Journal of Theoretical and Applied Information Technology
30th September 2021. Vol.99. No 18

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL FILTER BASED MULTIVARIATE FEATURE SELECTION TECHNIQUE FOR TEXT CLASSIFICATION

[1]RAVI KUMAR PALACHARLA, [2]VALLI KUMARI VATSAVAYI

[1]Department Of Information Technology, Bapatla Engineering College, Bapatla, AP

[2]Department Of CS&SE, AUCE (A), Andhra University, Visakhapatnam, AP

Email: [1]palacharla.ravikumar@gmail.com, [2]vallikumari@gmail.com

ABSTRACT

Text classification is a technique of assigning the known class label to the unknown textual documents. This technique assign single label or multiple labels to a specific document based on the content in the document. These techniques are used in various applications such as sentiment analysis, authorship analysis, fake news detection and spam email classification. In the text classification process, the words in the documents are considered as features. The most important words which are having more differentiating power are considered in the representation of a document. Identification of such words or features is a primary step in the classification process. The high dimensionality of data description is a primary issue in text classification. Huge number of features in the analysis not only decreases the performance of classification but also increase the computational time. In this work, a new feature selection technique based on Category specific Feature Distribution without Redundancy Information (CFDRI) is proposed to identify best informative features and eliminating the redundant features. The effectiveness of proposed feature selection technique is compared with existing techniques such as mutual information, information gain, chi square and relative discriminative criterion. The traditional Bag of Words technique is used to designate the documents as vectors. Term frequency and inverse document frequency measure is used to compute the vector value in the document vector representation. Various machine learning algorithms such as Decision Tree, Support Vector Machine, Naïve Bayes, k-Nearest Neighbour, Logistic Regression and Random Forest are used to generate the learned model. Six popular text classification datasets are used in this experiment to train different learning algorithms. The proposed feature selection technique obtained best accuracies for text classification when compared with the popular solutions for text classification.

Key Words: *Text Classification, Feature Selection Techniques, Bag of Words Model, machine Learning Algorithms, Accuracy*

1. INTRODUCTION

The huge volume of electronic textual information is increasing through the social media, digital libraries, news content, web pages and electronic mails with the exponential advancements in the information technologies and internet. To handle such huge information, text classification techniques play a crucial role to categorize the textual information. Text classification is defined as a technique of classifying the textual documents into predefined classes automatically. Text classification techniques are used in several applications like Document classification, Sentiment analysis, spam filtering from e-mails, Topic identification, Authorship Analysis, Web pages classification, Bioinformatics etc.

In the case of text representation, several authors have used different techniques to translate text into a numerical representation. The most common method is Bag of Words (BoW). In this method, the textual content is represented as a vector where each component represents a word or feature in the whole corpus vocabulary appears in the short text or not. Nowadays, most of the real-world problems are often described by a huge number of features. The curse of dimensionality is one of the primary problems faced by machine learning algorithms with huge features count. This problem makes difficult to train a classification algorithm efficiently and effectively on a huge number of features. High dimensionality is problematic to any classification algorithm. All features are not useful for representation and not contain necessary information. The irrelevant features deliver

misleading information, which led to performance degradation in classification. On the other hand, relevant features deliver useful information for the training of a classifier [1]. The redundant features provide similar or same information to the learning process of a classifier.

In general, the goal of feature selection is to recognize a minimal feature subset which is sufficient and necessary to solve classification problems. This task is achieved by removing redundant and irrelevant features from the original set of features. By applying feature selection as a data pre-processing step, it is expected that the less complex dataset will help efficiently train a classifier which is simpler more efficient and accurate than using all features. In a Feature Selection Algorithm (FSA), the searching mechanism and the evaluation criteria are the most important components which significantly affect the quality of final feature subsets. After the final feature subset is generalized, the original dataset is transformed to the new dataset by removing unselected features. New training and test sets which are generated from the new dataset are fed into a learning algorithm to obtain training and testing accuracies respectively.

In general, the extracted features (words) from documents are not always desirable or necessary for the text classification task. It is beneficial to decrease the feature space. The process of reducing the dimensionality is called feature selection. It was first presented by Salton et al., [2] which were able to reduce the document space into vectors and represented each document as one point in the total document space. Sebastian suggested [3] that the feature selection is necessary because the documents feature space is a high-dimensional vector in the set of documents. Another aspect is that the features that are non-relevant for the classification model can misguide the prediction. Noisy or redundant features are also desirable to remove as they can leave out relevant variables or just add computational consumption.

Most of the researchers are used the content bearing words in the documents as features for classification of textual documents. For a dataset that consists of n features, the number of possible feature subsets are $2^n - 1$. Identification of best subset of features by an exhaustive search over all possible subsets is unfeasible. The researchers understand that there is a need for sophisticated techniques to reduce the dimensionality of features. Dimensionality reduction techniques are divided into two classes such as feature selection and feature extraction. Feature extraction involves projecting the data onto a lower-dimensional space by combining or transforming the original features [4]. Principal Component Analysis (PCA) and Non-negative Matrix Factorization (NMF) are examples for feature extraction methods [5]. Feature selection preserves the original features and aims to identify redundant and irrelevant features for the output values in the data set. Both techniques are important in terms of computational time and for enhancing the performance of machine learning algorithms, but feature selection also makes it possible to gain insight on the way output values relate to the original features [6].

Feature selection is usually categorised into three types of methods such as filter method, wrapper method and embedded method [7, 8]. The Filter methods are using statistical methods to select relevant features from original set of features. The Information Gain (IG), Document Frequency Thresholding (DFT), Mutual Information (MI), Gini Index (GI), Chi-Square Statistic (CHI), Odds Ratio (OR) etc., are filter methods. The filter methods are classified into two methods such as univariate methods and multivariate methods. The univariate methods used a specific criterion to estimate the individual feature relevancy. These methods unable to remove redundant features but selects the relevant features effectively. The multivariate methods consider the correlation among the features and are efficient in identifying the relevant features as well as removal of redundant features. These methods are computationally inefficient when compared with univariate methods.

The wrapper method identify different features subsets first and then evaluated these subsets by using learning algorithms. The embedded method includes the process of feature selection into the training phase of learning algorithm. Because of utilization of learning algorithms in wrapper and embedded, these methods achieved good accuracy in the text classification when compared with the filter methods. The filter methods are taking an advantage of less computational time than other methods in dealing of high dimensional feature space. Both the wrapper and the embedded methods are slower and more complex than the filter methods. In comparison with other methods, wrappers generally attain good accuracy because they consider the direct interactions between the feature subset, the class, and the wrapped classification algorithm. However, the selected feature subset is optimized specifically for a learning algorithm. Therefore, a wrapper approach is less general than a filter approach.

The proposed filter based feature selection technique is used in different types of term distribution information such as number of

documents in positive class contain term, number of positive class documents doesn't contain term, number of negative class documents contain term, number of documents in negative class doesn't contain term, no. of classes contains the term, the occurrence count of term in whole dataset, the occurrence count of term in positive class documents, the occurrence count of term in documents of negative class. Different algorithms used different type of information to assign ranks to the terms. The existing measures considers the information of positive and negative class documents which contain term and ignore the number of times term occurred in positive and negative class of documents. In this work, a new filter based feature selection method based on Category specific Feature Distribution without Redundancy Information (CFDRI) is proposed to identify the most relevant features. The proposed CFDRI measure considers all possible information to assign ranks to the terms. This is main advantage of the proposed measure to achieve best accuracies for text classification on different datasets.

The performance of proposed filter method is compared with the performance of existing feature selection algorithms such as mutual information, information gain, relative discriminative criterion and chi square. The experiment conducted on six popular text classification datasets such 20 newsgroup, Reuters-21578, hate speech spreaders, IMDB dataset, Fake news dataset and Clickbaits dataset. Different machine learning algorithms such as Logistic Regression (LR), Naïve Bayes (NB), K-Nearest Neighbour (KNN), Decision Tree (DT) Random Forest (RF) and Support Vector Machine (SVM) are used to train on these datasets. The accuracy metric is used to estimate the efficiency of proposed techniques. The standard traditional BOW model is used to represent the vector representation of documents. Term Frequency Inverse Document Frequency (TFIDF) measure is used to determine the vector value of a feature in the representation of documents.

This chapter is organized in 10 sections. The existing approaches proposed for text classification is discussed in section 2. The machine learning algorithms used in this work are explained section 3. The performance metrics for evaluating the proposed system is described in section 4. The information of datasets used in this work is presented in section 5. The bag of words model is explained in section 6. The importance of feature selection algorithms, some of existing feature selection algorithms, proposed feature selection technique and TFIDF measure are described in section 7. The proposed method is discussed in section 8. The experimental results of feature selection algorithms are presented in section 9. The results are discussed in section 10. The conclusions of this work are mentioned in section 11 with future enhancements.

2. LITERATURE SURVEY

The number of textual information is increasing tremendously in the internet through digital libraries, emails, blogs and reviews. The document classification is an essential task to classify the information into different types. Different researchers performed this classification task by selecting features which contains suitable features to enhance the classification accuracy. In order to address the problem of high dimensionality, J. Li et al., proposed [9] a concept of feature selection to identify more informative and small subset of features from the original feature set. Feature selection techniques decrease the size of original feature set by eliminating redundant and irrelevant features. A key element of any feature selection algorithm is its evaluation criterion for choosing between alternative subsets of features. Most of the text classification approaches based on feature selection methods depends on term weight measures used in feature vector representation. Souad Larabi Marie-Sainte et al., proposed [10] an algorithms based on feature selection technique for Arabic Text Classification. The proposed algorithms applied successfully on various combinatorial problems. The Support Vector Machine with three performance evaluation metrics such as precision, recall and f1-score are used to validate the proposed technique. They used OSAC dataset in the experiment and compared with the popular methods of text classification. The proposed method attained a precision value of 0.994. The experimental results confirm that the accuracy of Arabic Text Classification was improved by using proposed method.

