

# A Fast Non-Redundant Feature Selection Technique for Text Data

**SYED FAWAD HUSSAIN<sup>1,4</sup>**, (Senior Member, IEEE), **HAFIZ ZAHEER-UD-DIN BABAR<sup>1,4</sup>**, **AKHTAR KHALIL<sup>2</sup>**, (Member, IEEE), **RASHAD M. JILLANI<sup>1,4</sup>**, (Senior Member, IEEE), **MUHAMMAD HANIF<sup>1,4</sup>**, AND **KHURRAM KHURSHID<sup>3</sup>**, (Member, IEEE)

<sup>1</sup>Machine Learning and Data Science (MDS) Laboratory, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi 23640, Pakistan

<sup>2</sup>IFAHJA Private Ltd., London EC1A 9PT, U.K.

<sup>3</sup>Department of Electrical Engineering, Institute of Space Technology, Islamabad 44000, Pakistan

<sup>4</sup>Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Topi 23640, Pakistan

Corresponding author: Rashad M. Jillani (rashid@ieee.org)

**ABSTRACT** Feature selection is critical in reducing the size of data and improving classifier accuracy by selecting an optimum subset of the overall features. Traditionally, each feature is given a score against a particular category (such as using Mutual Information) and the task of feature selection comes down to choosing the top  $k$  ranked features with the best average score across all categories. However, this approach has two major drawbacks. Firstly, the maximum or average score of a feature with a class might not necessarily determine its discriminating strength among samples of other classes. Secondly, most feature selection methods only use the scores to select the discriminating features from the corpus without taking into account the redundancy of information provided by the selected features. In this paper, we propose a new feature ranking score measure called the Discriminative Mutual Information (DMI) score. This score helps to select features that distinguish samples of one category against all other categories. Moreover, Non-Redundant Feature Selection (NRFS) heuristic is also proposed that explicitly takes the problem of feature redundancy into account when selecting the features set. The performance of our approach is investigated and compared with other feature selection techniques on datasets derived from high-dimensional text corpora using multiple classification algorithms. The results show that the proposed method leads to better classification micro-F1 score as compared to other state-of-the-art methods. In particular, the proposed method shows great improvement when the number of selected features are small as well as an overall higher robustness to label noise.


**INDEX TERMS** Feature selection, mutual information, label noise, classification.

## I. INTRODUCTION

With the rapid increase in our capacity to generate data, storing and retrieving data efficiently has become increasingly difficult. An interesting and active area of current research is to develop ways in which we can arrange data, particularly non-structured data such as text. Specifically, the tasks of document clustering and classification have become one of the most powerful tools to arrange data, e.g. text corpora [1], [2]. Text categorization is the process of assigning a class or category (from a set of existing classes) to a new test document on the basis of some classification model. The classification

model itself is built by learning patterns from training data whose category labels are known a priori [3]–[5].

Feature selection is an important and integral part of text classification. Textual data usually contains a very large vocabulary, thereby, making it susceptible to the well-known problem of *curse of dimensionality* [6], [7]. Within the dictionary set, however, only a subset of features (terms/words in this case) are helpful in discriminating between categories of documents while an overwhelming majority are usually quite generic in nature, connecting phrases, emphasizing a verb or a noun, etc. This can lead to 2 possible drawbacks - firstly, when we feed a document to a classifier, the majority of the non (or less) discriminative words might overshadow the finer patterns among the discriminative words; and secondly, with increasing dimensionality, any two documents taken at

The associate editor coordinating the review of this manuscript and approving it for publication was Hengyong Yu .

random tend to have almost the same similarity scores. Moreover, most classifiers such as Neural Networks [8], SVMs [9],  $k$ -Nearest Neighbors ( $k$ NN) [10], semantic based classification [11], [12] among many others, increase in computational complexity with increasing number of features. To cope with these problems feature extraction and feature selection have been proposed in the literature, for instance in [13]–[15]. Feature extraction is a conceptual feature reduction method which creates a new feature set based on transformation and combination of the original feature set [16]. Feature selection, the relatively more commonly used technique focus of our study, refers to the selection of a subset of features from the overall feature set, usually based on some scoring function [17].

Feature selection is usually done via two methodologies – filter methods and wrapper methods. Filters select a subset of the features as a preprocessing step whereas in the case of wrapper methods, the feature selection happens with the help of the classifier prediction, using the classification as an aid for feature selection and interactively selecting features. Filter methods are usually less complex and more widely used in document classification. The most widely used measures are the Mutual Information (MI), Information Gain (IG), Term Frequency (TF), etc. (see for instance [17]–[19]), which are used to assign a score to a particular feature. After tagging a score to each feature, the features are then sorted, and we select the  $k$  features with the highest score representing the  $k$  best (most informative) features.

We identify two major drawbacks with the above-mentioned classical feature selection techniques. Firstly, features are selected incrementally based on their score rank. Therefore, while the next selected feature may be discriminative, some of the features do not necessarily provide any new information i.e., they occur in the same document set as previously selected features. Rather, these new features may provide redundant information that already exist as a result of a prior selected feature. In this regard, several authors have suggested more advanced techniques that explicitly take *non-redundant feature selection* such as the Minimum Redundancy Maximum Relevance (MRMR) algorithm by [20], Normalized Mutual Information based Feature Selection (NMIFS) algorithm by [21], etc. Almost all of these algorithms, however, suffer from a high order of computational complexity. For instance, the MRMR feature selection algorithm has a complexity of  $O(vk^2)$  where  $v$  is the total number of features and  $k$  is the number of selected feature set.

Secondly, a score-based feature selection has to be estimated from the available data. This may become a problem when our category labels in the training data do not reflect the true nature of the data. This can be considered as *noise*. There could be different types of noise – for instance, the labels in the training data do not reflect the actual categories (label noise), the presence of outliers that may distort the information score of a feature, etc. When working with real world datasets, it is quite common that some of the training labels

might be mislabeled [22]. In general, the information about which samples might be incorrectly labeled as well as their percentage in the total data set are both missing. It has been shown that only a few mislabeled samples can cause a large percentage of the most discriminative features to remain unidentified [22]. In such cases, traditional techniques like the ones mentioned above may not generate a good representative subset of the features. Therefore, any feature selection technique should be resistant to label noise and base its selection on strong statistical correlations in addition to training labels. In our approach, we exploit the statistical relations existing within the data to select a feature subset. Hence, a mislabeled sample is less likely to hurt the feature selection process.

Our primary objective in this study is to reduce the drawbacks of traditional feature selection techniques without drastically increasing the time complexity. Therefore, we propose modifications in existing techniques such as Mutual Information or Information Gain to improve both the scoring criteria and the selection mechanism. These modifications help in ranking features based on their discriminative power (between categories) as well as in assuring some diversity in the set of features that will eventually be used by the classifiers. Therefore, we improve on the performance of traditional techniques but without employing sophisticated selection criteria, which may result in less redundancy among features, but is usually quite expensive (computationally).

## A. CONTRIBUTION

To minimize the first drawback, we examine the mutual information-based feature selection using both the average and maximum MI scores between features and categories. It is observed that when a given feature has high MI score with more than one class, it may not be highly discriminative during the classification process. Therefore, we modify the scoring criteria and propose a new scoring mechanism that ranks features that highly discriminate only one category from the rest. Therefore, our features are ranked on this new scoring rather than simple MI or IG scores.

The second drawback arises due to the selection technique. Features with high MI or IG scores may not necessarily aid the classification if they occur in the same set of samples. Therefore, it is important that the selection process chooses diverse yet highly discriminative features. However, the computational time in seeking non-redundant features in most advanced techniques is prohibitive. This is mostly because, at each iteration, they compare the new feature to be selected with each of the already selected feature(s). We, therefore, propose a simple yet intuitive way to keep track of the documents that have already been represented by the selected set of features. New features are now selected based on both their discriminative rank score and their coverage of unrepresented documents in the dataset until all documents are well represented.

Additionally, we also study the effect of label noise which is also a serious concern in both classification and feature selection. We compare and evaluate different feature selection

methods for their robustness and resistance to label noise. With random noise distribution in the training labels, feature scores might be misleading, which is known to adversely affect the accuracy of classification [23]. Because the whole process of feature selection depends upon labels in the training corpora, it is hard to develop an analytical model that can wholly and accurately eliminate the adverse effect of label noise. To this end, we perform several tests to provide empirical evidence that our proposed method normalizes a feature's information score both relative to its occurrence in the documents of a specific category as well as its information score in other categories.

In summary, the contributions in this paper are as follows:

- As a first contribution, we propose a new feature ranking score for document classification that helps in identifying highly discriminative features.
- Secondly, we propose a heuristic that selects non-redundant features in a computationally efficient manner.
- Finally, we explore and evaluate the behavior of several traditional and non-redundant feature selection techniques on data with label noise by varying the number of features and the percentage of noisy labels.

Our experiments show that the proposed method selects a better representative feature subset that is evident by an improved classification accuracy. Moreover, the proposed method is computationally less expensive than other advanced feature selection techniques while being equally or more tolerant to label noise on the tested datasets.

The rest of this paper is organized as follows. In Section II, we review several state-of-the-art feature scoring and selection techniques that also serves as a background of our proposed method. Section III presents the proposed fast and robust method for selecting features. Section IV details our experimental setup and gives a comparative analysis of the proposed approach with existing methods. Finally, we conclude our work in Section V.

## II. RELATED WORK

Many feature selection scores have been proposed in the literature both to decrease the computational complexity of classification algorithms and as a way to increase the classification accuracies. Earlier studies, such as [17], [24], concentrated on the classical approaches of using feature scores to select a set of ranked feature. During the last decade, several algorithms have been proposed that consider additional information, such as the information content being provided by each successive feature set. Perhaps the most popular among these approaches is by Peng *et al.* [20] called the minimum redundancy and maximum relevance (mRMR) feature-selection method, used to select a subset of genes from gene expression datasets. Another algorithm was proposed by [21] that uses mutual information to select the next feature to be added to an existing feature set, from the original dataset. More recently, authors in [25] proposed a technique by maximizing the global information gain (MGIG) for feature selection in

which the information content of the selected feature subset is considered as a whole rather than that of individual features making up the subset. These algorithms are based on metrics like Information gain, mutual information, and statistical tests such as the Chi-square for term scoring. We briefly describe some of these metrics and their related algorithms below.

### A. NOTATIONS

In the paper, we shall use  $\mathbf{D} \in \mathfrak{R}^{m \times v}$  to denote the data matrix, where  $m$  is the number of instances (here documents) and  $v$  is the number of features (here words). Let  $F = \{f_1, \dots, f_v\}$  denote the  $v$  features vectors where  $f_i \in \mathfrak{R}^m$ , and  $d_1 \dots d_m$  denote the  $m$  instances where  $d_i \in \mathfrak{R}^v$ . We can denote the matrix  $\mathbf{D} = \{f_1, \dots, f_v\}$  as a set of feature (column) vectors, or  $\mathbf{D} = \{d_1 \dots d_m\}$  as a set of document (row) vectors. Let  $S$  denotes the set of selected features  $f_1, \dots, f_k$ , and  $k$  be the number of selected features.  $C$  is a set of all the categories,  $c_i, i \in 1 \dots n$  where  $n$  is the number of categories. We will use  $f_i$  when simply referring to the  $i$ th feature and  $f_i$  when emphasizing its vector properties.

### B. SCORE BASED FEATURE SELECTION

There are numerous score-based feature selection methods that have been used in the literature. *Chi-square* test is a widely used statistical technique [26] which measures the lack of independence between a feature  $f$  and a class  $c$ . This particular technique is known to be better for multi-class problems but might ignore words with a lower count. *Document Frequency* (DF) is one of the simplest methods for feature selection where features are selected on the basis of their occurrence in documents and a predefined threshold is set so that only features having more occurrences will be selected. *Probability Ratio* is another technique which can be used to produce a ranking of features based on how frequently they are correctly identified as being relevant. The ratio is defined in terms of sample true positive rate over sample false negative rate. Another method is the *Filter Based Method* (FBM) that uses  $t$ -statistics to filter features. It assumes a positive value to one class and negative values to all other classes and uses a statistical approach to define the significance of each feature [27]. A detailed discussion and analysis of the first 3 techniques can be found in [17], [27], [28].

