, Iran, an M.Sc. degree in computer engineering from the Iran University of Science and Technology and a Ph.D. degree in computer science and engineering from the Michigan State University. He is currently an assistant professor in the School of Computer Engineering in Iran University of Science and Technology. He received scholarships for the School of Computer Science and Engineering University of Michigan, U.S. in 2001.

Mahdi Esmaili received a B.S. in software engineering from the University of Isfahan, his MS in computer science in 1999 in Iran and a Ph.D. degree in computer engineering from the University of Debrecen in Hungary. He is a faculty member of Islamic Azad University Kashan branch. He has taught in the areas of database and data mining and his research interests include data base, data mining and knowledge discovery.