In general, most of the researchers used accuracy, precision, recall and f1-score to estimate the perfromance of the proposed approaches for text classification. Some researchers observed that these measures are not sufficient to evaluate the importance of proposed methods. Gang Kou et al., addressed [11] the problem of Multiple Criteria Decision Making (MCDM). The classification evaluation measures like stability, performance and efficiency need to consider when evaluating the feature selection techniques in text classification with small datasets. Very few researchers used MCDM methods to evaluate the feature selection

based text classification. They exploited MCDM methods for evaluating small datasets by using text classification with feature selection techniques. The researchers designed an experimental study to compare five different MCDM methods such as TOPSIS, VOKOR, GRA, WSM and PROMOTHEE for evaluating the proposed approach. They used 10 feature selection techniques, nine performance measures for binary classification, seven performance measures for multi-class classification. They made recommendations pertaining to feature selection techniques based on the ranked results of MCDM methods. It was observed from the results, the PROMOTHEE performance is good when compared with other MCDM methods.

In machine learning, feature selection is very important step for improving learning accuracy. The researchers developed various multi label feature selection methods to decrease the dimensionality of data because more number of tasks is occurred in different fields based on multi label classification. Most of the existing wrapper based multi label feature selection techniques used multi-objective methods to select features. Hongbin Dong et al., presented [12] a Many objective optimization based Multi label Feature Selection (MMFS) technique to enhance the convergence and diversity of NSGA III. They proposed an enhanced version of NSGA III algorithm with two archives. In improved algorithm, a new mutation and crossover operators are designed for feature selection to enhance the effect of selection threshold θ on feature scale and improving the capability of exploration. They conducted experiment on 11 multi label datasets. The results show that the MMFS is able to eliminate irrelevant and redundant features, balance multiple objectives and achieved acceptable results for classification.

Several research works on machine learning was increased significantly over the last decade. Text classification is one of the research area widely used machine learning algorithms. Most of the big data systems need huge amount of information for analysis purpose. This includes some disadvantages like collection of huge data and processing cost of huge data. To overcome this disadvantage, several practitioners and researchers worked on different techniques to decrease the features count effectively that are used in classification. Tunchan Cura proposed [13] a method to optimize the number of features selected and performance of classification. It was observed from literature survey, most of the studies concentrated on either feature selection or text classification. In their work, the proposed method used the Parallel Local Search (PLS) technique to select best features and determine the classifier to achieve higher accuracy. The proposed method also able to determine solutions within a reasonable computation time for problems that are using extremely more features. The SVM classifier is trained by PLS technique. The SVM classifier selects desired number of features and minimizes the objective function. The minimization of objective function yield decision hyper-plane vector which helps in reduction of number of misclassifications.

The classification of vulnerability is an important task in development of software and maintenance of software quality. A typical model for classification of vulnerability generally includes a step of term selection, wherein identify the relevant terms through feature selection. The model also includes a step of term weighting and a step for training a classifier. In term weighting step, compute the weights of selected terms in a document. In general, most of the solutions to vulnerability classification widely used TF-IDF model as a term weighting metric. However, different issues hinder the efficiency of TF-IDF model for vulnerability classification. To overcome this problem, Jinfu Chen et al., proposed [14] a general framework for classification of vulnerability severity using Term Frequency Inverse Gravity Moment (TF-IGM). They compared TF-IDF and TF-IGM extensively with feature selection technique of information gain using five classification algorithms such as SVM, KNN, NB, DT and RF. The experiment conducted on 10 vulnerable datasets of 10 different software applications which contains 27248 security vulnerabilities. They observed from experimental results that the performance of TF-IGM for classification of vulnerability was good when compared with traditional term weight measures and also observed that the there is an improvement in vulnerability classification performance when feature selection algorithms are used.

The problem of high dimensionality is very important issue in classification of short texts due to its consequences on classifiers accuracy and computational cost. The better solution for this problem is selection of important features which represent the documents in a better way. Rasim Cekik et al., proposed [15] a new filter based feature selection technique known as Proportional Rough Feature Selector (PRFS). This technique used the rough set according to the value set for regional distinction to determine the documents that probably belongs to a class or that exactly belong to a class. The rough set helps to determine the effect of sparsity in the vector representation of terms. The proposed PRFS technique is compared with popular feature selection techniques such as information

gain, gini index, distinguishing feature selector, normalized difference measure and max–min ratio methods. The experiment conducted on four short text datasets by using different feature sizes with Macro-F1 measure. They observed from experimental results that the proposed PRFS achieved competitive or better efficiency when compared with other feature selection techniques in terms of Macro-F1 score.

Feature selection is a technique of determining the optimal feature subset for class label prediction by removing redundant features. The feature selection technique was achieved by nature inspired algorithms like S-shaped Binary Butterfly Optimization Algorithm (S-bBOA). Based on existing research works, the S-bBOA method doesn't consider relevancy and redundancy of features. Zohre Sadeghian et al., proposed [16] a method named as Information Gain binary Butterfly Optimization Algorithm (IG-bBOA) to address the constraints of S-bBOA. IG-bBOA method enhances the mean of the mutual information among class labels and features, the accuracy of classification. This method was used for minimizing the selected features count and also used in Ensemble Information Theory based binary Butterfly Optimization Algorithm (EIT-bBOA). The proposed method divided into three stages. In the first stage, a feature selection technique such as Minimal Redundancy-Maximal New Classification Information (MR- MNCI) was used to remove 80% of redundant and irrelevant features. In the second stage, the IG-bBOA was used to select best feature subset. Finally, the final feature subset is selected by using similarity based ranking method. The experiment conducted with six standard datasets which are collected from UCI repository. The experimental results show that the proposed method is good in most of the cases for enhancing the accuracy of classification.

Automatic text classification was become a primary task in pattern recognition problems because of the number of documents are increasing in digital form. The feature selection techniques are introduced in text classification for dimensionality reduction of feature space and improving the performance of classification. Mahdieh Labani et al., proposed [17] a new filter based feature selection technique called as Multivariate Relative Discrimination Criterion (MRDC). The proposed method used the concepts of maximal-relevancy and minimal-redundancy to concentrate on the decreasing of redundant features. At end, the proposed method estimates the usefulness of term by considering the document frequencies of each term. This method not only considers the redundancy among features using correlation metric but also recognizes the features which are having maximum relevancy with class. In MRDC method, learning algorithms was not used to evaluate the importance of the selected features because it is a filter method. The experiment was conducted on three real-world datasets to estimate the efficiency of the proposed MRDC method. The proposed MRDC performance was compared with other popular filter methods and detected that it shows best performance in most of the cases.

The text classification was become an important task in various applications of data science because of exponential increment of digital documents in the internet. The efficiency of text classification techniques are improved by selecting distinguishable features which are highly relevant with a class and low redundancy with other features. Mahdieh Labani et al., developed [18] a new feature selection technique based on multi-objective algorithm known as Multi-Objective Relative Discriminative Criterion (MORDC) to balance the features with maximal relevant to the target class against minimal redundant features. Durga Prasad Kavadi and Palacharla Ravikumar [19] has proposed a new weight measure to calculate the feature weights and machine learning algorithms such as Random Forest and Naive Bayes Multinomial algorithms produce the classification models by using the vectors of documents. The proposed method searched through solution space by employing a multi-objective evolutionary framework. The first objective function estimates the feature relevancy to the class and second function measures the correlation among the features. The proposed method is a multivariate filter method which was not used any learning algorithm to compute the importance of features selected. The efficiency of the proposed method assessed by conducting experiment on three real-world datasets and observed that the proposed method obtained better performance for classification in most of the cases when compared with popular feature selection techniques.

3. DATASET CHARACTERISTICS

In this work, the most well-known and benchmark datasets in the domain of text classification are used for the experimentation. For covering the aspect of classification type, four binary datasets and two categorical datasets are selected. The selected datasets have different numbers of classes (from 2 to 20), and different numbers of instances (from 200 to 120000). They are also from different real-world

areas such as sentiment analysis, fake news detection and hate speech spreaders detection. The datasets are selected with an expectation that they are well representatives of real-world problems. Table 1 shows the statistics pertaining to the datasets.

Table 1. Dataset Characteristics

S. No.	Dataset	Number of Classes	Number of Instances	Web Link
1	Hate Speech Spreaders (HPS)	2	200 (200 tweets in each instance)	https://pan.webis.de/clef21/pan21-web/author-profiling.html
2	Fake News (FN)	2	25200	https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset
3	IMDB	2	50000	https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews
4	20 News Group (20NG)	20	18828	http://qwone.com/~jason/20Newsgroups/
5	AG News dataset (AGN)	4	127600	https://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html
6	Clikbaits News (CBN)	2	32000	https://www.kaggle.com/vikassingh1996/news-clickbait-dataset

3.1 Hate Speech spreaders

The hate speech spreaders (HSP) dataset [19] contains 200 authors tweets with two classes such as hate speech spreaders, real news spreaders. The dataset is balanced that means both classes contain equal number of files i.e 100. Each author file contains 200 tweets. The positive class is when author file contains tweets of hate speech messages and the negative class is when the author file contains tweets of non-hate speech messages.

3.2 Fake News

Fake news dataset was gathered from different sources like truthful opinions about news articles of Reuters.com, fake news from unreliable Politifact website. The dataset consists of truthful articles of 12600 and fake news articles of 12600 [20]. The dataset is balanced which means that both classes contains equal number of articles. The two classes of news articles collected in 2016. Each article contains minimum number of 200 characters.

3.3 IMDB Movie Review

The Internet Movie Database (IMDB) Movie Review [21] is a sentiment (binary) classification dataset, consisting of 25,000 training and 25,000 testing records.