Statistical methods, such as those mentioned above, are simple and readily applicable but do not usually result in a high accuracy value for most classifiers [17]. Among the conventional approaches, these methods are considered to be the simplest feature selection methods with manageable complexity of  $O(v)$  where  $v$  is the number of terms. Information theory based scores are preferred for text categorization as they have been shown to give better classification accuracy [17], [19], [21], [29]. We discuss some of the popular scores below.

#### 1) MUTUAL INFORMATION (MI)

Mutual information (strictly, the point-wise mutual information) [17] separately measures the term's (word's) total

strength integrated with each category of documents. Mutual information shows how much a term is related to a particular category. Given  $f_i$ , a term from the set of terms, and  $c_j$ , a category from the set of categories, we define

$$(f_i; c_j) = \log \frac{P_r(f_i, c_j)}{P_r(f_i) P_r(c_j)} \quad (1)$$

where  $P_r$  denotes the probability. Naturally,  $I(f, c)$  results as zero if  $f$  and  $c$  are independent. The values from Equation 1 show the relationship of a given term with one given category. To compute the overall term's score, we use one of the following two ways:

$$I_{avg}(f_i; C) = \sum_{j=1}^n P_r(c_j) I(f_i, c_j) \quad (2)$$

$$I_{max}(f_i; C) = \max_{j=1}^n \{I(f_i, c_j)\} \quad (3)$$

We will use the same mutual information criterion given in Equation 1 in our proposed algorithm (Section III).

### 2) INFORMATION GAIN (IG)

Information gain [17] measures how much information we get for category prediction, given the presence or absence of a particular term. The average information gain from a term  $f$  is defined as

$$G(f) = - \sum_{i=1}^n P_r(c_i) \log P_r(c_i) + P_r(f) \times \sum_{i=1}^n P_r(c_i | f) \log P_r(c_i | f) + P_r(\bar{f}) \sum_{i=1}^n P_r(c_i | \bar{f}) \log P_r(c_i | \bar{f}) \quad (4)$$

where  $\bar{f}$  denotes the absence of  $f$ . Information gain basically shows variation in entropy with respect to the presence or absence of the particular term,  $f$ . IG is also called as expected MI.

Although score-based feature selection methods are easy to use and also have less complexity, they do not consider the existing feature set when selecting a new feature to be added to the set of selected features. Rather, these algorithms only base their decision on the individual score of the term. As mentioned previously, this may result in new features having redundant information getting selected. Thus, some documents may be represented by many features while other documents might get less or even no features in the reduced feature space after feature selection.

### C. ADVANCED FEATURE SELECTION METHODS

To avoid the drawbacks mentioned in the last section, several algorithms have been proposed that consider the existing selected features for making an informed decision on the next feature to be selected. These algorithms have been called as *non-redundant feature selection* methods or *higher order feature selection* methods in the literature [20], [25]. These algorithms consider multiple factors before choosing a feature,

such as redundancy problem, balancing features according to classes, etc. as an integrated part of feature selection.

#### 1) MUTUAL INFORMATION BASED FEATURE SELECTION (MIFS)

The authors in [30] used this approach, based on mutual information by formulating a criteria for feature selection, except for the first feature. The first feature is selected based on maximum mutual information among all the features. Additional features will be selected if they maximize the following

$$I(C; f_i) - \beta \sum_{f_s \in S} I(f_s; f_i) \quad (5)$$

where  $\beta$  is a user defined parameter that regulates the relative importance of redundancy among candidate feature and the set of already selected features. The first part of the equation calculates relevance of a particular feature vector to be added with respect to the category, and the second part estimates the redundancy among the  $i$ th feature  $f_i$  with respect to already selected feature set,  $S$ .

#### 2) MINIMUM REDUNDANCY AND MAXIMUM RELEVANCE (mRMR)

In [20], authors presented an approach based on minimizing the redundancy while retaining the relevance of the feature set. This approach is also based on the mutual information criteria. The key ideas of this algorithm are maximizing the relevance of the feature while minimizing the redundancy with existing feature set. The authors calculate the weightage of a feature  $f_i$  not only with respect to the classes,  $I(C, f_i)$ , but also considering the mutual information with the selected features,  $I(f_i, f_j)$ . MRMR selects that feature which maximizes following criteria

$$\max_{f_i \in F - S_{k-1}} [I(f_i; C) - \frac{1}{k-1} \sum_{f_j \in S_{k-1}} I(f_i; f_j)] \quad (6)$$

Here  $F$  is the set of actual features and  $S_{k-1}$  is the set of selected features, prior to selecting the  $k^{\text{th}}$  feature. A drawback in the MIFS algorithm was that with the increase in number of selected feature set, the right-hand term (cumulative sum) increases in magnitude which could overshadow the value of the left hand term (MI score). As a result, the discriminatory power of the later features is less relevant to its non-redundancy score. In mRMR, this drawback has somewhat been resolved by normalizing the total sum with the size of  $S$  instead of  $\beta$  factor in MIFS.

#### 3) NORMALIZED MUTUAL INFORMATION FEATURE SELECTION (NMIFS)

In [21], authors extended the framework used by MIFS to use the average normalized MI as a measure of redundancy between the  $i$ th feature and the already selected feature set. They have defined normalized MI between  $f_i$  and  $f_s$ ,  $NI(f_i, f_s)$  as the MI normalized by the minimum entropy of both features. Selection of the first feature takes place in a similar

fashion as in MIFS. Afterwards, all other features are selected based on maximizing the following

$$\max_{f_i \in F-S} [I(f_i; c) - \frac{1}{|S|} \sum_{f_j \in S} MI(f_i; f_j)] \quad (7)$$

The technique works similarly to mRMR except that it uses normalized MI among the selected feature set and feature vector to be added instead of non-normalized MI as has been used in mRMR.

#### 4) MAXIMIZING GLOBAL INFORMATION GAIN (MGIG)

All the higher order or non-redundant feature selection algorithms discussed above have a polynomial time complexity in  $O(vk^2)$ . Recently, Shang *et al.* proposed a novel feature selection technique known as the MGIG [25]. They have tried to reduce the computational complexity for selecting features. To this end, they proposed to use a new metric, called GIG or Global Information Gain, as a method to compute the information score of a feature. Feature selection is now based on how much a feature contributes toward achieving global information gain. Selecting a particular feature depends on how much it maximizes

$$\arg \max(p_r(\tilde{f}_{S_{k+1}})H(p_r(C | \tilde{f}_{S_{k+1}})) - p_r(f_{k+1})H(p_r(C | f_{k+1}))) \quad (8)$$

where  $S$  is the subset of selected features,  $k$  is the number of features already selected,  $H$  is the entropy measure, and  $\tilde{f}$  is a virtual term that results from combining all currently selected features into a single term. Thus, instead of comparing a new feature with each individual feature in the selected set, we need to perform only one comparison.

#### 5) BALANCED INFORMATION GAIN BASED FEATURE SELECTION (BIGFS)

Information gain is computed based on probabilities and does not consider the frequencies of occurrences of words in documents. The authors in [31] argue that not only does the frequency of occurrence matters, but also that positive correlation is a major factor in text classification while negative correlation has a secondary role. This means that the occurrence of a term vs. the non-occurrence of a term should be treated in a different manner. In their approach, they propose using a frequency-based occurrence (instead of binary) and also introduce the notion of a *balance factor* to control the conditional entropy in the computation of IG.

#### 6) COMPOSITION OF FEATURE RELEVANCY (CFS)

More recently, the authors in [55] proposed a modification to the feature score and selection strategy of mRMR technique. The relevancy score is computed by computing the joint conditional information of the candidate feature with a category given the set of selected features, and the feature redundancy is given by the joint information of the candidate feature, categories, and selected features. Results have shown that

CFS has been able to outperform state-of-the-art techniques on a variety of datasets.

Similarly, several other methods have been proposed for feature selection, for instance [18], [32], [33]. A complete survey of feature selection methods is beyond the scope of this paper. However, a summary of commonly employed feature selection algorithms appears in Table 1. We restrict ourselves to those methods using mutual information or related scores, and omit other approaches such as [56], [57] which are based on search using evolutionary strategies. More recent methods have also been proposed, particularly those using information theory and suitable for high-dimensional data such as in [58]–[62].

An excellent survey of mutual information based feature selection methods, along with an empirical study of their drawbacks can be found in [28] and [58]. For a comprehensive survey on feature selection for text classification, we refer interested readers to [63], while a survey on feature selection using optimization schemes is presented in [64].

All of the algorithms discussed above try to deal with the problem of redundancy during feature selection, while retaining their discriminatory values (information contents). A general drawback of these approaches, however, is the computation complexity since each feature warrants a comparison with all selected features to minimize redundancy. This usually runs into polynomial time and might be a major hurdle in using these higher order feature selection algorithms in the real world, where the number of features could be quite large.

### III. PROPOSED NON-REDUNDANT FEATURE SELECTION (NRFS)

In this section, we detail our proposed feature selection technique called Non-Redundant Feature Selection (NRFS). The proposed algorithm is based on the mutual information (Section II, subsection B.1) criteria. However, we improve the method in two aspects – the modified score to better judge the discriminatory power of a feature, and a heuristic based feature selection method to select non-redundant features. We show why using the sum (or taking the maximum) of individual *feature-to-class* MI scores might not be a good idea which is the motivation behind our first improvement. Secondly, we show a potential drawback in the traditional mutual information selection method and motivate the need of selecting the features heuristically, instead of on the basis of ranked score alone. The proposed algorithm is a combination of these two strategies which we discuss below.

#### A. DISCRIMINATIVE MUTUAL INFORMATION (DMI) SCORE FOR FEATURE SELECTION

Mutual Information based feature selection, as proposed in [17], first calculates the MI score of each feature with respect to each class. The global MI score assigned to a particular feature is based on either Equation (2) or (3) discussed in the previous section. These features are then sorted based on their MI scores (in descending order) and the algorithm

**TABLE 1.** Summary of important feature selection techniques.

Selection Method	Year	Key idea/advantage/application
Mutual Inf. Max. [24]	1992	Incremental search for mutual information maximization
MIFS [30]	1994	Mutual information-based feature selection
Yang et al. [17]	1997	Information Metrics. Point-wise mutual information
Yang and Moody [45]	2000	Joint mutual information between candidate feature and each of selected feature
Kwak and Choi [46]	2002	An improvement to MIFS for when distribution is more uniform
FBM [27]	2002	Uses t-statistics to filter features by assuming positive and negative values between classes.
WAPMI [44]	2005	Weighted average point-wise mutual information
MRMR [20]	2005	Based on minimum redundancy and maximum relevance, similar to MIFS. Redundancy term is less important as more features are selected.
mMIFS-U [47]	2007	Conditional mutual information between class and feature condition upon already selected features utilized as classifier independent criteria
CMIM [48]	2012	Eliminate redundant features using cumulate conditional mutual information minimization criterion
Herman et al. [49]	2013	Takes into account both the class-dependent and class-independent correlation among features.
MGIG [25]	2013	Globally maximizes information. Faster than MRMR and similar methods.
BIGFS [31]	2014	Uses a balance factor to control the conditional entropy in the computation of IG.
Improved IG [50]	2015	Features are selected by categories and are merged by an optimized method. Tailored for usage with SVM classifier.
cMFDR [51]	2016	cMFDR computes one threshold per category to assure that every category contributes with a different number of features.
FMIFS-MD [53]	2016	Fuzzy MI based non-dominated solution using feature-class fuzzy MI and feature-feature fuzzy MI scores.
Peng & Fan [52]	2017	By optimizing lower bound of conditional mutual information
SFR [54]	2018	Uses subspace feature clustering to identify feature clusters
CFS [55]	2018	Similar to MRMR and uses composition of feature relevancy
Wang et al. [59]	2019	Uses rough set theory based relative neighborhood self-information on both lower and upper approximations.
PRFS [60]	2020	Proportional Rough Feature Selection based on rough set for regional distinction
Liu et al. [61]	2020	Independent feature space search using relative doc-term frequency difference for class correlation and redundancy
Hossny et al. [62]	2020	Uses text mining specifics e.g., word count, word forms such as n-gram, skip-gram, etc.
Gao et al. [65]	2020	min-redundancy and max-dependency (MRMD) using relevancy with a class given selected features

selects the top  $k$  features from the dataset. This, however, may not be a good strategy to find highly discriminative features as explained below.

Consider the point-wise mutual information of three features –  $f_1$ ,  $f_2$ , and  $f_3$  with four categories  $c_1$ - $c_4$  as given in Table 2. The values in the table show the respective MI scores of each feature with respect to a category label. Using

**TABLE 2.** MI score calculation via 2 different objective functions.

Categories	$f_1$	$f_2$	$f_3$
$c_1$	1.50	1.50	1.2
$c_2$	1.42	0.89	0.2
$c_3$	1.80	0.72	0.1
$c_4$	0.8	0.55	0.2

the traditional MI criteria (Equation 3 in Section II, subsection B) with the *max* function, for instance, would select feature  $f_1$  as the topmost feature since it has maximum score of 1.80 with category  $c_3$ . This, however, ignores the fact that feature  $f_1$  also has a high MI score with other categories. Therefore, even though feature  $f_1$  has the highest global MI score, it has less discriminative capability to categorize a sample in the test data since its occurrence does not signify a high probability of the test sample belonging to a particular category. Similarly, using the average of MI scores (Equation 2 in Section II, subsection B) results in features  $f_1$ ,  $f_2$ , and  $f_3$  have a global MI score of 1.38, 0.915, and 0.425 respectively. As before, feature  $f_1$  will get selected due to its highest average MI score. In reality, however, even though feature  $f_3$  has a smaller MI score of just 1.2 with category  $c_1$  but its score with the rest of the classes are relatively lower. This implies that it may have more discriminative power as compared to  $f_1$  or  $f_2$ . Hence, feature  $f_3$  is a better candidate to help predict the category of a test sample as compared to  $f_1$  or  $f_2$ .

In order to avoid selecting features with less discriminative powers, we propose to modify the criterion for determining the most discriminative feature. In our proposed method, the final global MI score is computed as a ratio such that features that have a higher variation between their maximum MI score with a particular category and their scores with the rest of the categories are preferred over features with a lower variation. Thus, it is more important that a feature's MI score with one category stands out over its scores with the other categories rather than only having a high MI score. Let  $n$  be the number of categories,  $c_i$  represent the  $i^{\text{th}}$  category of documents,  $N_i$  be the number of documents belonging to category  $c_i$ ,  $f$  be the feature,  $\bar{f}$  denote not occurrence of  $f$ , and  $c_j$  be the category. Then, the Discriminative Mutual Information (DMI) is computed using the equation below (9), as shown at the bottom of the next page:

The first part of the equation helps to deal with imbalanced data where number of documents in the categories are unequal, while the second part takes the ratio of the MI score of a feature with a category with respect to its scores with other categories. Applying the DMI score given by Equation (9) on the example given in Table 2 shows the effect of this modification. The new DMI criteria now normalizes the scores in relation to other categories. Assuming equal documents (say 250) in each category, the new score for features  $f_1$ ,  $f_2$ , and  $f_3$  would be 0.16, 0.23, and 0.8 respectively. Therefore, new ranking of the features would be  $f_3 > f_2 > f_1$  and  $f_3$  would be the top ranked feature due to its more discriminating power.

### B. HEURISTIC BASED FEATURE SELECTION

A second and more significant improvement in our proposed feature selection algorithm is the criteria for selecting features by avoiding redundancy as a result of incremental feature selection. The idea here is that selection of a given feature should depend, in addition to its DMI score about the categories, on the *new information* it brings into the subset of already selected features. Here, by *information* we mean that the feature occurs in document(s) not previously represented by the existing subset of selected features. In this way, the new feature adds to the information of the category and can be used by a classification algorithm. A feature with a slightly lower DMI score but one that helps in discriminating more documents should be preferred over a feature with a higher score that has redundant information. We could use the graph analogy to better understand the idea behind our proposed method. Consider documents as nodes in a graph and a feature occurring in two documents as an edge (connection) between these two nodes. Then, our goal is to select features that will provide a full graph connectivity.

The methodology of feature selection is as follows. Recall that  $F$  represents a set of feature vectors  $\{f_1 \dots f_v\}$  corresponding to  $D$ . We maintain a vector  $R_{m \times 1}$  whose length is equal to the number of documents  $m$  in the corpus. Let  $R_j \in \{0,1\}$   $j=1 \dots m$  be the bit representation of the  $j^{\text{th}}$  element of the vector indicating which is set to 1 if the  $j^{\text{th}}$  document is represented by the set of selected features or 0 if it is not. Let  $\{C_i\}$  be the set of rows indices of rows belonging to category  $c_i$  and  $\cup$  represent the bitwise OR operator on vectors, we define  $\delta_f$  as an indicator function to represent whether adding the feature  $f$  to the set of existing features will result in an addition of information or not. Mathematically,

$$\delta_{f,i} = \begin{cases} 1 & \text{if } \left[ \sum_j^m \hat{R}_{j \in (C_i)} \right] \\ 0 & \text{otherwise} \end{cases} \text{ where } \hat{R} = (R \cup f) - R \tag{10}$$

When deciding on a new feature, say  $f_i$ , we consider both the information content of  $f_i$  (using the Discriminative Mutual Information score as described in equation (9) and the non-redundant content of the feature. The final score is, therefore, a combination of Equation (9) and Equation (10) and is given by

$$\delta_{\text{Score}}(f) = [DMI(f) * \delta_{f,i}] \tag{11}$$

### C. OVERALL ALGORITHM

In our method, a feature (f) is only selected if adding it to R will result in new documents getting representation.

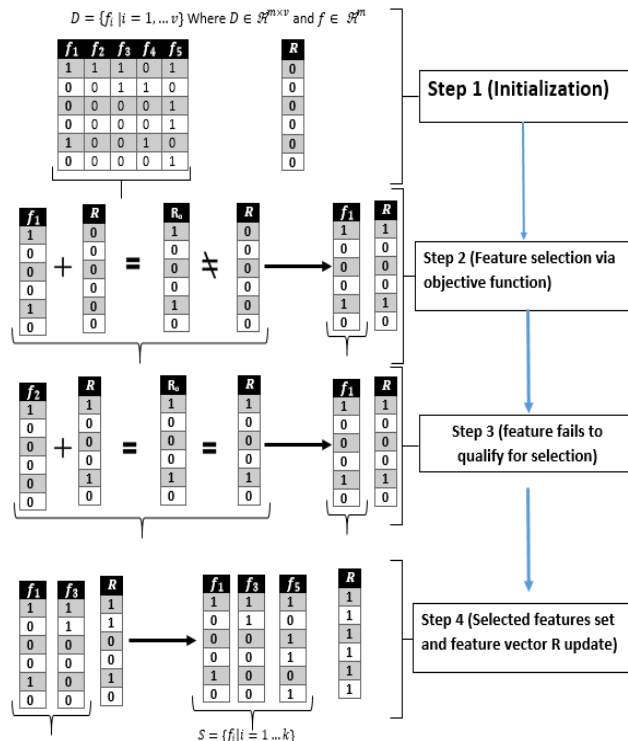


FIGURE 1. An example of non-redundant intelligent feature selection.

This guarantees that the new feature being selected is non-redundant. After selecting this new feature, the feature vector  $R$  is updated to incorporate the new information. Hence, any new feature  $f_i$  needs to be compared to only one feature vector,  $R$ , rather than all existing features  $f_1 \dots f_{i-1}$ , which significantly speeds up the selection process. Moreover, the next feature to be selected is based on its DMI score and its value of  $\delta_{f,i}$  thus avoiding re-computing the scores of the remaining features. A comparative analysis with other state-of-the-art algorithms is given in Table 3 while the algorithm is given in Table 4.

### D. A WORKED EXAMPLE

To better explain the motivation behind our proposed approach and to help understand the working of the algorithm, we use the following example given in Figure 1. The matrix  $D$  represents a typical document-term matrix and the vector  $R$  represents a bit representation of the documents as explained previously.

- 1) The dataset  $D$  has 6 documents and 5 features. The vector  $R$  of size  $6 \times 1$  is initialized with all by default 0 values, since none of feature vector has been selected

$$DMI(f) = \frac{N_i}{\sum_{j=1, j \neq i}^n N_j} \times \frac{P(f \wedge c_i) \log_2 \left( \frac{P(f \wedge c_i)}{P(f) \times P(c_i)} \right) + P(\bar{f} \wedge c_i) \log_2 \left( \frac{P(\bar{f} \wedge c_i)}{P(\bar{f}) \times P(c_i)} \right)}{\sum_{j=1, j \neq i}^n P(f \wedge c_j) \log_2 \left( \frac{P(f \wedge c_j)}{P(f) \times P(c_j)} \right) + P(\bar{f} \wedge c_j) \log_2 \left( \frac{P(\bar{f} \wedge c_j)}{P(\bar{f}) \times P(c_j)} \right)} \tag{9}$$

TABLE 3. Summary of comparison with other state-of-the-art algorithms.

Objectives/Algorithms	TS	MI	IG	MGIG	BIGFS	MRMR	CFS	NRFS
Representation of all documents	✗	✗	✗	✗	✗	✗	✗	✓
Non-Redundant	✗	✗	✗	✓	✓	✓	✓	✓
Informative	✗	✓	✓	✓	✓	✓	✓	✓
Simple and Fast	✓	✓	✓	✗	✗	✗	✗	✓

TABLE 4. Pseudocode for the NRFS algorithm.

<b>Algorithm:</b> <i>Non Redundant Feature Selection</i>	
<b>Input:</b> Data matrix $\mathbf{D}$ , number of desired features $k$ .	
<b>Output:</b> A reduced data matrix $\mathbf{D}'$ with dimensions of $m \times k$	
<b>Initialization:</b> $S = \{\emptyset\}$ is the set of selected features. $\bar{S} = \{\emptyset\}$ is the set of non-selected features. $\mathbf{R} = \mathbf{0}$ is a vector of size $m \times 1$ , whose values are set to 0.	
1. Calculate the Discriminative Mutual Information score $DMI(f_i; C)$ for each feature vector $f_i \in F$ .	
2. Sort all the features in $F$ based on their MI scores (in descending order)	
3. Select the first feature $f_1$ from sorted set of features and $Set S \leftarrow \{f_1\}$	
<b>repeat:</b>	
4. increment $i$	
5. set $F \leftarrow F - f_i$	
6. Update $\mathbf{R}$ as follows: $if score(f_i) > 0$ $S \leftarrow S \cup \{f_i\}$ $\mathbf{R} \leftarrow \mathbf{R} + f_i$ $else$ $\bar{S} \leftarrow \bar{S} \cup \{f_i\}$	
<b>Until</b> $ S  = k$ OR $F = \{\emptyset\}$	
7. If NRFS fails to verify $ S  = k$ then select the remaining top $k -  S $ features from $\bar{S}$	
<b>End</b>	

yet. At the initial stage,  $S$  is also initialized to  $\{\varphi\}$ . Similarly,  $\bar{S}$  is also initialized to  $\{\varphi\}$ .

- 2) As a first step, the first feature vector,  $f_1$  is selected. This feature occurs in both document  $d_1$  and  $d_5$ . Hence, we add to the selected features set  $S$ , and set bits 1 and 5 in  $\mathbf{R}$  to 1.
- 3) We now consider the next feature with highest MI,  $f_2$  for possible selection. Since  $f_2$  only occurs in document  $d_1$ , the resultant vector  $f_2 + \mathbf{R}$  does not change the previous value of  $\mathbf{R}$ . Thus, this feature will be rejected with no change in  $\mathbf{R}$ . The feature is added to the set of rejected features,  $\bar{S}$ , for possible selection at a later stage.
- 4) We repeat the above steps and select features  $f_1, f_3$  and  $f_5$ , each of which occur in at least one new document not represented previously in  $\mathbf{R}$ .