3.4 20 Newsgroups

The 20 Newsgroups (20 NG) collection [22] is one of the popular

datasets in text clustering and text classification. The dataset includes 18,828 posts on 20 different topics and divided into train

data and test data. The train and test data are divided according to a specific date. Each newsgroup corresponds to a specific topic. 66% of documents in dataset (12426) are used as training data and 34% of documents in dataset (6402) are used as test data.

3.5 AG News

The AG news topic classification dataset [23] contains four classes with 30,000 training samples and 1,900 testing samples for each class, a total of 120,000 training and 7,600 testing records. This dataset is collected by the ComeToMyHead academic search engine from more than 2,000 news sources during a period of almost one year.

3.6 Clickbaits News dataset

The publishers of online content generally used attractive titles or headlines for their content in order to catch the attention of users to their websites. These titles are popularly called as clickbaits which exploit a user's curiosity gap and lure them to click on these links. The clickbait dataset contains two classes such as clickbait and non-clicbait news. The dataset consists of total 32000 rows of which 16000 clickbait labels and 16000 non-clickbait labels [24].

4. EVALUATION MEASURES

The evaluation measures are used to estimate the efficiency of the proposed system with help of machine learning algorithms. In order to give a more accurate evaluation of the results of the classification performance, a confusion matrix is utilised. The confusion matrix includes the original class label on one dimension and the predicted class label on the other dimension. The binary version of the confusion matrix is a particular case of the confutation matrix, having one of the two classes designated to a positive class and the other class described as negative. Table 2 shows the confusion matrix with different possible outcomes of the two classes.

Table 2. Confusion Matrix

		Predicted Class	
		Positive	**Negative**
Actual Class	**Positive**	TP	FN
	Negative	FP	TN

In this table, TP is number of given documents classifies as positive and is also in the actual positive class, FP is number of given documents classifies as positive but is in the actual negative class, FN is number of given documents classifies as negative but is in the actual positive class, TN is number of given documents classifies as negative and is also in the actual negative class.

The researchers used various measures like recall, precision, $F1$-score as well as the accuracy to check the efficiency of the developed system. The Precision is the percentage of documents that the classifier labels as relevant that is actually relevant. Equation (1) is used to calculate Precision.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (1)$$

The Recall is the fraction of positive documents that have been correctly classified over the total amount of positive documents. Equation (2) is used to compute the recall.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (2)$$

F1-Score is the harmonic mean between the precision and recall measures. Equation (3) is used to compute the F1-Score.

$$\text{F1} - \text{Score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3)$$

Accuracy measures the portion of documents that are correctly classified. Equation (4) is used to compute the accuracy. Accuracy ranges from 0 to 1, in which 0 means all documents are incorrectly classified and 1 that all documents are correctly classified.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

The accuracy measure is used in this work to test the efficiency of our proposed approach.

5. MACHINE LEARNING ALGORITHMS (MLAS)

The MLAs are developed a system by examining the characteristics of dataset instead of following the programmed instructions [25, 26]. The performance of machine learning system for a specific task is improved automatically by observing more instances or examples of data. The machine learning tasks are divided into different type of tasks such as supervised, unsupervised, semi-supervised, reinforcement and transfer learning [27, 28].

In supervised learning, all instances or examples given to a machine learning system are labelled by desired outputs, which are known in advance. The

task of supervised learning is to generate a function which maps input to one of the known outputs [29]. Two popular supervised learning tasks are classification and regression. In unsupervised learning, the desired outputs or labels are not known for the instances. The unsupervised learning algorithms extract a common pattern from the instances which are used to group similar instances together. Clustering is probably the most common task in unsupervised learning. In semi-supervised learning only a few labelled instances are provided while most instances are not labelled. This learning technique extract useful information from both labelled and unlabelled instances which is used to either predict the class labels of unlabelled instances correctly or infer a mapping function from the inputs to the outputs.

In reinforcement learning, a machine learning system directly interacts with an environment through a sequence of actions. Each action will result in a reward or punishment based on feedback from the environment. The task is to achieve a certain goal by learning a sequence of actions with the best goodness. In transfer learning, the main task is to reuse knowledge obtained from a source problem to improve the learning performance on different but related target problems. The two problems have different learning tasks, different feature spaces and different data distributions.

Classification is one of the primary tasks in supervised learning, which aims to assign a class label to an unknown instance [30]. In a classification process, a classifier is needed to detect the label of unseen instances. The classifier makes decisions based on values of features that describe the instances. The classifier is obtained by training a classification algorithm on a set of labelled instances. A classification problem is known as a binary classification when dataset contains two class labels only. When the number of class labels is more than two, the classification problem is known as a multi-class problem. An example of classification application is to predict whether news is a fake news or real news. Based on the information of news, the classification algorithm is trained to capture the characteristics of the news. After that, the learned classifier takes the features of news as an input to predict whether the news is real news or fake news.

There are two main processes such as training and testing in a classification system. During the training process, a classification algorithm is learned by using a set of instances, which is called a training set. The learned classifier is then evaluated on another set of instances which are unseen during the training process. The set of instances used in the testing process is called a test set. Each instance is represented by a vector of feature values which are numeric or categorical. The features have a significant effect on the learning time and the classification performance of the learned classifier.

In k-fold cross-validation, a dataset is randomly divided into k subsets or folds with near-equal sizes. The partitioning process ensures that the class distribution in each fold roughly remains the same as in the whole dataset. After that, each fold is then used for testing process exactly once while the rest of the dataset is used for training the classification algorithm. Consequently, the classification algorithm is trained k times, which results in k experiments with k different accuracies. The overall classifier performance is the average of the k accuracies.

In classification, overfitting [31] is a common problem. The goal of classification is to achieve as high testing accuracy as possible. However, when there are too many parameters and the training phase does not involve any regularization pressure, the learned classifier starts remembering all characteristics of the training data. This memorization leads to very high training accuracy. However, there is a risk that the learned model also fits with noisy instances in the training set. Consequently, due to the lack of generality the learned model has a poor prediction ability which results in a low testing accuracy. The phenomenon in which the learned classifier performs well on the training set and badly in predicting the testing instances is called overfitting. In contrast to overfitting, underfitting is another problem where the model to learn is too simple with too few degrees of freedom. Consequently, the learned model does not fit the data well enough which also leads to a poor performance on unseen/testing data.

In this work, the experiment performed on various machine learning algorithms such as K-Nearest Neighbour (KNN), Decision Tree (DT), Naïve Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM) and Random Forest (RF) for text classification.

5.1 K-nearest Neighbour Classifier (KNN)

KNN [32] is a type of instance-based learning approach, which simply remembers all the training instances instead of inducing any classification rule. The new instance is compared with all training instances to determine its class label. Firstly, the distances between the new instance and every training instance are calculated. After that the K nearest training instances (neighbours) of the new instance are identified, where K is a user-predefined

small integer number. The most popular class label among the K nearest neighbours is assigned to the new instance. Many distance measures are used in KNN such as Euclidean distance (continuous data), Manhattan (discrete data). KNN is considered as a lazy-learning algorithm since its learning phase is very minimal. Although KNN is a simple learning algorithm, it performs well on many real-world problems. In addition, KNN is a nonparametric learning algorithm because it does not require any assumption about the probability distribution of the dataset.

A limitation of KNN is it is slow and requires a large memory, especially when the training set has a large number of features or instances. In addition, KNN is sensitive to noise especially when K is small due to no learning rule.

5.2 Decision Tree (DT)

DT [33] is one of the nonparametric classification algorithms in data mining and is applied on numeric, categorical or mixture data types. DT maps instances to class labels by building a tree-based prediction model. In a tree, each inner node is called a decision stump that corresponds to a single feature of the instance. The arc from an inner node is usually labelled by a value of the feature. Each leaf of the tree is a class label. To classify an instance, its feature values will be compared with the decision stump in each inner node until reaching a leaf node. In order to build a tree, the most important step is to determine which feature and the feature's value (splitting point) should be used at each inner node. The most common strategy is a top-down greedy search to select the best feature for each inner node, which can split the source feature set into subsets with smallest impurities. Different DT algorithms use different metrics to measure the feature subset's impurity, for instance, C4.5 [34] uses information gain, CART [35] uses Gini Index and CHAID [36] uses Chi-squared test.

Each decision tree model was seen as a set of "if-then-else" decision rules, which makes it simple to understand and interpret. In addition, it is able to handle both numeric and categorical data. DT is also robust to noise and scale well with large datasets. However, due to having only one feature in each inner node, DT usually does not perform well when there are complex interactions between features [34].

5.3 Random Forest

Random forest is an ensemble based machine learning method for both classification and regression tasks. In DT algorithms, when a tree grows too deep, it tends to overfit the training set. In this case, the tree simply models the noise in the training set rather than represents the relationship between inputs (features) and output (class label). Breiman [37] proposed a new classification algorithm, called random forest, which averages multiple decision trees to reduce the variance. Each random forest contains a set of decision trees, which are learned from different parts of the training set. In particular, each decision tree is constructed by randomly selecting instances with replacements from the original training set. In addition, the tree learning algorithm is also modified by using a random subset of features at each inner node instead of the original feature set. This process is also known as "feature bagging". After a number of decision trees are learned by using their own training sets, they are combined to form a random forest classifier. The classifier is then classified unseen instances by applying a voting scheme. Random forest avoids the usual overfitting problem of decision trees by applying a general technique of bootstrap aggregating or bagging to the decision tree learners.

5.4 Support Vector Machines (SVMs)

In 1963, SVMs were originally proposed by Vapnik to solve binary classification problems. Recently, SVMs have been extended to adapt to multi-class problems [38]. The main goal of SVMs is to build or construct one or more hyperplanes to split a given dataset into multiple subsets corresponding to different class labels. There might be many hyper planes that can split the data but the selected hyper plane should maximize the distances with its nearest training instances.