**E. COMPUTATIONAL COMPLEXITY**

In this section, we take a step-by-step look at the computational complexity of NRFS. In the first step, it will have same computational complexity as mutual information i.e.  $O(mv)$  where  $v$  is total number of feature vectors in data corpus with  $m$  documents. For the sorting step, we may use either of heapsort or mergesort as both have computational complexity of  $O(v \log v)$ . Note that both these steps are also

performed in all the classical mutual information based feature selection algorithm as well as the higher order feature selection algorithms. In the feature selection procedure, a single feature selection takes  $O(m)$  because it just needs to have just one comparison with the vector  $\mathbf{R}$ , instead of all the currently selected features  $S$  as in [21]. A single selection of feature vector takes  $O(m)$  and for selecting  $k$  feature, it will take  $O(km)$ . Hence, the overall computational complexity is computed as  $O(v)$  for mutual information computations,  $O(v \log v)$  for sorting algorithm, and  $O(km)$  for the comparison with  $\mathbf{R}$  for each of the  $k$  features. Therefore, the computational complexity of the MI based classical feature selection algorithm is given by,

$$O(v \log v) \tag{12}$$

since the  $v \gg m$  in most cases. As this is used by all the algorithms, we ignore this bit and only consider the overhead cost. The additional term in our algorithm is the feature comparison, thus giving it a complexity of

$$O(km) \tag{13}$$

which is negligible in the overall complexity since  $k \ll v$  and, therefore, the complexity is comparable to that of the classical feature selection algorithms (described in Section II,



subsection B). As can be seen, the computational complexity is better than either of mRMR, NMIFS, or CFS all of which have a complexity term of  $O(mvk^2)$ . For MGIG, the computational complexity is given by  $O(mvk)$  since all features are compared with the virtual feature at each iteration.

#### IV. EVALUATION AND RESULTS

This section evaluates the performance of the NRFS algorithm and compares it with other state-of-the-art methods. We provide an empirical analysis of several popular feature selection techniques employed in text categorization. It is usually not straight forward to compare which feature selection technique is better since different feature subsets might have different inherent characteristics and may be suited for some particular tasks. As this paper is focused on supervised feature selection for document classification, we evaluate the effect of feature selection on text categorization results. In particular, we generate several data sets with the same characteristics (e.g. number of documents, number of features, etc.), using different feature selection methods and classify them using popular supervised classification methods. It is then possible to evaluate the effect of feature selection based on the final classification results, since all other parameters are kept constant.

The evaluation is performed on several key aspects. Firstly, we explore the behavior of the NRFS technique on several text categorization datasets. We evaluate the task of text categorization on 6 datasets, using 5 different number of selected features and employing 4 categorization techniques (Section IV, subsection D.1). This will help us analyze the behavior of the technique with increasing the number of selected feature set as well as with the complexity of the problem (in terms of number of document categories). Secondly, we further compute the results of categorization at the two extreme ends (with low and high number of features) of the feature sets and compare the performance with other state-of-the-art algorithms and on different categorization algorithms (Section IV, subsection D.2).

To study the robustness of the feature selection techniques, in particular its resistance to label noise, we design a new set of experiments where a noise factor is introduced into the training data. In particular, we are interested in the performance of the feature selection techniques in the presence of label noise, which is a popular form of noise in supervised learning. We steadily increase the percentage of label noise in the training data and compute the micro-F1 of the test results. As previously, we explore by varying the number of selected features using various feature selection techniques and on different datasets (Section IV, subsection E.2). Finally, we again test other state-of-the-art algorithms on the same task at both low and high number of selected features (Section IV, subsection D.3).

In the following, we first describe the benchmark datasets used for our evaluation (Section IV, subsection A), the various document categorization techniques (Section IV, subsection B) used, and the performance evaluation criteria (Section IV,

subsection C) before analyzing NRFS and other feature selection algorithms and evaluation the categorization results (Section IV, subsection D and Section IV, subsection E).

#### A. BENCHMARK DATASET

In our experimental study, we employed three popular benchmark text corpora, namely the *20Newsgroup*<sup>1</sup>, *Reuters30*<sup>3</sup>, and *TDT*<sup>3</sup>. To perform a diverse and fair evaluation, we create different datasets from these corpora with a mixture of attributes such as balanced and imbalance datasets, small and large number of categories, similar and different topics, etc. The datasets were pre-processed to remove stop words, perform stemming, and converted to lower case.

The *20Newsgroup* corpus is also a well-known corpus that has nearly 20,000 document samples, divided evenly into 20 classes corresponding to news items collected over a period of time. Interesting subsets from this corpus have been used in the literature, for instance, by fixing the number of documents and varying the number of categories. We choose popular subsets used in the literature [10], [11], [34], [35], namely M2, M5 and M10 corresponding to 2, 5 and 10 categories of the *20Newsgroup* corpus. The overall vocabulary, after pre-processing is 111,868 words. The number of documents is fixed at 500 and M2, M5, and M10 have 250, 100, and 50 documents per class. For each dataset, we created 10 different subsets to reduce any bias. For instance, for the M2 dataset, we randomly select 250 documents each from 1000 documents related to Middle East politics and world politics respectively. This is repeated 10 times to generate 10 different M2 subsets.

The *TDT* corpus is one of the newly designed corpus which comprises text data from different news channels and newsgroups. It has a vocabulary of approximately 57,000 words. We create a subset from this corpus having 20 classes and 75 documents each and used the same pre-processing as before. All of M2, M5, M10, and TDT are balanced datasets in that they have equal documents per class but with varying number of classes. The M10 and TDT having higher number of classes present a challenging task since the number of documents per class decrease.

For a detailed comparison, we also use imbalanced datasets in our experiments. The *Reuters-21578* corpus is one of the most acknowledged dataset and contains documents collected from the Reuters newswire in 1987. In the original dataset, some of the documents belong to different categories from a total of 135 categories. We discarded documents belonging to different categories and were left with only 66 categories with total vocabulary of 35247. The dataset is highly imbalanced since different categories contain vastly different number of documents. The largest category has 3923 documents while the smallest 23 categories have less than 10 documents. We used a popular variant of this corpus by selecting the top 30 categories referred to as the *Reuters30* dataset.

<sup>1</sup><http://qwone.com/~jason/20Newsgroups/>

<sup>3</sup><https://sites.google.com/site/fawadsyed/datasets>

The IMB20 is a second imbalanced dataset created from the 20Newsgroup corpus. As before, we use the same steps to generate this highly skewed dataset. An arithmetic sequence of 21 entries is generated between 0 – 1 and assigned to each category (except 0). These entries are then used as sampling probabilities to randomly select documents for each category which form the new dataset. The datasets use a 50–50 split for training and testing. In order to avoid any bias, the experiment is repeated 10 times, each time selecting the training and test documents at random and we report the mean F1 score from these 10 runs.

## B. CLASSIFIERS

To avoid any bias introduced by a classifier towards a particular feature selection method, we employ 4 popularly used classifiers including Naïve Bayesian Classifier, Support Vector Machine,  $k$ -Nearest Neighbours and Multinomial logistic regression-based classification. These approaches were selected since they have been used in the literature, are readily available and also do not use sophisticated approaches that could bias our results. For example, link based classifiers such as [11], [36] that could bias towards the non-redundancy characteristics by exploiting word-word associations, and [37], [38] that incorporate feature selection into the categorization process.

*Naïve Bayesian* classifier [39] uses posterior probability through Bayes formula to classify a document sample. It is a simple but effective classifier which assumes that the presence or absence of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. Documents are classified by computing their probability of being in any one of the given categories and assigned to the category having maximum probability.

*Support Vector Machines* (SVM) [9] is a popular and powerful classifier which performs binary classification (data having 2 categories) by finding a maximal marginal hyper plane in the data, based on the training data. Since SVM basically works for binary class data, we have implemented multi class SVM by using a well-known *one-vs-rest* approach [40]. We use the linear kernel and keep the default parameters (such as soft margin) in Matlab.

*k-Nearest Neighbors* ( $k$ NN) is one of the most used classifiers that is based on finding distances between vectors. It classifies documents on the basis of their nearest  $k$ -neighbors<sup>2</sup> in the training data. The category of the test vector is then decided using a majority voting. We used the Cosine similarity measure as it has been shown to give good results for document classification [11], primarily since it normalizes the vector (document) length.

*Multinomial Logistic Regression* (MLR) [41] is a classification technique for multiclass prediction. The categorical dependent data is modeled as a combination of the independent variables (attributes) and the class of the test data is

<sup>2</sup>Not to be confused with the parameter  $k$  used for the number of selected features earlier.

determined using a probabilistic approach. MLR is a multi-class extension of the logistic regression model. We use the implementation available in Matlab ® using default values.

## C. PERFORMANCE MEASURES

To evaluate our algorithm in comparison to other methodologies, we employ to popularly used F1 measure. The F1 measure uses both the *precision* and the *recall*. *Precision* is the ratio of the correct results (true positives) to all results predicted as correct (true positives + false positives). *Recall* is the ratio of the correct results (true positives) to the actual number of true results. The F1 measure is then computed as the Harmonic mean of *precision* and *recall*. The F1 measure is a fairly good measure of performance since it measures how many documents were correctly classified where a value of 0 indicates a bad classification while a value of 1 indicates a perfect classification. Note that for uni-labeled data (one document can belong to only one class), the *precision* and *recall* values are equal and the F1 measure is, therefore, equivalent to the accuracy measure.

The F1 score is further computed in 2 ways. The MacroAveragedF1 score simply treats each category as having equal number of documents (irrespective of their actual sizes). It is basically the degree of agreement of the observed labels and the actual labels or the percentage of number of correctly classified data to total classified data. The MacroAveragedF1, or shortened to *MacroF1*, is given by

$$MacroF1 = \frac{\sum_{i=1}^n \frac{2 \times p_i \times r_i}{p_i + r_i}}{n} \quad (14)$$

where  $p_i$  and  $r_i$  are the *precision* and *recall* values of category  $i$  respectively, and  $n$  is the number of categories. In the case where the category is imbalanced, a second measure known as the MicroAveragedF1 (or simply *MicroF1*) is preferred since it takes the difference sizes of the categories into account. The *MicroF1* score is given by

$$MicroF1 = \frac{2 \times \frac{\sum_{i=1}^n tp_i}{\sum_{i=1}^n (tp_i + fp_i)} \times \frac{\sum_{i=1}^n tp_i}{\sum_{i=1}^n (tp_i + fn_i)}}{\frac{\sum_{i=1}^n tp_i}{\sum_{i=1}^n (tp_i + fp_i)} + \frac{\sum_{i=1}^n tp_i}{\sum_{i=1}^n (tp_i + fn_i)}} \quad (15)$$

where  $tp_i$ ,  $fp_i$ , and  $fn_i$  are the true positive, false positive, and false negative values respectively for the category  $i$ .

## D. RESULTS AND ANALYSIS

We use the datasets generated in Section IV, subsection A as our benchmark dataset and perform several experiments to analyze the behavior of our algorithm. We design our experiments to study the behavior of the proposed feature selection technique by comparing the micro and macro F1 scores of the different classifiers on different datasets under varying settings. The details of our results and subsequent discussion is given in the following sub-sections.

### 1) EFFECT OF NUMBER OF FEATURES ON NRFS

In this experiment, we consider the case where the number of categories is fixed for each dataset while the number of

TABLE 5. Average Micro-F1 score for 10 experiments on varying number of features using NRFS.