In many practical problems, SVMs achieve good performance [39]. However, users usually have to provide a good kernel function for SVMs for non-linear cases. SVM is able to handle sparsity of the data representation by using right Kernel functions or regularization methods. The most popular kernels available are linear, polynomial, RBF (Radial Basis Function) and sigmoid. In terms of efficiency, SVMs have a high computational cost and require a large memory in the training phase when there is more number of dimensions [40].

5.5 Naïve Bayes Classifiers – NB

In Bayesian classifiers, a probabilistic model is learned which is then used to predict the class label of unseen data. Bayesian classifiers assume that the relationships between features and the class label was described in terms of probability distributions and the features are conditionally independent to the given class label [26]. Based on the training set, Bayesian classifiers induce the conditional probability distribution of each feature given the class label as well as the probability distribution of the class label. The two distributions are then used to calculate how likely an unseen instance belongs to each class [41]. One of the most common and straightforward Bayesian classifiers is Naive Bayes (NB). NB is an efficient classification algorithm in many real-world problems and its assumption is violated due to the complex interactions between features.

There are three different types of models for the Naïve Bayes Classifier such as Multivariate Bernoulli Naive Bayes, Multinomial Naive Bayes and Gaussian Naive Bayes [42]. The most commonly used for a document classification purpose is the Multivariate Bernoulli Naive Bayes and Multinomial Naive Bayes model [43]. The Multivariate Bernoulli model is approaching the domain with the bag of words distribution as it focuses on the presences and absences of features. The Multinomial model includes frequency in its feature distribution approach. McCallum, A. et al., suggests [43] that the Multinomial model performs better than the Bernoulli model in both academic types of research tests and in real-world application problems.

5.6 Logistic Regression (LR)

Logistic regression (LR) is a classification technique in machine learning that originates from the field of statistics [44]. It is a common method for binary classification problems as a result of its low computational cost. The goal of LR is similar to linear regression that is finding the weight of each input (coefficient). The difference is in the transformation function, called the logistic function. This logistic function transforms the values to a range between 0 and 1and this predicts the class label based on the rules or probabilities. LR works better by removing the correlated attributes or those that are not related to the output.

6. BAG OF WORDS (BOW) APPROACH

Traditionally, the Bag Of Words model popularly used by several researchers in different research domains for document vector representation. It is widely used in text classification area. Bag of words is a segment that treats every word as a feature. This is a way of modelling data with machine learning. BOW model ignores the ordering of the words, semantics of words and relationship among words while representing the document vectors [45]. The Fig. 1 shows the BOW model. In this model, first pre-processing techniques like stopword removal and stemming are applied on the dataset to remove unwanted information. Extract the terms based on their frequency in the total dataset and considered top frequent terms as bag of words. The extracted bag of word features are used to represent the documents as a distribution of vectors. By adding representational weighting to the features, the words in the documents give an actual meaning of the context. Term weight measures are used to compute the weight of a word in a vector presentation. In general, this technique used the term frequency of the words in the documents as a weighting scheme in the vector representation. In this work, TFIDF measure is used to calculate the term weight in the document representation. In this measure, less important words which are used more often in our everyday language will lose importance and more

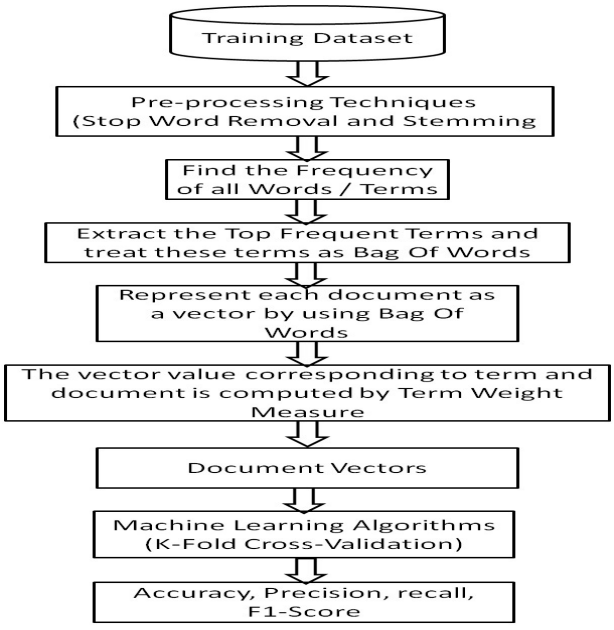

Figure 1. The Steps In BOW Model

specific words which define the topic and document itself will get a higher score. The document vectors are passed to the machine learning algorithms to produce the model. This model used the k-fold cross validation technique to evaluate the proposed approach performance in the form of precision, recall, accuracy and F1-score. A document classifier is explained as a function that maps an input attribute vector of words from the document to predict a class that correlates to the input. The classifier needs training to learn from the labelled input to determine the class.

Different pre-processing techniques like stop words removal and stemming are used to clean unnecessary words or characters [46]. Stopword removal is interpreted as a straightforward process as it is not algorithmic, compared to stemming, which is implemented with an algorithmic approach. Stopwords refers to common or short function words that are not relevant in the text data regarding the classification of the differences of the documents. The inclusion of stop words gives faulty predictions because the stopwords usually have a higher occurrence and affect the document representation. Stemming refers to the processing of truncate the words to its root stem, which enables it to map the words from the same root stem. Stemming was firstly introduced by Lovins (1968) [47] and significant advancements was achieved further in stemming algorithms. Porter (1997) developed [48] a stemmer which contributed to a more aggressive stemming algorithm which creates more classes. In this work, porter stemmer is used for stemming.

7. FEATURE SELECTION TECHNIQUES BASED APPROACH FOR TEXT CLASSIFICATION

In this work, feature selection technique based approach is proposed for text classification. The fig. 2 shows the model for proposed approach. In this approach, firstly preprocessing techniques like stopword elimination and stemming process are applied on training dataset to clean irrelevant data. After cleaning the dataset, extract all terms. The feature selection techniques are used to compute the relevancy scores of term pertaining to the class. Pearson Correlation Coefficient is used to compute the redundancy among features and this redundancy information is removed from the relevancy score of a term. Identify the minimum number of terms or features for document representation based on the updated relevancy scores. The documents are represented as vectors by using identified features. The TFIDF measure is used to determine the vector value in the document vector representation. The document vectors are used to train the machine learning algorithms that generates learning model. This model predicts the performance of the proposed approach.

In general, each feature selection algorithm contains four important steps such as initialization, Subset generation, Subset evaluation and stopping criterion [49]. Initialization step initializes feature selection algorithm with original set of features. Subset generation generates one or more candidate feature subsets by using a searching mechanism. The starting point of the searching mechanism can be any feature subsets which can contain none of the original features, all of the original features or some randomly selected original features. Subset evaluation step measures the goodness of each generated feature subset by using an evaluation criterion also known as fitness function. The fitness

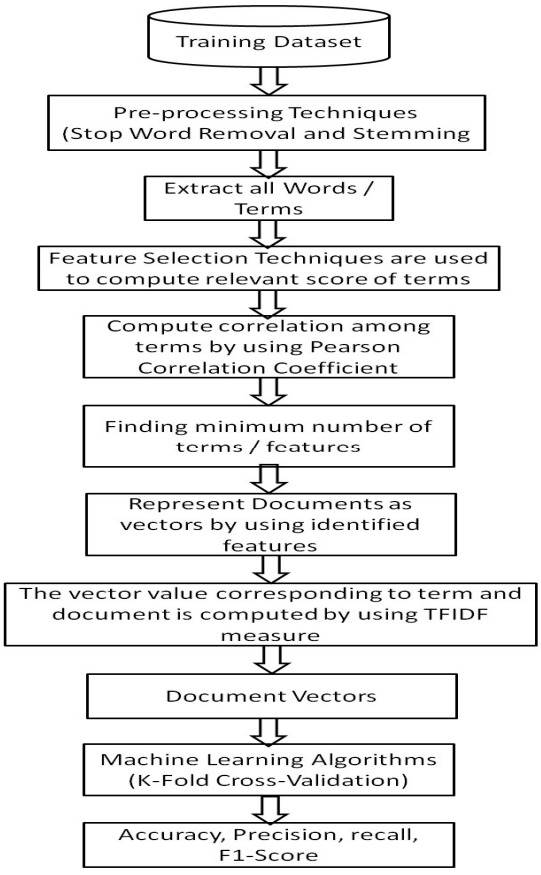

Figure 2. The Feature Selection Technique Based Approach For Text Classification

function guides the searching mechanism to explore promising regions in the search space. The stopping criterion is divided into two types based on evaluation function and generation procedure. In evaluation function stopping criteria, the feature selection process stopped when the best fitness value is not improved for a finite number of steps. In generation procedure based criteria, the feature selection process stopped after maximum number of iterations are reached or predefined number of features are selected.

Feature reduction is an important pre-processing technique for classification. The aim of feature reduction is diminishing the dimensionality of a dataset while maintaining or improving performance of classification over using all features. The feature reduction methods are divided into two techniques such as feature selection and feature extraction. The feature extraction identifies a less number of new features by transforming or combining the original features. The feature selection selects the most important features. The feature selection algorithms are classified as three classes such as filter, wrapper and embedded.

7.1 Types of Feature Selection Algorithms

The feature selection methods are divided into three classes such as filter, wrapper and embedded based on the evaluation criterion [50].

7.1.1 Filter Methods

Filter methods used data characteristics to evaluate feature subsets, which do not involve any classification algorithm. Compared with wrappers, it is more difficult for filters to consider multi-way feature interactions because the feature subset is evaluated in an independent way of any learning algorithm. Filters provide more general solutions than wrappers [51]. In addition, filters are also less computationally intensive than wrappers. However, since these methods do not take into account the interaction between the selected feature subset and the learning algorithm, they usually achieve lower classification accuracies than wrappers [52].