			500	1000	1500	2000	2500
Balanced Dataset	M2	kNN	<b>0.94</b> ±0.01	<b>0.94</b> ±0.01	<b>0.94</b> ±0.02	<b>0.94</b> ±0.02	<b>0.95</b> ±0.02
		SVM	0.92±0.01	0.92±0.01	0.93±0.02	0.92±0.02	0.94±0.03
		NB	0.89±0.02	0.89±0.02	0.89±0.02	0.89±0.02	0.89±0.03
		MLR	0.60±0.02	0.59±0.02	0.59±0.02	0.60±0.02	0.60±0.01
	M5	kNN	0.82±0.02	<b>0.85</b> ±0.02	<b>0.88</b> ±0.02	<b>0.90</b> ±0.02	<b>0.91</b> ±0.02
		SVM	0.85±0.03	0.84±0.03	0.84±0.02	0.84±0.02	0.84±0.02
		NB	<b>0.89</b> ±0.04	<b>0.85</b> ±0.04	0.78±0.04	0.75±0.04	0.74±0.04
		MLR	0.30±0.03	0.29±0.03	0.260±0.02	0.26±0.02	0.24±0.02
	M10	kNN	<b>0.69</b> ±0.02	<b>0.69</b> ±0.04	<b>0.68</b> ±0.04	<b>0.68</b> ±0.04	0.64±0.03
		SVM	0.67±0.02	0.67±0.03	0.66±0.02	0.64±0.03	<b>0.70</b> ±0.04
		NB	<b>0.69</b> ±0.03	<b>0.69</b> ±0.04	0.65±0.04	0.63±0.04	0.62±0.07
		MLR	0.19±0.02	0.19±0.02	0.17±0.02	0.16±0.02	0.16±0.02
TDT	kNN	<b>0.94</b> ±0.01	<b>0.94</b> ±0.01	<b>0.94</b> ±0.01	<b>0.94</b> ±0.01	<b>0.93</b> ±0.01	
	SVM	0.93±0.01	0.93±0.01	0.93±0.02	0.92±0.02	0.88±0.03	
	NB	0.92±0.02	0.92±0.01	0.92±0.01	0.92±0.03	0.91±0.04	
	MLR	0.23±0.03	0.23±0.02	0.22±0.03	0.21±0.02	0.19±0.02	
Imbalanced Dataset	IMB20	kNN	<b>0.72</b> ±0.02	<b>0.72</b> ±0.01	0.72±0.02	0.71±0.01	0.71±0.02
		SVM	0.71±0.01	0.71±0.01	<b>0.73</b> ±0.01	<b>0.72</b> ±0.01	<b>0.72</b> ±0.01
		NB	0.65±0.02	0.65±0.02	0.67±0.02	0.68±0.02	0.68±0.02
		MLR	0.21±0.03	0.21±0.01	0.22±0.03	0.25±0.02	0.25±0.03
	Reuters30	kNN	<b>0.71</b> ±0.02	<b>0.71</b> ±0.02	<b>0.71</b> ±0.02	<b>0.67</b> ±0.02	<b>0.66</b> ±0.02
		SVM	0.66±0.02	0.66±0.04	0.55±0.04	0.55±0.04	0.53±0.06
		NB	0.26±0.04	0.26±0.03	0.28±0.03	0.29±0.03	0.32±0.03
		MLR	0.13±0.03	0.12±0.03	0.13±0.02	0.12±0.02	0.11±0.02

features vary from 500 to 2500 at intervals of 500. Therefore, for the same number of documents, we use NRFS to select increasing number of features. This series of experiments is aimed at exploring the behavior of the NRFS technique with increasing number of features and also to test which of the classification techniques performs better. We used the NRFS to select features by selecting the datasets at random and applied the classification algorithm to get the Micro-F1 value (macro-F1 scores are given in Appendix for imbalanced dataset). Each experiment is repeated 10 and we report the average and standard deviation values. The results are shown in Table 5.

From the table, we observe the following. Firstly, as can be expected, the datasets with the lower number of categories result in higher micro-F1 values (M2 over M5 and M10). Secondly, as given by the average results in the last 3 rows, using kNN and SVM gave better results than NB irrespective of the number of features chosen, with kNN giving slightly better averages over SVM. One important observation, however, is that just using 500 features resulted in an equal of better micro-F1 value as compared to higher number of features in most of the cases. This result may at first seems a little strange given that usually micro-F1 values either remains stable or increase (initially) with higher number of features being selected. One possible explanation of this is that the proposed NRFS algorithm selects the most informative (using

the modified MI score) and non-redundant features, thus providing coverage to all training documents with a small number of features.

Another interesting observation is the results obtained using MLR which gives quite low micro-F1 values as compared to other algorithms. This may be due to the nature of the data since textual data is usually very sparse and using a generative model might not necessarily be a good thing. Naïve Bayesian treats features independently, therefore, two features that are highly correlated to a given label but also to each other will both get a high weight. In the case of MLR, however, since features are considered to be dependent, the weights are adjusted accordingly. This may not fare well since a test data might only contain a subset of the correlated features (due to the sparse nature of the data). Besides, and as discussed in [42], NB may outperform LR (and consequently MLR) when the training data is small. Our results are consistent with these observations.

As discussed in Section III, subsection B, if the number of selected features is less than the desired number of features and all training documents have been represented, the rest of the features are selected using the criteria of MI. Therefore, selecting fewer but *non-redundant* and *highly discriminatory* features yield higher micro-F1 values, which may actually decrease as increasing the number of features will lead to features with lower MI score being selected. We shall further

discuss this aspect when we compare the results obtained using NRFS and other techniques in the next section.

## 2) COMPARISON OF NRFS WITH OTHER TECHNIQUES

In this experiment, we analyze the performance of NRFS to those of other state-of-the-art feature selection techniques commonly employed for text categorization. We compare the performance of NRFS with 6 other feature selection techniques namely MI, IG, mRMR, MGIG, CFS, and BIGFS (see Section II for description of techniques). We consider the two extreme cases of 500 and 2500 features respectively and plot the micro-F1 values on all the 5 datasets and using all 4 classifiers. This results in 240 experimental values and is shown in Figure 2.

For more complex datasets such as the M10, Reuters30 and TDT (having 10, 30 and 20 classes respectively), we can see that NRFS performs quite well. It significantly outperforms the classical methods (MI and IG) and also realize better micro-F1 values than more sophisticated algorithms (MGIG, mRMR and CFS) and 500 features, as shown in Figure 2. For the other classifiers, NRFS is comparable or better than all other algorithms, with the exception of NB where mRMR yields slightly better results. Using 500 features, NRFS gives an average micro-F1 (over all 5 datasets) of 0.82, 0.81 and 0.72 for  $k$ NN, SVM and NB as compared to the second best technique (mRMR) having 0.77, 0.81 and 0.74 and much better than the baseline MI technique that yields 0.53, 0.51 and 0.42, respectively.

For the complex dataset (having many categories) and when using a small number of feature set, NRFS ensures both that the features are highly discriminatory in nature (due to the modified MI criteria) and diverse (covering new, previously not represented documents). The former helps the classifier into building a strong classification model, while the later ensures that documents in the test data are better represented. Since textual data is sparse, a diversified feature set helps in unseen documents having more non-zero elements in their document vectors. Another important observation here is seen when the number of features are increased to 2500. As the number of features selected increases, NRFS becomes more and closer to using MI since, one all documents in the training datasets have been represented by some features, the rest of the features are chosen on the MI scores. It is also interesting to note that with increasing features, the more advanced but also computationally expensive algorithms, mRMR and MGIG, report better micro-F1 values for higher number of features. This can be attributed to the fact that while NRFS simply adds the features based on MI scores, these techniques continue to search for new features that bring least redundancy as compared to the existing feature set.

## E. ROBUSTNESS OF THE FEATURE SELECTION ALGORITHM

As discussed in Section I, it is often difficult to obtain completely reliable labels both in terms of their label accuracies and outliers. In order to analyze the robustness of the

different selection techniques, we introduce label noise into our datasets and re-run the experiments to compare the robustness of the different techniques. In our case, we consider noise with no known distribution or prior knowledge i.e. neither the percentage of noise, nor the category-wise presence is known. Our aim is to evaluate the robustness of the proposed as well as other state-of-the-art approaches to label noise. In the following sections, we describe the process of introducing label noise and test the robustness of the feature selection algorithms to label noise.

### 1) NOISE INSERTION IN TARGET LABELS

To test the effect of noise on our proposed NRFS method, we create a modified version of the datasets used previously, ensuring we use the same training/test split. More precisely, we introduce random label noise into the dataset by keeping the original document term matrix as it is, but randomly changing the category labels of some of the document with a certain probability,  $p$ . Noise is inserted using a normal distribution and is not category specific, i.e. a 5% noise means 5% of the total training documents have been intentionally mislabeled using existing labels. For instance, a document belonging to category 3 in the M5 dataset (with 5 classes) has been mislabeled as 1, 2, 4 or 5. We create 4 different noise levels by varying the amount of inaccurate labels creating datasets with 5, 10, 15 and 20 percent label noise. To generate different a noise level, we vary the value of  $p$  as given below.

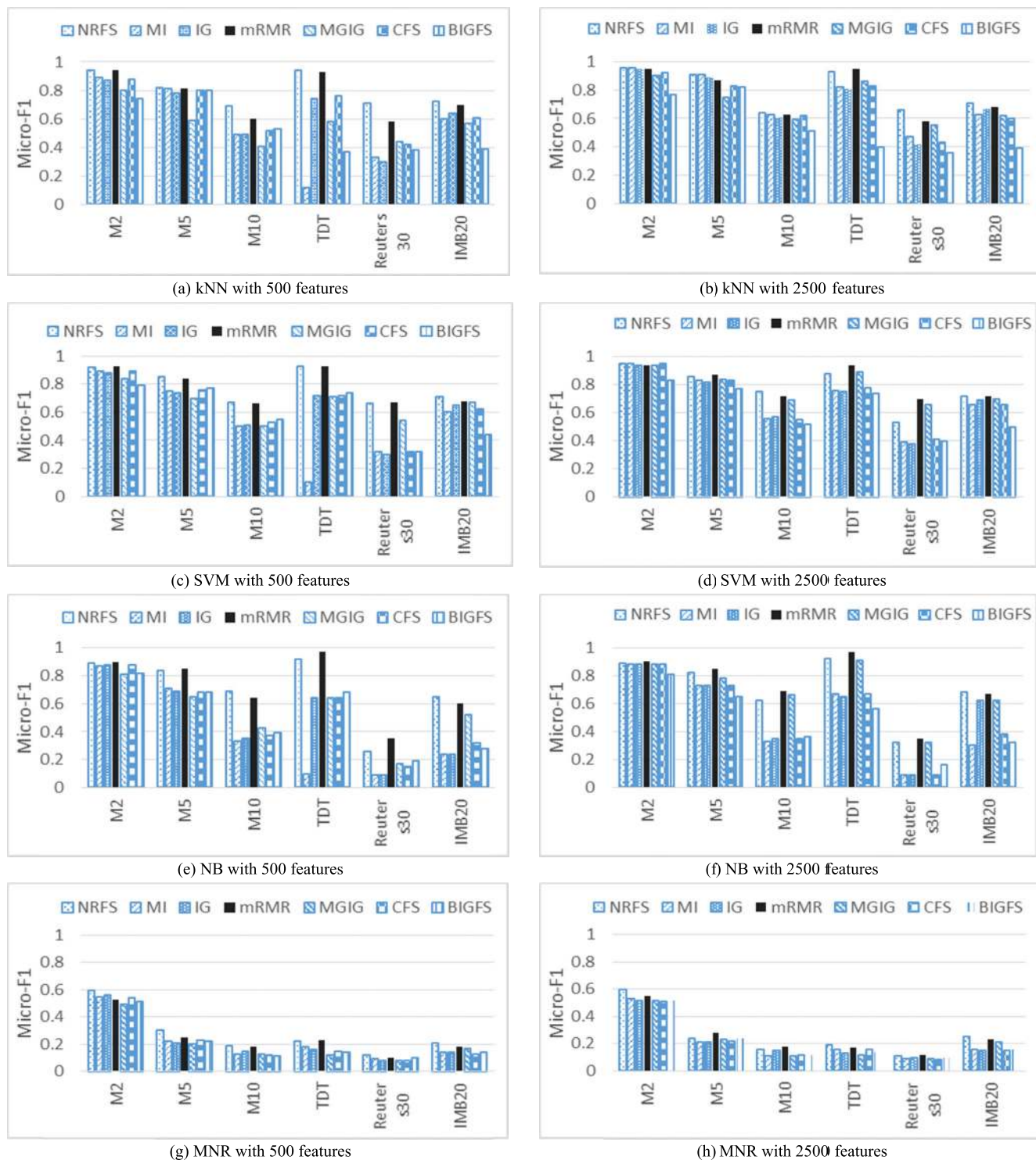
1. Let  $Y$  be a  $m \times 1$  vector having target labels  $c_j$ , where  $j \in 1 \dots n$
2. For each document label  $c_j$  corresponding to document  $Y_i$ , generate a random number  $x$ .

$$\begin{aligned} &\text{if } x \leq p \\ &\quad \text{set } Y_i = c_l \text{ where } i \neq l \\ &\text{else} \\ &\quad Y_i = c_j \end{aligned}$$

### 2) RESULTS OF FEATURE SELECTION WITH LABEL NOISE

To evaluate the robustness of the algorithm to label noise, we use the same datasets as before. We fix the number of features to be selected at 500 and introduce varying percentage of label noise in the training dataset by setting the value of  $p$  from 5% to 20% with steps of 5%. We select the desired number of features using the different feature selection techniques and test their classification accuracy. We then use the NRFS algorithm to select the features and use  $k$ NN, Naïve Bayesian, SVM and MLR classifiers as before to classify the test dataset. The results corresponding to the micro-F1 scores are presented in Table 5 (macro-F1 scores for imbalanced datasets are given in Appendix ).

We would like to mention here that there exist several classification algorithms that have been specifically developed for handling label noise. For instance, the authors in [43] have proposed an algorithm that handles label noise explicitly while trying to learn from the training dataset. Similarly,



**FIGURE 2.** Comparison of Micro-F1 values for NRFS with other feature selection methods for different number of features.

there exists techniques that can be used to reduce the effect of label noise by pre-processing the data or by employing several algorithms to independently classify the data, and then using a consensus approach to form the final classification. These techniques, for instance, boosting and bagging,

ensemble methods, etc., can be used to reduce the effects that label noise might have on a single algorithm. In this paper, however, we avoid these algorithms since we are primarily concerned with the task of feature selection and would like to analyze its effect on document classification. Employing

TABLE 6. Micro-average F1 score in the presence of label noise (500 features) for NRFS.

			5%	10%	15%	20%
Balanced Datasets	M2	kNN	0.83±0.03	<b>0.81±0.05</b>	<b>0.81±0.06</b>	<b>0.76±0.06</b>
		SVM	<b>0.85±0.03</b>	0.82±0.02	0.78±0.04	0.75±0.04
		NB	0.77±0.08	0.70±0.09	0.67±0.11	0.64±0.11
		MLR	0.60±0.02	0.61±0.02	0.59±0.03	0.56±0.03
	M5	kNN	<b>0.74±0.03</b>	<b>0.72±0.04</b>	<b>0.67±0.04</b>	0.66±0.03
		SVM	0.71±0.03	0.66±0.02	0.61±0.03	<b>0.58±0.03</b>
		NB	0.56±0.05	0.48±0.04	0.41±0.05	0.37±0.07
		MLR	0.30±0.02	0.32±0.03	0.31±0.03	0.28±0.03
	M10	kNN	<b>0.67±0.03</b>	<b>0.64±0.04</b>	<b>0.62±0.05</b>	<b>0.59±0.03</b>
		SVM	0.64±0.03	0.59±0.02	0.53±0.03	0.53±0.03
		NB	0.56±0.03	0.53±0.04	0.46±0.06	0.41±0.05
		MLR	0.23±0.02	0.21±0.02	0.23±0.03	0.23±0.03
TDT	kNN	<b>0.95±0.01</b>	<b>0.94±0.01</b>	<b>0.94±0.01</b>	<b>0.94±0.01</b>	
	SVM	0.91±0.01	0.9±0.01	0.89±0.01	0.88±0.01	
	NB	0.93±0.01	0.92±0.02	0.93±0.02	0.92±0.02	
	MLR	0.23±0.02	0.25±0.03	0.22±0.03	0.23±0.03	
Imbalanced Datasets	IMB20	kNN	0.66±0.01	0.65±0.01	0.64±0.01	0.62±0.01
		SVM	<b>0.67±0.01</b>	<b>0.67±0.01</b>	<b>0.65±0.01</b>	<b>0.63±0.01</b>
		NB	0.62±0.01	0.60±0.02	0.56±0.02	0.56±0.02
		MLR	0.22±0.02	0.24±0.03	0.23±0.03	0.22±0.03
	Reuters	kNN	<b>0.71±0.02</b>	<b>0.69±0.02</b>	<b>0.68±0.04</b>	<b>0.68±0.02</b>
		SVM	0.65±0.03	0.61±0.04	0.6±0.03	0.57±0.03
		NB	0.2±0.07	0.15±0.03	0.1±0.04	0.06±0.02
		MLR	0.13±0.04	0.14±0.05	0.13±0.04	0.12±0.03

these sophisticated methods would make it harder to study the effect resulting from the feature selection process and might be influenced by the pre-processing steps or the classifiers. Hence, we used the same standard pre-processing steps and classifiers as previously. Interested readers may refer to [23] for a recent, comprehensive survey on classification in the presence of label noise.

We re-ran the experiments on the different datasets with varying amounts of label noise. As previously, we create 10 sets for each dataset by randomly selecting the documents and report the average micro-F1 values. The results are shown in Table 6. Firstly, in the M2 dataset with only two classes, we observe that using NRFS the results remains fairly good and accurate even with up to 15% label noise. Even with higher number of categories, the results are quite stable and only after introducing 20% noise does the micro-F1 value shows a decline, except in case of dataset TDT in which again stable results can be observed. Secondly, the effect of noise is also dependent on the classifier.

The  $k$ NN classifier, for instance, uses the top  $k$  nearest neighbors to decide the class of a test document. It is reasonable to assume that, in the presence of label noise, the distribution of features occurring in the mislabeled data will be disturbed. Features that were very informative (occurred in mostly a single category) will now be distributed across different categories. The modified MI measure used by the proposed algorithm will result in a lower score for such features, thereby reducing their chance of being selected. Hence, those features that mainly occur in a single category, i.e. features that are both informative and occur in the correctly labeled documents, will tend to be selected. This, in turn, will

result in documents with the correct labels getting chosen as the nearest neighbors.

Sophisticated algorithms such as SVM, on the other hand, use the training data to create a hyperplane. Mislabeling the training data would have an effect on the support vectors, and hence, the learned hyperplane weights. This problem is compounded when using multiple hyperplanes for multi-class classification. This effect is also reflected in our results and the micro-F1 scores using SVM show a relatively significant impact with the introduction of label noise, except in case of M2 where SVM shows better performance than the other two classifiers. Similarly, the Naïve Bayes classifier uses probability, which may also be affected by label noise.

Furthermore, we compare the impact of noise when using the other feature selection methods – MI, IG, mRMR and MGIG. We plot the micro-F1 values for the 2 boundary cases of 5% and 20% noise (where the micro-F1 score starts decreasing).

The results are shown in Figure 3. Using NRFS, the results are significantly better than the traditional techniques such as MI and IG. The same is true of the other higher-order feature selection algorithms – mRMR and MGIG. When comparing NRFS to the other higher order algorithms, the proposed NRFS technique is seen to be slightly less accurate in datasets with a smaller number of categories. However, it significantly outperforms all algorithms for the M10, Reuters and TDT datasets having 10, 30 and 20 categories respectively. Thus, as seen previously, NRFS better selects the feature set when the number of categories is higher making it more robust to introducing label noise.

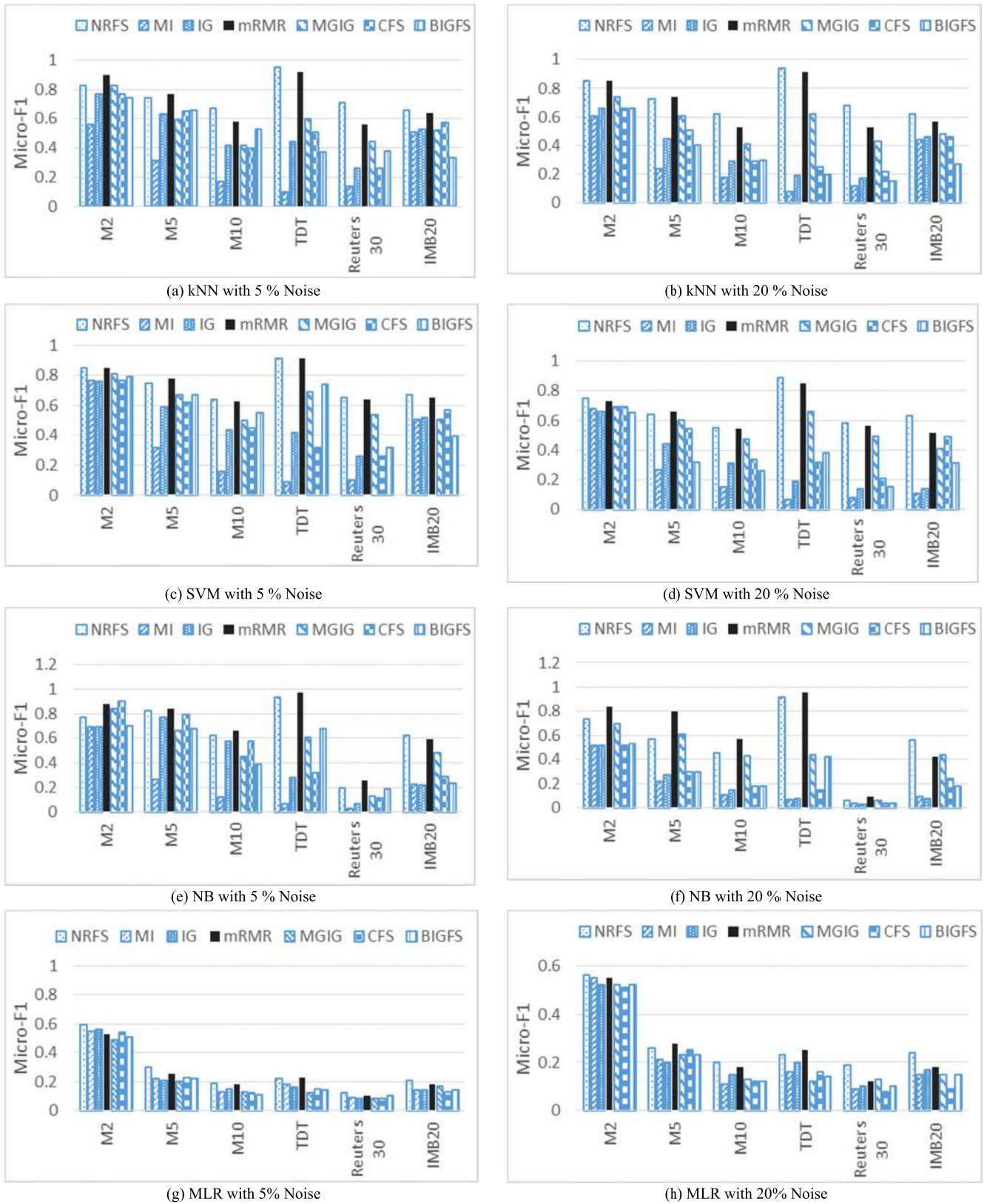


FIGURE 3. Comparison of NRFS with other feature selection methods at different label noise levels.

As previously, the result when using  $k$ NN is higher than the other two classifiers, while SVM also outperforms the NB approach. The mRMR algorithms gives better performance for lower number of categories, while NRFS is comparable to the MGIG feature selection method.

#### F. COMPLEXITY ANALYSIS OF NRFS

Finally, we show the computational behavior when using our NRFS algorithm. As mentioned in Section III, subsection E, the time complexity of NRFS is dependent on the number of selected features and the number of documents since each feature require comparison with a vector of size  $m \times 1$ , where  $m$  is the number documents. All experiments were performed using a Core 2 Duo machine running Windows 8® with 2GB of memory. The results are shown in Figure 4 below.

The baseline algorithms MI, IG, BIGFS have similar time complexities (see Section III, subsection D) but give a much lower micro-F1 value. MGIG, while an improvement over MI and IG, also results in lower micro-F1 values as compared to NRFS (note that MGIG has a higher complexity than NRFS). The interesting comparison here is between the proposed NRFS and the mRMR (and CFS, which has similar complexity) algorithm as these are the two best performing algorithms. We ignore the time to calculate the MI score since it is used by all the algorithms. The results are averages over 10 runs and show the time needed to select the features once the score has been computed and the features sorted. It is evident that the time taken using NRFS is significantly lower than the time taken by the mRMR algorithm and only slightly higher than the base line MI based method (since the running time shown here is in additional to using MI, we assume a constant zero for using MI). In comparison, the gain in micro-F1 score when using NRFS as compared to simple MI is significant as seen in the previous sections.