7.1.2 Wrapper Methods

Wrappers used classification algorithm to calculate fitness values of feature subsets. Wrapper methods usually attain better accuracies for classification than filters. Wrappers generate feature subsets with poorer generality to other classification algorithms than filters. In addition, during the searching process, a classification algorithm is repeatedly trained to evaluate feature subsets, which usually causes a long computational time. Therefore, wrapper methods are usually more expensive than other methods.

In comparison with filter-based feature selection approaches, the wrapper based approaches generally attain better classification performance since the evaluation process explicitly considers the interactions among chosen feature subset and the classification algorithm. However, the effectiveness of wrappers comes along with their expensive computational cost since it involves a learning process during each evaluation step.

7.1.3 Embedded Methods

In embedded approaches, the selection process is a part of the training procedure of a classification algorithm. Embedded methods consider the interactions among feature subset and the classification algorithm. However, embedded methods are only applicable to some specific algorithms such as DT and SVM.

In comparison with wrappers, embedded approaches are more efficient and may still maintain a good classification performance. Decision Tree is used as a feature selection technique in embedded methods. Particularly, features used in the final tree are considered as a good feature set. SVM also be used as an embedded-based feature selection method, where each weight in the learned SVM model is considered as the importance of the corresponding feature.

7.2 Existing Feature Selection Algorithms

In this work, four filter based feature selection techniques such as chi square, information gain, mutual information and relative discriminative criterion are used for determining the relevant features.

7.2.1 Information gain

Information Gain (IG) is feature selection technique in text classification that measure the amount of information got for a given category by having a term in a document or absent the term [53]. IG feature selection chooses the terms which scores higher gain information. The Information Gain of a term T is computed by using equation (5).

$$IG(T) = -\sum_{j=1}^{m} P(C_j)\log\left(P\left(C_j\right)\right) + P(T)\sum_{j=1}^{m} P\left(C_j/T\right)\log\left(P\left(C_j/T\right)\right) +$$

$$P(\overline{T}) \sum_{j=1}^{m} P(C_j/\overline{T}) \log(P(C_j/\overline{T}) \quad (5)$$

Where, m is classes count, $P(C_j)$ is the proportion of documents count in class C_j relative to total documents count in training dataset. P(T) and $P(\overline{T})$ are the proportion of documents contain term T and doesn't contain term T in whole dataset respectively. $P(C_j|T)$ and $P(C_j|\overline{T})$ are the proportion of class C_j documents contain term T and doesn't contain term T respectively.

7.2.2 Chi-square

Chi square measure determines the dependency score among the class and a feature [49]. The high score indicates the feature is more relevant to the given class and low score indicates the feature is less informative to the class. The equation (6) designates the CHI2 measure for term t in a specific class C_j.

$$\chi^2(t, C_j) = \frac{N(AD-BC)^2}{(A+B)(A+C)(B+D)(C+D)} \quad (6)$$

In this measure, A is number of class C_j documents contains term t, B is the number of other than class C_j documents contain the term t, C is the number of class C_j documents doesn't contain term t and D is the number of other than class C_j documents doesn't contain the term t, N is the total documents count. The equation (7) is used to compute the CHI2 of a term across all classes.

$$\chi^2(t) = \sum_{i=1}^{m} P(C_i)\, \chi^2(t, C_i) \quad (7)$$

Where, m is number of classes, $P(C_i)$ is the proportion of documents in class C_i relative to the total number of documents in the training dataset.

7.2.3 Mutual information

Mutual information (MI) determines the relation among the features and the classes [49]. MI measures the mutual dependence of a feature T_i and a category C_j. The MI among the term T_i and class C_j is computed by using equation (8).

$$MI(T_i, C_j) = \log\left(\frac{P(T_i/C_j)}{P(T_i)}\right) \quad (8)$$

Where, $P(T_i|C_j)$ is the proportion of class C_j documents contain the term T_i, $P(T_i)$ is proportion of documents in all classes contain the term T_i. The MI score of a term T in all classes is computed by using equation (9).

$$MI(T) = \max_{j=1}^{m}\left(MI(T_i, C_j)\right) \quad 9)$$

Where, m is the classes count.

7.2.4 Relative Discriminative Criterion (RDC)

The RDC measure consider the differences among the number of positive and negative class of documents contains the term w_i [17]. The equation (10) is used to determine the score of a feature w_i by using RDC feature selection technique.

$$RDC\left(w_i, tc_j(w_i)\right) = \left(\frac{|df_{pos}(w_i) - df_{neg}(w_i)|}{\min(df_{pos}(w_i), df_{neg}(w_i)) \times tc_j(w_i)}\right) \quad (10)$$

Where, $df_{neg}(w_i)$ and $df_{pos}(w_i)$ are the number of negative and positive class of documents contain feature w_i at least one time respectively. The feature w_i may repeat many times in a specific document. The $tc_j(w_i)$ is the occurrence count of feature w_i in class c_j documents. It was recorded as array $tc_j(w_i) = [tc_1(w_i), tc_2(w_i), \ldots, tc_m(w_i)]$.

7.3 Feature Selection Technique Based On Category Specific Feature Distribution Without Redundancy Information (CFDRI)

The main aim of feature selection is selecting the maximal discriminative capability features and compact feature subset. In text classification process, thousands of features are involved and the classification becomes a high dimensional problem. Most of the features in the feature space not having little or no discriminative power to predict the class label of a text document. The feature relevancy indicates that the feature is always necessary in the process of class label prediction. Feature redundancy is defined as a type of correlation among features. The feature selection step not only improves the classification performance but also reduce the storage requirements.

The proposed feature selection technique named as Category specific Feature Distribution without Redundancy Information (CFDRI) which considers the information of term distribution in positive class of documents and negative class of documents, documents of positive and negative class contains the term, the occurrence of term in positive and negative class of documents to compute the relevancy of a term to a specific class. Equation (11)

is used to compute the CFDRI measure of a term in positive class.

$$\mathrm{CFDRI}(T_i, C_{pos}) = TF_i \times \log\left(\frac{P(C_{pos})}{P(C_{pos}/T_i)}\right) \times \frac{|C|}{CF_i} \times \left(\frac{P(C_{pos}/T_i)}{P(C_{pos}/\overline{T_i})}\right) \times \frac{P(T_{i,pos})}{P(T_{i,neg})} - \frac{\sum_{T_i \neq T_j, \forall T_j \in BOW} \mathrm{corr}(T_i, T_j)}{f} \quad (11)$$

Where, TF_i is the term occurrence count of T_i in total dataset, $P(C_{pos})$ is proportion of documents in positive class, $P(C_{pos}|T_i)$ and $P(C_{neg}|T_i)$ are the proportion of positive and negative class of documents contains term T_i respectively, $P(C_{pos}|\overline{T_i})$ and $P(C_{neg}|\overline{T_i})$ are the proportion of positive and negative class of documents doesn't contains term T_i respectively, $|C|$ is total classes count, CF_i is number of classes contain term T_i, $P(T_{i,pos})$ and $P(T_{i,pos})$ are the proportion of term T_i frequency in positive and negative class of documents respectively. $\mathrm{corr}(T_i, T_j)$ is the correlation among terms T_i and T_j, m is the number of classes in dataset. $\mathrm{CFDRI}(T_i, C_{pos})$ computes the term relevant score pertaining to positive class C_{pos}, $\mathrm{CFDRI}(T_i, C_{neg})$ computes the term relevant score pertaining to negative class C_{neg}. Equation (12) is used to compute the score of a term in whole dataset.

$$\mathrm{CFDRI}(T) = \max_{k=1 \text{ to } m}(\mathrm{CFDRI}(T, C_k)) \quad (12)$$

After computing the term relevant score pertaining to the class, the next step is selecting the features. The selected features have high relevance to the classes and low correlation with other features. The terms which are having good amount of score are considered as high relevant features. The correlation among selected term and all other terms are calculated by using correlation measure. In this work, Pearson Correlation Coefficient (PCC) measure is used to calculate the correlation between two numerical features.

The approximation of the PCC for random samples $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ is determined by using equation (13).

$$\mathrm{PCC}(x, y) = \frac{n\sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n\sum x_i^2 - (\sum x_i)^2}\sqrt{n\sum y_i^2 - (\sum y_i)^2}} \quad (13)$$

Where, PCC(x, y) is PCC between x and y, n is number of observations, x_i is the value of x in i^{th} observation, y_i is the value of y in i^{th} observation.

The PCC computes the ratio between how much *X* and *Y* vary with each other, and how much each of the two variables varies themself. If the absolute value of the PCC is close to 1, the variables are strongly correlated. If the PCC is close to zero, they are uncorrelated.

7.4 TF-IDF

TF-IDF is popularly used in different applications like Natural Language Processing and Information retrieval for determining the significance of a term. It is a term weight measure which uses statistical information to compute the term importance specific to a document. According to the Term Frequency (TF) measure, the terms which are occurred frequently in all documents are having more importance with respect to less frequent terms.

The Inverse Document Frequency (IDF) inversely proportional to the document frequency. IDF measure assigns less importance to the terms which are frequently occurred in all documents. It gave more weight to the terms which are occurred less in documents. Equation (14) shows the IDF measure.

$$\mathrm{IDF}(T_i) = \mathrm{LOG}\left(\frac{|D|}{1 + DF_i}\right) \quad (14)$$

Where, $|D|$ is total count of documents in dataset D, DF_i is the count of documents in dataset D contains term T_i at least one time.

The TFIDF of term T_i in document D_k is computed by using equation (15).

$$\mathrm{TFIDF}(T_i, D_k) = \mathrm{TF}(T_i, D_k) \times \mathrm{IDF}(T_i) \quad (15)$$

Where, $\mathrm{TF}(T_i, D_k)$ is the occurrence count of term T_i in document D_k.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

8. EXPERIMENTAL RESULTS OF FEATURE SELECTION TECHNIQUES

In this experiment, the feature selection technique based approach is proposed for text classification. In this approach, various feature selection techniques are used to recognize the most relevant features to the class based on the scores of the terms. Top scored 8000 terms are used in this experiment to represent the document vectors. The experiment started with 2000 terms and increased by 2000 in every iteration. The experiment conducted with five feature selection techniques such as MI, IG, CHI2, RDC and CFDRI. Six machine learning algorithms are used to train the learning model as well as assessing the performance of the proposed approach. Six popular text classification datasets are used in this experiment for text classification.

The table 3 shows the accuracies of six machine learning algorithms when trained with six datasets by using the features selected by the mutual information technique. In table 3, it was observed that the RF classified attained good accuracies when compared with other classifiers. For IMDB dataset, the SVM classifier obtained 0.831% accuracy when the document vectors are represented with top 8000 terms. The RF classifier achieved an accuracy of 0.750% when trained with top 4000 features on HSS dataset. The RF classifier achieved accuracies of 0.855%, 0.740% and 0.834% for AGN, CBN and 20NG datasets respectively when experimented with 6000 features. For FN dataset, the RF classifier obtained 0.727% accuracy when the document vectors are represented with top 8000 terms.

The Table 4 Shows The Accuracies Of Six Machine Learning Algorithms When Trained With Six Datasets By Using The Features Selected By The Information Gain Technique. In Table 4, the RF classifier achieved good accuracies for text classification. The RF classifier obtained accuracies of 0.839, 0.859 and 0.762 for IMDB, AGN and HSS respectively when documents are represented as 6000 terms. For CBN, FN and 20NG, the RF classifier attained accuracies of 0.767, 0.742 and 0.841 respectively when experimented with 8000 terms.

The table 5 shows the accuracies of six machine learning algorithms when trained with six datasets by using the features selected by the Chi square feature selection technique. In table 5, the RF classifier achieved accuracies of 0.869, 0.767, 0.778 and 0.867 for AGN, HSS, CBN and 20NG datasets respectively when top scored 6000 terms are used in experiment. For IMDB and FN, the RF classifier attained accuracies of 0.849 and 0.767 respectively when 8000 terms are used to represent the document.

The table 6 shows the accuracies of six machine learning algorithms when trained with six datasets by using the features selected by the Relative Discriminative Criterion feature selection technique. In table 6, the RF classifier achieved good accuracies when compared with other classifiers. The RF classifier obtained accuracies of 0.858, 0.789, 0.799, 0.794 and 0.901 for IMDB, HSS, CBN, FN and 20NG datasets respectively when experimented with 8000 terms. For AGN dataset, 0.887 accuracy is achieved by RF classifier when 6000 terms used in experiment.

The table 7 shows the accuracies of six machine learning algorithms when trained with six datasets by using the features selected by the proposed CFDRI feature selection technique. In table 7, RF classifier got 0.872 and 0.894 accuracies for IMDB and AGN datasets respectively when 6000 terms are used in the vector representation. The top scored 8000 terms and RF classifiers obtained good accuracies of 0.837, 0.814, 0.831 and 0.917 for HSS, CBN, FN and 20NG datasets respectively

Table 3. The Accuracies Of Text Classification When Experimented With Features Selected By Mutual Information

Machine Learning Algorithms / Features Selected by Feature Selection Techniques - Datasets		NBM	SVM	DT	RF	KNN	LR
2000	IMDB	0.731	0.793	0.653	0.821	0.606	0.815
	AGN	0.732	0.809	0.792	0.844	0.667	0.805
	HSS	0.687	0.725	0.600	0.713	0.676	0.725
	CBN	0.612	0.696	0.596	0.721	0.585	0.703
	FN	0.634	0.717	0.600	0.705	0.602	0.697
	20NG	0.772	0.807	0.757	0.819	0.713	0.803
4000	IMDB	0.691	0.812	0.656	0.819	0.574	0.808
	AGN	0.772	0.794	0.779	0.842	0.666	0.814
	HSS	0.612	0.737	0.587	**0.750**	0.663	0.745
	CBN	0.624	0.712	0.607	0.735	0.599	0.714
	FN	0.650	0.720	0.614	0.716	0.611	0.703
	20NG	0.794	0.815	0.756	0.825	0.720	0.814
6000	IMDB	0.637	0.815	0.667	0.823	0.589	0.812
	AGN	0.775	0.795	0.775	**0.855**	0.659	0.811
	HSS	0.650	0.722	0.625	0.725	0.687	0.732
	CBN	0.635	0.723	0.619	**0.740**	0.607	0.722
	FN	0.667	0.728	0.623	0.724	0.619	0.712
	20NG	0.787	0.819	0.762	**0.834**	0.729	0.821
8000	IMDB	0.635	**0.831**	0.656	0.829	0.568	0.826
	AGN	0.803	0.786	0.789	0.851	0.645	0.809
	HSS	0.612	0.712	0.675	0.732	0.650	0.739
	CBN	0.649	0.729	0.624	0.738	0.613	0.729
	FN	0.634	0.723	0.617	**0.727**	0.624	0.719
	20NG	0.791	0.823	0.769	0.828	0.722	0.813

Table 4: The Accuracies Of Text Classification When Experimented With Features Selected By Information Gain

Machine Learning Algorithms / Features Selected by Feature Selection Techniques - Datasets		NBM	SVM	DT	RF	KNN	LR
2000	IMDB	0.731	0.793	0.656	0.829	0.606	0.805
	AGN	0.733	0.814	0.796	0.847	0.679	0.806
	HSS	0.689	0.735	0.625	0.742	0.687	0.730
	CBN	0.634	0.705	0.613	0.723	0.602	0.712
	FN	0.650	0.725	0.627	0.714	0.622	0.711
	20NG	0.791	0.811	0.766	0.821	0.726	0.810
4000	IMDB	0.698	0.816	0.672	0.836	0.577	0.817
	AGN	0.778	0.797	0.783	0.856	0.684	0.819
	HSS	0.617	0.747	0.650	0.753	0.662	0.755
	CBN	0.646	0.717	0.628	0.743	0.615	0.737
	FN	0.667	0.734	0.633	0.729	0.637	0.720
	20NG	0.806	0.826	0.774	0.829	0.734	0.817
6000	IMDB	0.639	0.818	0.661	**0.839**	0.593	0.829
	AGN	0.781	0.799	0.779	**0.859**	0.677	0.813
	HSS	0.658	0.742	0.700	**0.762**	0.689	0.742
	CBN	0.659	0.726	0.635	0.755	0.624	0.740
	FN	0.684	0.744	0.648	0.736	0.641	0.726

	20NG	0.815	0.829	0.769	0.836	0.745	0.826
8000	**IMDB**	0.645	0.835	0.660	0.829	0.574	0.818
	AGN	0.808	0.802	0.785	0.852	0.663	0.815
	HSS	0.637	0.732	0.687	0.757	0.667	0.747
	CBN	0.671	0.720	0.640	**0.767**	0.629	0.751
	FN	0.671	0.739	0.639	**0.742**	0.646	0.734
	20NG	0.812	0.835	0.777	**0.841**	0.739	0.831

Table 5: The Accuracies Of Text Classification When Experimented With Features Selected By Chi Square

Machine Learning Algorithms / Features Selected by Feature Selection Techniques - Datasets		**NBM**	**SVM**	**DT**	**RF**	**KNN**	**LR**
2000	**IMDB**	0.739	0.799	0.665	0.838	0.579	0.814
	AGN	0.739	0.816	0.809	0.855	0.690	0.829
	HSS	0.698	0.752	0.682	0.750	0.675	0.751
	CBN	0.662	0.715	0.643	0.745	0.626	0.728
	FN	0.677	0.736	0.645	0.734	0.639	0.727
	20NG	0.804	0.824	0.773	0.834	0.740	0.824
4000	**IMDB**	0.691	0.818	0.668	0.830	0.592	0.823
	AGN	0.788	0.803	0.801	0.864	0.697	0.835
	HSS	0.701	0.757	0.697	0.769	0.695	0.768
	CBN	0.675	0.728	0.651	0.767	0.635	0.737
	FN	0.684	0.748	0.654	0.753	0.647	0.738
	20NG	0.819	0.841	0.781	0.854	0.751	0.835
6000	**IMDB**	0.629	0.819	0.679	0.841	0.574	0.837
	AGN	0.797	0.807	0.803	**0.869**	0.704	0.841
	HSS	0.692	0.749	0.702	**0.767**	0.702	0.751
	CBN	0.690	0.737	0.668	**0.778**	0.642	0.745
	FN	0.691	0.754	0.667	0.759	0.652	0.741
	20NG	0.814	0.838	0.785	**0.867**	0.753	0.846
8000	**IMDB**	0.618	0.845	0.668	**0.849**	0.614	0.846
	AGN	0.811	0.815	0.818	0.858	0.693	0.832
	HSS	0.697	0.750	0.713	0.755	0.708	0.769
	CBN	0.689	0.743	0.672	0.773	0.649	0.756
	FN	0.687	0.751	0.672	**0.767**	0.666	0.752
	20NG	0.818	0.847	0.791	0.863	0.760	0.853

Table 6: The Accuracies Of Text Classification When Experimented With Features Selected By Relative Discriminative Criterion

Machine Learning Algorithms / Features Selected by Feature Selection Techniques - Datasets		**NBM**	**SVM**	**DT**	**RF**	**KNN**	**LR**
2000	**IMDB**	0.743	0.804	0.673	0.836	0.608	0.821
	AGN	0.749	0.817	0.809	0.876	0.708	0.842
	HSS	0.687	0.775	0.705	0.771	0.701	0.772
	CBN	0.687	0.721	0.669	0.762	0.651	0.743
	FN	0.693	0.757	0.651	0.757	0.645	0.745
	20NG	0.812	0.842	0.787	0.876	0.759	0.854
4000	**IMDB**	0.692	0.819	0.685	0.839	0.619	0.838
	AGN	0.793	0.816	0.812	0.879	0.717	0.848