The mRMR criteria, while also resulting in a better micro-F1 score than MI and, when the number of features are higher, than NRFS too, comes at a high computational cost. At each iteration, new features need to be compared to all existing features for MI. Even using a smart implementation, where the results of previous comparisons are stored, the time complexity is still very high. In our case, the total number of training data features,  $v$ , is roughly 63,000. In most medium to large size text corpora, these features easily run into the hundreds of thousands which will further increase the time complexity of mRMR.

#### G. SIGNIFICANCE TEST

Finally, we perform a significance test to validate the superiority of using NRFS as compared to other tested methods. This test is necessary to indicate the statistical superiority of an algorithm compared to other techniques. For instance, since data is selected at random to generate training and test partitions, it is possible that one feature selection algorithm could perform better in one test but worse in the other. In our results, we used a paired sample  $t$ -test which is used to determine whether there is a significant difference between

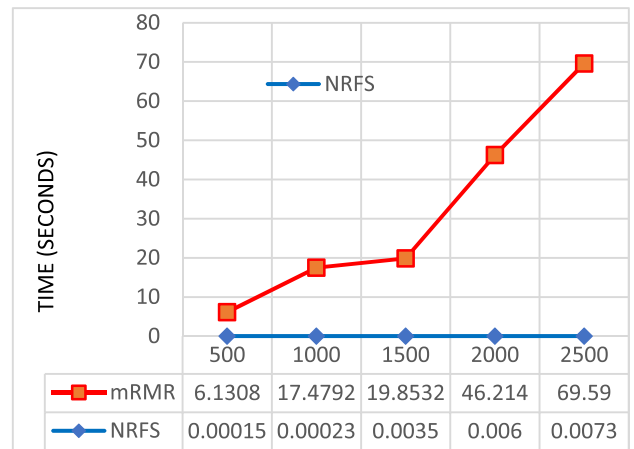


FIGURE 4. Time taken to select features using NRFS and mRMR selection criteria (seconds).

the average values of the same measurement made under two different conditions (algorithms). The test is based on the paired differences between the distributions of the accuracy values produced by the two algorithms being compared. The usual null hypothesis is that the difference in their mean values is zero while the alternate hypothesis is that they belong to different distributions.

We chose the SVM classifier for this test as it is amongst the most frequently used classification technique. Moreover, the results presented in the earlier sections also show that SVM gives the best overall accuracy values across the datasets and, therefore, is the classifier of choice for classifying data. The accuracy value is noted for each of the algorithm on the same dataset repeatedly to form the two distributions. The results are shown in Table 7 and Table 8. We test the hypothesis at  $\alpha = 0.05$  meaning that a  $p$ -value less than 0.05 denotes a rejection of the null hypothesis with smaller values signifying a strong rejection. The  $t$ -value represents the difference between the distributions in units of standard error, i.e. a higher value denotes a strong rejection of the null hypothesis. We can see that in all the cases, the results generated by NRFS are statistically significant meaning that the null hypothesis is rejected except in the case of mRMR. Even then, in the case of noise, the results generated by NRFS are both higher and statistically significant. Moreover, since mRMR takes a lot more time to execute as seen previously (Figure 4), we can ignore this value.

#### V. DISCUSSION AND CONCLUSION

In this paper, we have proposed a novel feature selection technique that is based on the score-based feature selection method using mutual information. Our experiments using benchmark datasets and standard classifiers show that NRFS outperforms the traditional MI feature selection as well as a more sophisticated MRMR and MGIG techniques in most of the datasets tested. Moreover, the proposed NRFS algorithm has been shown to be more robust to label noise. From our results, NRFS gives a higher level of micro-F1 measure when



**TABLE 7. NRFS vs other algorithms on different Datasets (using SVM, 500 features, and 0% noise).**

	<i>h</i>	Paired differences				<i>t</i> -val	df	<i>p</i> -val	
		Mean	Std. Dev.	Std. Err. Mean	Confidence interval				
					Lower				Upper
NRFS – MI	1	0.3422	0.0454	0.0143	0.3120	0.3724	23.8229	118	0.0000
NRFS – IG	1	0.3610	0.0496	0.0157	0.3272	0.3948	22.334	118	0.0000
NRFS – mRMR	0	0.0267	0.0543	0.0172	-0.0085	0.0618	1.5945	118	0.1282
NRFS – MGIG	1	0.1316	0.0847	0.0268	0.0811	0.1821	5.4783	118	0.0000
NRFS – CFS	1	0.1169	0.0436	0.0138	0.0864	0.1475	8.0353	118	0.0000
NRFS – BIGFS	1	0.1927	0.0434	0.0137	0.1571	0.2283	11.3638	118	0.0000

**TABLE 8. NRFS significance with other using different datasets (using SVM, 500 features, and 20 % noise).**

	<i>h</i>	Paired differences				<i>t</i> -val	df	<i>p</i> -val	
		Mean	Std. Dev.	Std. Err. Mean	Confidence interval				
					Lower				Upper
NRFS – MI	1	0.2502	0.0509	0.0161	0.2140	0.2865	14.4947	118	0.0000
NRFS – IG	1	0.1801	0.0598	0.0189	0.1406	0.2196	9.5822	118	0.0000
NRFS – mRMR	1	0.0468	0.0423	0.0134	0.0077	0.0860	2.5126	118	0.0217
NRFS – MGIG	1	0.0893	0.0698	0.0221	0.0491	0.1295	4.6643	118	0.0002
NRFS – CFS	1	0.1353	0.0730	0.0231	0.0938	0.1769	6.8508	118	0.0000
NRFS – BIGFS	1	0.1931	0.0603	0.0191	0.1549	0.2313	10.6100	118	0.0000

**TABLE 9. Average Macro-F1 score by varying number of features using NRFS.**

		500	1000	1500	2000	2500
IMB20	kNN	0.70±0.02	0.70±0.01	0.71±0.02	0.72±0.01	0.71±0.02
	SVM	0.69±0.01	0.69±0.01	0.70±0.01	0.71±0.01	0.71±0.01
	NB	0.63±0.02	0.64±0.02	0.67±0.02	0.66±0.02	0.68±0.02
	MLR	0.19±0.03	0.19±0.01	0.20±0.03	0.25±0.02	0.27±0.03
Reuters30	kNN	0.67±0.02	0.67±0.02	0.68±0.02	0.65±0.02	0.63±0.02
	SVM	0.60±0.02	0.62±0.04	0.64±0.04	0.64±0.04	0.62±0.06
	NB	0.22±0.04	0.22±0.03	0.23±0.03	0.22±0.03	0.24±0.03
	MLR	0.12±0.03	0.12±0.03	0.13±0.02	0.14±0.02	0.16±0.02

the selected feature set is small, since we guarantee that each incoming feature is both highly discriminative and non-redundant, covering all documents in the training data.

The modified scoring criterion is not only limited to feature selection but can also be used in other areas of research, for instance in term weighting schemes. We have shown that this criterion outperforms the simple average or maximum mutual information between a feature and all categories. Moreover, the non-redundant feature selection heuristic proposed in this paper has an advantage over existing techniques such as the MRMR and MGIG. The feature selection criteria of NRFS is much more efficient making it capable of handling large amounts of data while maintaining the same accuracy as the other mentioned techniques.

As a future work, we intend to extend this algorithm to deal with the task of multi-label classification, where the statistical co-occurrence relation when selecting new features

**TABLE 10. Average Macro-F1 score by varying Noise factor of features using NRFS.**

		5%	10%	15%	20%
IMB20	kNN	0.59±0.01	0.59±0.01	0.58±0.02	0.56±0.02
	SVM	0.61±0.01	0.60±0.01	0.60±0.01	0.58±0.01
	NB	0.55±0.03	0.55±0.03	0.54±0.04	0.54±0.04
	MLR	0.17±0.02	0.17±0.02	0.18±0.03	0.16±0.03
Reuters30	kNN	0.51±0.02	0.51±0.02	0.51±0.04	0.50±0.02
	SVM	0.55±0.03	0.55±0.02	0.54±0.02	0.53±0.02
	NB	0.19±0.07	0.17±0.03	0.17±0.04	0.16±0.02
	MLR	0.12±0.04	0.12±0.05	0.11±0.04	0.12±0.03

could potentially be exploited to determine the multiple categories to which a document belongs. We are also pursuing the ideas presented in this paper as a term weighting scheme in document classification to increase the micro-F1 score of a classifier when using the vector space model.

**APPENDIX**

See tables 7–10.

**REFERENCES**

[1] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge, U.K.: Cambridge Univ. Press, 2007.  
 [2] S. F. Hussain and A. Suryani, “On retrieving intelligently plagiarized documents using semantic similarity,” *Eng. Appl. Artif. Intell.*, vol. 45, pp. 246–258, Oct. 2015.