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

	HSS	0.625	0.784	0.717	0.786	0.712	0.750
	CBN	0.695	0.739	0.685	0.787	0.667	0.759
	FN	0.705	0.769	0.662	0.772	0.651	0.757
	20NG	0.827	0.856	0.798	0.881	0.772	0.862
6000	**IMDB**	0.621	0.854	0.677	0.855	0.628	0.845
	AGN	0.806	0.826	0.828	**0.887**	0.711	0.856
	HSS	0.612	0.789	0.726	0.778	0.725	0.787
	CBN	0.702	0.748	0.692	0.794	0.675	0.768
	FN	0.714	0.786	0.677	0.785	0.660	0.763
	20NG	0.825	0.867	0.806	0.898	0.781	0.875
8000	**IMDB**	0.627	0.856	0.676	**0.858**	0.623	0.849
	AGN	0.829	0.829	0.816	0.872	0.705	0.851
	HSS	0.612	0.802	0.712	**0.789**	0.718	0.800
	CBN	0.712	0.754	0.689	**0.799**	0.682	0.774
	FN	0.708	0.775	0.689	**0.794**	0.673	0.772
	20NG	0.839	0.873	0.801	**0.901**	0.799	0.886

Table 7: The Accuracies Of Text Classification When Experimented With Features Selected By Proposed CFDRI

Machine Learning Algorithms / Features Selected by Feature Selection Techniques - Datasets		**NBM**	**SVM**	**DT**	**RF**	**KNN**	**LR**
2000	**IMDB**	0.767	0.812	0.691	0.859	0.640	0.832
	AGN	0.789	0.828	0.841	0.881	0.739	0.866
	HSS	0.700	0.800	0.742	0.806	0.737	0.802
	CBN	0.709	0.745	0.693	0.787	0.680	0.765
	FN	0.716	0.784	0.687	0.797	0.668	0.778
	20NG	0.831	0.862	0.807	0.886	0.802	0.872
4000	**IMDB**	0.715	0.830	0.702	0.863	0.669	0.840
	AGN	0.814	0.833	0.833	0.885	0.746	0.873
	HSS	0.717	0.817	0.750	0.829	0.725	0.797
	CBN	0.715	0.761	0.704	0.792	0.687	0.771
	FN	0.729	0.804	0.704	0.812	0.676	0.789
	20NG	0.845	0.877	0.814	0.893	0.819	0.880
6000	**IMDB**	0.658	0.861	0.724	**0.872**	0.681	0.854
	AGN	0.832	0.837	0.849	**0.894**	0.758	0.872
	HSS	0.722	0.822	0.765	0.815	0.765	0.812
	CBN	0.724	0.773	0.712	0.808	0.699	0.785
	FN	0.734	0.817	0.716	0.823	0.690	0.805
	20NG	0.849	0.884	0.825	0.908	0.827	0.891
8000	**IMDB**	0.683	0.865	0.711	0.869	0.667	0.851
	AGN	0.834	0.846	0.842	0.889	0.747	0.872
	HSS	0.712	0.825	0.750	**0.837**	0.762	0.827
	CBN	0.738	0.789	0.719	**0.814**	0.702	0.796
	FN	0.738	0.820	0.722	**0.831**	0.707	0.813
	20NG	0.832	0.893	0.819	**0.917**	0.822	0.899

Journal of Theoretical and Applied Information Technology
30th September 2021. Vol.99. No 18

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

9. DISCUSSION OF RESULTS

Fig. 3 shows the accuracies of IMDB dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 6000 terms achieved best accuracy of 0.872 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is enhanced when the number of terms is increased to represent the document vectors in most of the situations.

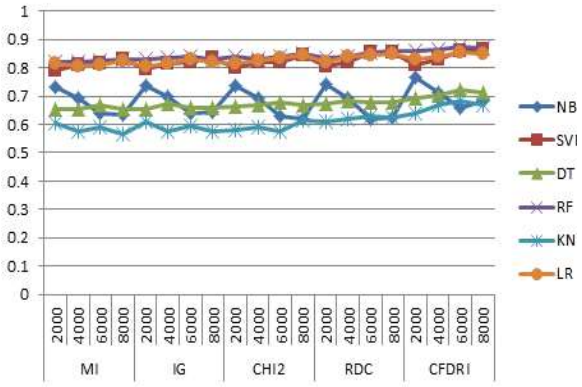

Figure 3. The Accuracies Of IMDB Dataset

Fig. 4 shows the accuracies of AGN dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 6000 terms achieved best accuracy of 0.894 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is improved when the number of terms is increased to represent the document vectors in most of the situations.

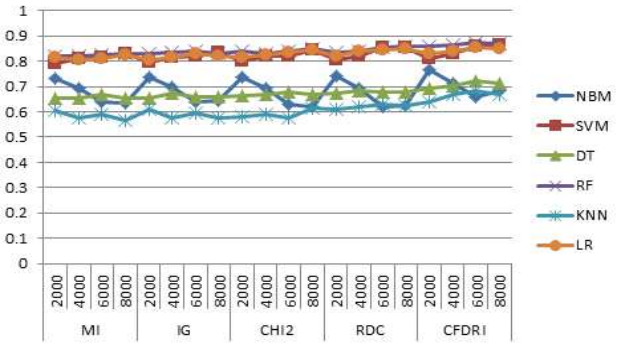

Figure 4. The Accuracies Of AGN Dataset

Fig. 5 shows the accuracies of HSS dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 8000 terms achieved best accuracy of 0.837 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is improved when the number of terms is increased to represent the document vectors in most of the situations.

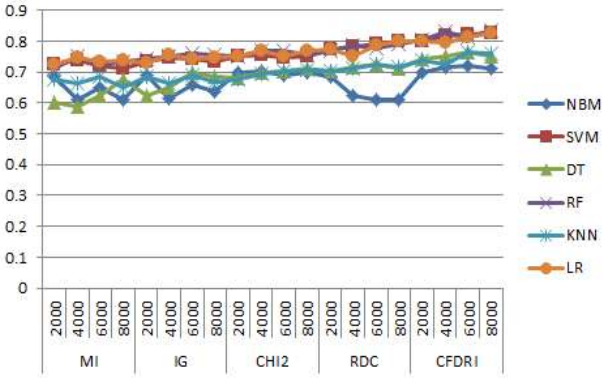

Figure 5. The Accuracies Of HSS Dataset

Fig. 6 shows the accuracies of CBN dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 8000 terms achieved best accuracy of 0.814 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is improved when

the number of terms is increased to represent the document vectors in most of the situations.

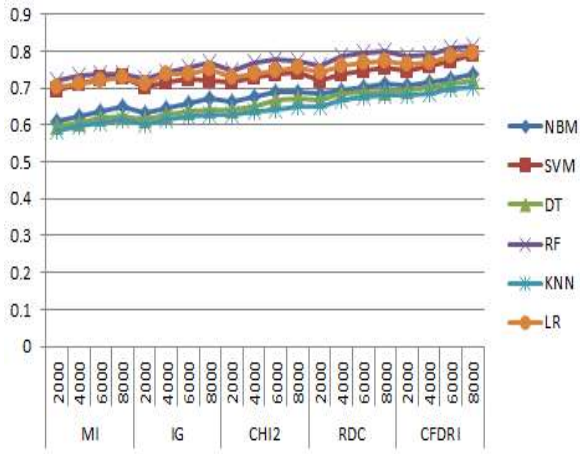

Figure 6. The Accuracies Of CBN Dataset

Fig. 7 shows the accuracies of FN dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 8000 terms achieved best accuracy of 0.831 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is improved when the number of terms is increased to represent the document vectors in most of the situations.

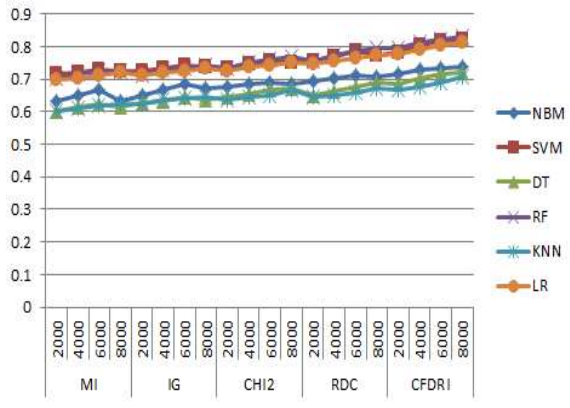

Figure 7. The Accuracies Of FN Dataset

Fig. 8 shows the accuracies of 20NG dataset when experimented with different features selected by the feature selection techniques and different machine learning algorithms. The combination of proposed CFRDI feature selection technique and RF classifier with top ranked 8000 terms achieved best accuracy of 0.917 for text classification when compared with other feature selection techniques and other machine learning algorithms. It was observed that the accuracy is improved when the number of terms is increased to represent the document vectors in most of the situations.