- [3] G. G. Dagher and B. C. M. Fung, "Subject-based semantic document clustering for digital forensic investigations," *Data Knowl. Eng.*, vol. 86, pp. 224–241, Jul. 2013.
- [4] Y. Liang, K. P. Chow, L. C. K. Hui, J. Fang, S. M. Yiu, and S. Hou, "Towards a better similarity measure for keyword profiling via clustering," in *Proc. IEEE 37th Annu. Comput. Softw. Appl. Conf. Workshops (COMPSACW)*, Jul. 2013, pp. 16–20.
- [5] D. Wang, J. Wu, H. Zhang, K. Xu, and M. Lin, "Towards enhancing centroid classifier for text classification-A border-instance approach," *Neurocomputing*, vol. 101, pp. 299–308, Feb. 2013.
- [6] A. Kalousis, J. Prados, and M. Hilario, "Stability of feature selection algorithms: A study on high-dimensional spaces," *Knowl. Inf. Syst.*, vol. 12, no. 1, pp. 95–116, May 2007.
- [7] G. Zervas and S. M. Ruger, "The curse of dimensionality and document clustering," in *Proc. Artif. Intell. Inf.*, Glasgow, U.K., 1999, p. 19.
- [8] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Netw.*, vol. 2, no. 5, pp. 359–366, Jan. 1989.
- [9] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in *Proc. 10th Eur. Conf. Mach. Learn. (ECML)*, 1998, pp. 137–142.
- [10] S. F. Hussain, G. Bisson, and C. Grimal, "An improved co-similarity measure for document clustering," in *Proc. 9th Int. Conf. Mach. Learn. Appl.*, Dec. 2010, pp. 190–197.
- [11] S. F. Hussain and G. Bisson, "Text categorization using word similarities based on higher order co-occurrences," in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Columbus, OH, USA, 2010, pp. 1–12.
- [12] S. F. Hussain and S. Bashir, "Co-clustering of multi-view datasets," *Knowl. Inf. Syst.*, vol. 47, no. 3, pp. 545–570, Jun. 2016.
- [13] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Comput. Electr. Eng.*, vol. 40, no. 1, pp. 16–28, Jan. 2014.
- [14] N. Kushwaha and M. Pant, "Link based BPSO for feature selection in big data text clustering," *Future Gener. Comput. Syst.*, vol. 82, pp. 190–199, May 2018.
- [15] A. Fahad, Z. Tari, I. Khalil, A. Almalawi, and A. Y. Zomaya, "An optimal and stable feature selection approach for traffic classification based on multi-criterion fusion," *Future Gener. Comput. Syst.*, vol. 36, pp. 156–169, Jul. 2014.
- [16] I. Guyon, S. Gunn, M. Nikravesh, and L. Zadeh, *Feature Extraction, Foundations and Applications*, 1st ed. Berlin, Germany: Springer, 2006.
- [17] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 1997, pp. 412–420.
- [18] X. Zhao, D. Li, B. Yang, H. Chen, X. Yang, C. Yu, and S. Liu, "A two-stage feature selection method with its application," *Comput. Electr. Eng.*, vol. 72, pp. 468–481, Oct. 2018.
- [19] S. P. Rajamohana and K. Umamaheswari, "Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection," *Comput. Electr. Eng.*, vol. 67, pp. 497–508, Apr. 2018.
- [20] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
- [21] P. A. Estevez, M. Tesmer, C. A. Perez, and J. M. Zurada, "Normalized mutual information feature selection," *IEEE Trans. Neural Netw.*, vol. 20, no. 2, pp. 189–201, Feb. 2009.
- [22] A. A. Shanab, T. M. Khoshgoftar, R. Wald, and A. Napolitano, "Impact of noise and data sampling on stability of feature ranking techniques for biological datasets," in *Proc. IEEE 13th Int. Conf. Inf. Reuse Integr. (IRI)*, Aug. 2012, pp. 415–422.
- [23] B. Frenay and M. Verleysen, "Classification in the presence of label noise: A survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 845–869, May 2014.
- [24] D. D. Lewis, "Feature selection and feature extraction for text categorization," in *Proc. Workshop Speech Natural Lang.*, 1992, pp. 212–217.
- [25] C. Shang, M. Li, S. Feng, Q. Jiang, and J. Fan, "Feature selection via maximizing global information gain for text classification," *Knowl.-Based Syst.*, vol. 54, pp. 298–309, Dec. 2013.
- [26] Y. Li, C. Luo, and S. M. Chung, "Text clustering with feature selection by using statistical data," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 5, pp. 641–652, May 2008.
- [27] H. Liu, J. Li, and L. Wong, "A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns," *Genome Informat.*, vol. 13, pp. 51–60, 2002.
- [28] V. Bolón-Canedo, N. Sánchez-Marroño, and A. Alonso-Betanzos, "A review of feature selection methods on synthetic data," *Knowl. Inf. Syst.*, vol. 34, no. 3, pp. 483–519, Mar. 2013.
- [29] J. Peng, S. Tang, L. Zhang, and R. Liu, "Information retrieval of mass encrypted data over multimedia networking with N-level vector model-based relevancy ranking," *Multimedia Tools Appl.*, vol. 76, no. 2, pp. 2569–2589, Jan. 2017.
- [30] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Trans. Neural Netw.*, vol. 5, no. 4, pp. 537–550, Jul. 1994.
- [31] Z. Gao, Y. Xu, F. Meng, F. Qi, and Z. Lin, "Improved information gain-based feature selection for text categorization," in *Proc. 4th Int. Conf. Wireless Commun., Veh. Technol., Inf. Theory Aerosp. Electron. Syst. (VITAE)*, May 2014, pp. 1–5.
- [32] E. Kikтова-Vozarikova, J. Juhar, and A. Cizmar, "Feature selection for acoustic events detection," *Multimedia Tools Appl.*, vol. 74, no. 12, pp. 4213–4233, Jun. 2015.
- [33] R. Hu, D. Cheng, W. He, G. Wen, Y. Zhu, J. Zhang, and S. Zhang, "Low-rank feature selection for multi-view regression," *Multimedia Tools Appl.*, vol. 76, no. 16, pp. 17479–17495, Aug. 2017.
- [34] G. Bisson and F. Hussain, "Chi-sim: A new similarity measure for the co-clustering task," in *Proc. 7th Int. Conf. Mach. Learn. Appl.*, 2008, pp. 211–217.
- [35] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-theoretic co-clustering," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, New York, NY, USA, 2003, pp. 89–98.
- [36] S. Chakraborti, N. Wiratunga, R. Lothian, and S. Watt, "Acquiring word similarities with higher-order association mining," in *Proc. Int. Conf. Case-Based Reasoning*, Belfast, U.K., Aug 2007, pp. 61–76.
- [37] H. Uğuz, "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm," *Knowl.-Based Syst.*, vol. 24, no. 7, pp. 1024–1032, Oct. 2011.
- [38] J. Yang, Y. Liu, X. Zhu, Z. Liu, and X. Zhang, "A new feature selection based on comprehensive measurement both in inter-category and intra-category for text categorization," *Inf. Process. Manage.*, vol. 48, no. 4, pp. 741–754, Jul. 2012.
- [39] A. McCallum and K. Nigam, "A comparison of event models for naive bayes text classification," in *Proc. AAAI Workshop Learn. Text Categorization*, vol. 752, 1998, pp. 41–48.
- [40] J. Weston and C. Watkins, "Multi-class support vector machines," Univ. London, Surrey, U.K., Tech. Rep. CSD-TR-98-04, 1998.
- [41] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Hoboken, NJ, USA: Wiley, 2001.
- [42] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2002, pp. 841–848.
- [43] N. Natarajan, I. Dhillon, P. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 1196–1204.
- [44] K. M. Schneider, "Weighted average pointwise mutual information for feature selection in text categorization," in *Proc. Eur. Conf. Princ. Data Mining Knowl. Discovery*. Berlin, Germany: Springer, 2005, pp. 252–263.
- [45] H. H. Yang and J. Moody, "Data visualization and feature selection: New algorithms for non-Gaussian data," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, pp. 687–693.
- [46] N. Kwak and C.-H. Choi, "Input feature selection for classification problems," *IEEE Trans. Neural Netw.*, vol. 13, no. 1, pp. 143–159, 2002.
- [47] J. Novovičová, P. Somol, M. Haindl, and P. Pudil, "Conditional mutual information based feature selection for classification task," in *Proc. Iberoamerican Congr. Pattern Recognit.*, 2007, pp. 417–426.
- [48] Y. Zhang and Z. Zhang, "Feature subset selection with cumulate conditional mutual information minimization," *Expert Syst. Appl.*, vol. 39, no. 5, pp. 6078–6088, Apr. 2012.
- [49] G. Herman, B. Zhang, Y. Wang, G. Ye, and F. Chen, "Mutual information-based method for selecting informative feature sets," *Pattern Recognit.*, vol. 46, no. 12, pp. 3315–3327, Dec. 2013.
- [50] J. Xu and H. Jiang, "An improved information gain feature selection algorithm for SVM text classifier," in *Proc. Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discovery*, Sep. 2015, pp. 273–276.
- [51] R. C. P. Fragoso, H. W. P. Roberto, and D. C. C. George, "Class-dependent feature selection algorithm for text categorization," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, 2016, pp. 3508–3515.

- [52] H. Peng and Y. Fan, "Feature selection by optimizing a lower bound of conditional mutual information," *Inf. Sci.*, vols. 418–419, pp. 652–667, Dec. 2017.
- [53] N. Hoque, H. A. Ahmed, D. K. Bhattacharyya, and J. K. Kalita, "A fuzzy mutual information-based feature selection method for classification," *Fuzzy Inf. Eng.*, vol. 8, no. 3, pp. 355–384, Sep. 2016.
- [54] R. Chen, N. Sun, X. Chen, M. Yang, and Q. Wu, "Supervised feature selection with a stratified feature weighting method," *IEEE Access*, vol. 6, pp. 15087–15098, 2018.
- [55] W. Gao, L. Hu, P. Zhang, and J. He, "Feature selection considering the composition of feature relevancy," *Pattern Recognit. Lett.*, vol. 112, pp. 70–74, Sep. 2018.
- [56] Q. Al-Tashi, S. J. A. Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Binary optimization using hybrid grey wolf optimization for feature selection," *IEEE Access*, vol. 7, pp. 39496–39508, 2019.
- [57] H. Peng, C. Ying, S. Tan, B. Hu, and Z. Sun, "An improved feature selection algorithm based on ant colony optimization," *IEEE Access*, vol. 6, pp. 69203–69209, 2018.
- [58] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing*, vol. 300, pp. 70–79, Jul. 2018.
- [59] C. Wang, Y. Huang, M. Shao, Q. Hu, and D. Chen, "Feature selection based on neighborhood self-information," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 4031–4042, Sep. 2020.
- [60] R. Cekik and A. K. Uysal, "A novel filter feature selection method using rough set for short text data," *Expert Syst. Appl.*, vol. 160, Dec. 2020, Art. no. 113691.
- [61] Y. Liu, S. Ju, J. Wang, and C. Su, "A new feature selection method for text classification based on independent feature space search," *Math. Problems Eng.*, vol. 2020, pp. 1–14, May 2020.
- [62] A. H. Hossny, L. Mitchell, N. Lothian, and G. Osborne, "Feature selection methods for event detection in Twitter: A text mining approach," *Social Netw. Anal. Mining*, vol. 10, no. 1, pp. 1–15, Dec. 2020.
- [63] X. Deng, Y. Li, J. Weng, and J. Zhang, "Feature selection for text classification: A review," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3797–3816, 2019.
- [64] W. Liu and J. Wang, "A brief survey on nature-inspired metaheuristics for feature selection in classification in this decade," in *Proc. IEEE 16th Int. Conf. Netw., Sens. Control (ICNSC)*, May 2019, pp. 424–429.
- [65] W. Gao, L. Hu, and P. Zhang, "Feature redundancy term variation for mutual information-based feature selection," *Appl. Intell.*, vol. 50, no. 4, pp. 1272–1288, Apr. 2020.



**SYED FAWAD HUSSAIN** (Senior Member, IEEE) received the M.S. degree in computer science from Pierre and Marie Curie University, Paris, and the Ph.D. degree in computer science from the University of Grenoble, France.

He is currently an Associate Professor with the Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology. He is also a Seasoned Researcher with several dozen research publications to his credit. His current research interests include machine learning, big data, unsupervised learning, similarity metrics, information retrieval, and bioinformatics. He has also contributed as a technical member in several committees at the national level. He is a Professional Member of ACM.



**HAFIZ ZAHEER-UD-DIN BABAR** received the B.S. degree in computer science from the Namal College, University of Bradford, and the M.S. degree from the Ghulam Ishaq Khan Institute of Engineering Science and Technology. He is currently pursuing the Ph.D. degree with Radboud University, The Netherlands.

His research interests include data mining and machine learning, particularly development of new algorithms.



**AKHTAR KHALIL** (Member, IEEE) received the Ph.D. degree in information and communications technology from the University of Loughborough.

He has vast experience with the Research and Development career. He is currently the Director of Research and Development, Ifahja Ltd., U.K. He has many articles and patents to his credit. His research interests include pattern recognition, document analysis, content-based image retrieval, and image processing. He received the Top 20 British

Inventions of the Year from the Gadget Show Live, U.K., the Toyota Manufacturing Innovation Challenge Award, the National Innovation Award, Pakistan, and so on.



**RASHAD M. JILLANI** (Senior Member, IEEE) received the M.Sc. degree in computer science from IIU, Islamabad, Pakistan, in 2000, and the Ph.D. degree in computer engineering from Florida Atlantic University, in 2012.

He has over ten years of industrial experience as a Software Engineer. He was with several renowned companies, such as Verizon Inc., AT&T Laboratories, and so on. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology. He is also an Expert on digital audio-visual communications systems with over ten years of experience in multimedia research, development, and standardization. His research interests include video compression, video transcoding, and complexity reduction in mobile video devices.



**MUHAMMAD HANIF** received the M.Sc. degree in information technology (signal processing) from the Tampere University of Technology, Tampere, Finland, and the Ph.D. degree in image processing from the College of Engineering and Computer Science, The Australian National University, Canberra, ACT, Australia.

He was a Researcher with the Faculty of Engineering, Tampere University of Technology. He was with the Computer Vision and Robotics Research Group, National ICT Australia (NICTA), DTAT61. He was also with the CNR-Istituto di Scienza e Tecnologie dell'Informazione A. Faedo. He is currently an Assistant Professor with the Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan. His main research interests include blind image deconvolution and sparse image processing. He received the European Research Consortium for Informatics and Mathematics (ERCIM) Fellowship for the Postdoctoral from the Italian National Research Council (CNR), Pisa, Italy, from January 2017 to November 2019.



**KHURRAM KHURSHID** (Member, IEEE) received the M.S. and Ph.D. degrees from Paris Descartes University, France, in 2006 and 2009, respectively.

Since 2011, he has been a Professor with the Institute of Space Technology, Islamabad, where he heads the Department of Electrical Engineering. He is also a Project Manager with the Small Satellite Program, iCUBE. He has several publications to his credit and worked on several funded projects. His research interests include pattern recognition, document analysis, and image processing. He serves as an Editor for *Journal of Space Technology*.