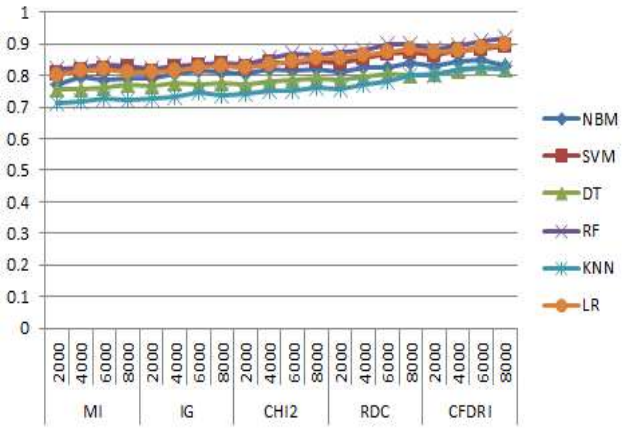

Figure 8. The Accuracies Of 20NG Dataset

10. CONCLUSIONS

Feature extraction and feature selection are two main steps for reducing the dimensionality of feature space. Feature extraction creates new feature set from original features by combining features. Feature selection select features based on the score of terms. Feature selection techniques are divided into three classes like filter, wrapper and embedded. Filter and embedded methods use more general information about the data set, while wrapper methods are methods that directly take into account the performance of a classification algorithm in the feature selection procedure. The embedded method is similar to the wrapper method which also includes the feature selection process in the learning algorithm. It uses machine learning methods for selection of the features. In this work, feature selection technique based text classification approach is proposed wherein a new feature selection technique was proposed to extract best relevant features. The experiment is conducted with 6 machine learning algorithms such as LR, SVM, KNN, NB, DT and RF to evaluate the performance of the proposed feature selection technique based approach for text classification. Six popular text

classification datasets such as HPS, FN, IMDB, 20NG, AGN and CBN are used in this work for text classification. For IMDB dataset, the RF classifier with top ranked 6000 terms achieved best accuracy of 0.872 for text classification. For AGN dataset, RF classifier with top ranked 6000 terms achieved best accuracy of 0.894 for text classification. For HSS dataset, RF classifier with top ranked 8000 terms achieved best accuracy of 0.837 for text classification. For CBN dataset, RF classifier with top ranked 8000 terms achieved best accuracy of 0.814 for text classification. For FN dataset, RF classifier with top ranked 8000 terms achieved best accuracy of 0.831 for text classification. For 20NG dataset, RF classifier with top ranked 8000 terms achieved best accuracy of 0.917 for text classification. The RF classifier achieved best results on all datasets when compared with other classification algorithms such as LR, SVM, KNN, NB and DT. The CFDRI feature selection technique attained best accuracies for text classification on all datasets compared with other feature selection techniques like IG, CHI2, MI and RDC.

It was observed from the results that the feature selection techniques obtained good accuracies for text classification. In future work, it was planned to implement different term weight measures to specify the importance of a term in the vector representation. It was also planned to implement an alternative document vector representation to avoid the problems with BOW model.

REFERENCES

[1] H. Zhao, A. P. Sinha, and W. Ge, "Effects of feature construction on classification performance: An empirical study in bank failure prediction," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2633–2644, 2009.

[2] Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. Communications of the *ACM*, 18(11), 613-620.

[3] Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1), 1-47.

[4] J. Yan et al., "Effective and Efficient Dimensionality Reduction for Large-Scale and Streaming Data Preprocessing", *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 3, pp. 320-333, March, 2006.

[5] L. Zhao, G. Zhuang and X. Xu, "Facial Expression Recognition Based on PCA and NMF", *In Proc. Proceedings of the 7th World Congress on Intelligent Control and Automation*, 2007.

[6] G. Chandrashekar and F. Sahin, "A survey on feature selection methods", *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16-28, January, 2014.

[7] Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. *WSEAS transactions on computers*, 4(8), 966-974.

[8] Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. bioinformatics, 23(19), 2507-2517.

[9] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, p. 94, 2017.

[10] Souad Larabi Marie-Sainte, Nada Alalyani, "Firefly Algorithm based Feature Selection for Arabic Text Classification", *Journal of King Saud University – Computer and Information Sciences*, 32 (2020), 320–328.

[11] Gang Kou, Pei Yang, Yi Peng, Feng Xiao, Yang Chen, Fawaz E. Alsaadi, "Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods", *Applied Soft Computing Journal*, 86 (2020), 105836, 1-14.

[12] Hongbin Dong, Jing Sun, Xiaohang Sun, Rui Ding, "A many-objective feature selection for multi-label classification", *Knowledge-Based Systems*, 208 (2020), 106456, 1 – 20

[13] Tunchan Cura , "Use of support vector machines with a parallel local search algorithm for data classification and feature selection", *Expert Systems With Applications* (2019).

[14] Jinfu Chen, Patrick Kwaku Kudjo, Solomon Mensah, Selasie Aformaley Brown, George Akorfu, "An automatic software vulnerability classification framework using term frequency-inverse gravity moment and feature selection", *The Journal of Systems and Software* 167 (2020) 110616, 1 – 20.

[15] Rasim Cekik, Alper Kursat Uysal, "A novel filter feature selection method using rough set for short text data ", *Expert Systems with Applications* 160 (2020) 113691, 1 – 15.

[16] Zohre Sadeghian, Ebrahim Akbari, Hossein Nematzadeh, "A hybrid feature selection method based on information theory and binary butterfly optimization algorithm", *Engineering Applications of Artificial Intelligence*, 97 (2021), 104079, 1 – 13.

[17] Mahdieh Labani, Parham Moradi, Fardin Ahmadizar, Mahdi Jalili, "A novel multivariate filter method for feature selection in text classification problems", *Engineering Applications of Artificial Intelligence*, 70 (2018), 25–37.
[18] Mahdieh Labani, Parham Moradi, Mahdi Jalili, "A multi-objective genetic algorithm for text feature selection using the relative discriminative criterion", *Expert Systems With Applications* 149 (2020), 113276, 1 – 21.
[19] Durga Prasad Kavadi, Palacharla Ravikumar, Dr.Kamini SrinivasaRao, "A New Supervised Term Weight Measure for Text Classification", *International Journal of Advanced Science and Technology*, 29(06), pp. 3115 - 3128, 2020.
[20] https://pan.webis.de/clef21/pan21-web/author-profiling.html
[21] https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset
[22] https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews
[23] K. Lang. (2008, January 2018). 20 Newsgroups. Available: http://qwone.com/~jason/20Newsgroups/
[24] ComeToMyHead. (2004, January 2018). AG's corpus of news articles. Available: https://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html
[25] https://www.kaggle.com/vikassingh1996/news-clickbait-dataset
[26] Y. Kodratoff, Introduction to machine learning. Morgan Kaufmann, 2014.
[27] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, Machine learning: An Artificial Intelligence Approach. *Springer Science & Business Media*, 2013.
[28] S. J. Russell and P. Norvig, Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,, 2016.
[29] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.
[30] N. M. Nasrabadi, "Pattern recognition and machine learning," *Journal of Electronic Imaging*, vol. 16, no. 4, p. 049901, 2007.
[31] L. Olshen, C. J. Stone, et al., "Classification and regression trees," *Wadsworth International Group*, vol. 93, no. 99, p. 101, 1984.
[32] T. Dietterich, "Overfitting and undercomputing in machine learning," *ACM Computing Surveys (CSUR),* vol. 27, no. 3, pp. 326–327, 1995.
[33] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.
[34] J. R. Quinlan, "Induction of decision trees," Machine Learning, vol. 1, no. 1, pp. 81–106, 1986.
[35] J. R. Quinlan, C4. 5: programs for machine learning. Elsevier, 2014.
[36] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, Classification and regression trees. CRC press, 1984.
[37] G. V. Kass, "An exploratory technique for investigating large quantities of categorical data," Applied Statistics, pp. 119–127, 1980.
[38] L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
[39] B. Schölkopf and C. J. Burges, Advances in kernel methods: support vector learning. MIT press, 1999.
[40] M. A. Hearst, S. T. Dumais, E. Osman, J. Platt, and B. Scholkopf, "Support vector machines," IEEE Intelligent Systems and their Applications, vol. 13, no. 4, pp. 18–28, 1998.
[41] J. Valyon and G. Horváth, "A weighted generalized ls-SVM," Periodica Polytechnica Electrical Engineering, vol. 47, no. 3-4, pp. 229–252, 2003.
[42] P. Langley,W. Iba, and K. Thompson, "An analysis of bayesian classifiers," in AAAI, vol. 90, pp. 223–228, 1992.
[43] Manning, C. D., Raghavan, P., & Schütze, H. (2008). Scoring, term weighting and the vector space model. Introduction to information retrieval, 100, 2-4.
[44] McCallum, A., & Nigam, K. (1998). A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization (Vol. 752, No. 1, pp. 41-48).
[45] Pranckevičius, T., & Marcinkevičius, V. (2017). Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification. *Baltic Journal of Modern Computing*, 5(2), 221
[46] Agarwal, B., & Mittal, N. (2014). Text classification using machine learning methods-a survey. *In Proceedings of the Second International Conference on Soft Computing for Problem Solving (SocProS 2012),* December 28-30, 2012 (pp. 701-709). Springer, New Delhi.
[47] Aggarwal, C. C., & Zhai, C. (Eds.). (2012). Mining text data. *Springer Science & Business Media*.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[48] Lovins, J. B. (1968). Development of a stemming algorithm. Mech. *Transl. Comput. Linguistics*, 11(1-2), 22-31.
[49] Porter, M. F. (1997). An algorithm for suffix stripping program. Editors JS Karen, and P. Willet, *Readings in Information Retrieval, San Francisco, Morgan Kaufmann*.
[50] M. Dash and H. Liu, "Feature selection for classification," *Intelligent Data Analysis*, vol. 1, no. 1, pp. 131–156, 1997.
[51] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *The Journal of Machine Learning Research*, vol. 3, pp. 1157–1182, 2003.
[52] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *Journal of Bioinformatics and Computational Biology*, vol. 3, no. 02, pp. 185–205, 2005.
[53] G. H. John, R. Kohavi, K. Pfleger, et al., "Irrelevant features and the subset selection problem," in Machine Learning: *Proceedings of the Eleventh International Conference*, pp. 121–129, 1994.
[54] M.H. Aghdam, N. Ghasem-Aghaee, M.E. Basiri, Text feature selection using ant colony optimization, *Expert Syst. Appl.* 36 (3) (2009) 6843–6853.