西 南 交 通 大 学 学 报

第 55 卷第 5 期
2020年 10 月

JOURNAL OF SOUTHWEST JIAOTONG UNIVERSITY

Vol. 55 No. 5
Oct. 2020

ISSN: 0258-2724 **DOI：10.35741/issn.0258-2724.55.5.1**

Research article

Computer and Information Science

A NOVEL FRAMEWORK FOR IDENTIFYING TWITTER SPAM DATA USING MACHINE LEARNING ALGORITHMS

使用機器學習算法識別推特垃圾郵件數據的新穎框架

Susana Boniphace Maziku[a,*], A. R. Rahiman [b], Abdullah Mohammed[c], Mohd Taufik Abdullah[d]

[a] Department of Communication Technology and Network, Faculty of Computer Science and Information Technology Universiti Putra Malaysia,43400 UPM Serdang, Selangor, Malaysia, maziku8@gmail.com.
[b]Department of Communication Technology and Network, Faculty of Computer Science and Information Technology, University Putra Malaysia, 43400 UPM Serdang, Malaysia, amir_r@upm.edu.my
[c] Department of Communication Technology and Networks, Faculty of Computer Science and Information Technology, UniversitiPutra Malaysia,43400 UPM Serdang, MALAYSIA, abdullah@upm.edu.my.
[d]Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia, taufik@upm.edu.my.

Received: June 6, 2020 ▪ Review: September 14, 2020 ▪ Accepted: October 15, 2020

Abstract

Nowadays, Twitter has become one of the most popular social media in the world. However, its popularity makes it an attractive platform for spammers to spread spam. Twitter spam becomes a severe issue. It is referred to as unsolicited tweets containing malicious links that direct victims to external sites containing malware downloads, terrorists, phishing, drug sales, scams, etc. Previous studies have approached spam detection as a classification problem, high dimension, time-consuming problem, which requires new methods to address the problems. This study introduces a novel framework for identifying Twitter spam data based on machine learning algorithms. By initializing data pre-processing for clean-up, noise removal, and unpredictable unfinished data, reducing the number of features in the tweet dataset using mutual information is the study's methods. The feature selection is introduced to select the most important from the extracted high-dimensional best features and feed the selected features into the minimum Redundancy and Maximal Relevance algorithm and apply random forest for classification. This study allows us to achieve higher classification accuracy and speed. The effectiveness evaluation being confirmed by experiment results show that accuracy is improved by 90% in 0hr 0m 20s time, compared with the existing system, the completion time is 2.022 seconds, and the accuracy is 80%. The research results contribute significantly to the field of cyber-security by forming a real-time system using machine learning algorithms.

Keywords: Natural Language Processing, Machine Learning, Twitter Spam, Feature Selection, minimum Redundancy and Maximal Relevance Algorithm

摘要 如今，Twitter 已成為世界上最受歡迎的社交媒體之一。但是，其受歡迎程度使其成為垃圾郵件發送者傳播垃圾郵件的誘人平台。 推特 垃圾郵件成為 一

個嚴重的問題。
它被稱為未經請求的推文，其中包含將受害者定向到包含惡意軟件下載，恐怖分子，網絡釣魚，藥品銷售，詐騙等的外部站點的惡意鏈接。以前的研究已將垃圾郵件檢測視為分類問題，高維度，耗時的問題，這需要新的方法來解決這些問題。這項研究介紹了一種基於機器學習算法識別Twitter垃圾郵件數據的新穎框架。通過初始化數據預處理以進行清理，消除噪聲和無法預測的未完成數據，使用互信息減少推文數據集中特徵的數量是本研究的方法。引入了特徵選擇，以從提取的高維最佳特徵中選擇最重要的特徵，並將選定的特徵輸入最小冗餘和最大相關性算法中，並應用隨機森林進行分類。這項研究使我們能夠實現更高的分類準確性和速度。實驗結果證實了有效性評價，結果表明，與現有系統相比，在 0小時0分鐘20秒 時間內精度提高了 90%，完成時間為 2.022 秒， 精度為 80%。研究結果通過使用機器學習算法形成實時系統，對網絡安全領域做出了重要貢獻。

关键词: 自然語言處理，機器學習，Twitter垃圾郵件，功能選擇，最小冗餘和最大相關性算法

I. INTRODUCTION

Twitter popularity has grown in recent years and has become an essential source of real-time information sharing and news dissemination. Inevitably, Twitter's growth is accompanied by a significant increase in Twitter spam, usually called unsolicited tweets with malicious links. The spam directs victims to external sites with malware downloads, phishing, drug sales, scams, etc. [1],[2],[3]. A sequence of incidents has shown that Twitter spam affects the user experience and poses threats of significant damage beyond social media platforms [4]. For instance, in September 2014, the nationwide Internet collapsed in New Zealand was triggered by a Twitter spam campaign, which spread the DDOS malware that leaked nude photos from Hollywood celebrities[5], [6].

The traditional approach focused on developing accurate models for identifying spam and spammers; however, fewer studies focused on identifying the most influencing factors in this identification. Though the methods such as blacklisting methods, labeling the data, a statistical method, syntax analysis, graph models based on features are difficult to collect data, consume memory space, and time. Although there is a continuous effort by researchers and practitioners to develop accurate spam detection systems, social media users still receive tremendous amounts of tweet spam every day[7].

Several approaches have been proposed for improving performance accuracy, including feature selection techniques, and assist in selecting relevant features [8]. Because real-world datasets are generally high-dimensional, feature selection techniques help reduce the dimensionality of data by eliminating inappropriate and unnecessary tweets features[9]. However, when inappropriate features are included in the classification process, it also consumes more memory and processing time[10].

The feature selection methods are being classified into three types: (i) Filter method, (ii) Wrapper method, and (iii) Embedded method. The filter approach first lists any classification algorithm's main features and uses variable ranking techniques to score variables[11]. The highest variables are selected from the pool of variables resulting in the removal of less relevant variables. Examples of filtering methods are t-test, information gain, chi-square, relief, mutual information (MI). Wrapper methods rely on comparing the quality of several classifiers built on different features[12]. Examples are Neighbor Rough Set (NRS), Distinct Sector Cardinality (DSC), and Neighborhood End System (NTS). Embedded method feature selection is performed by considering the classification algorithm[13], [14]. Therefore, the search for the optimal subset of features is built into the created classifier and can be seen as a search for merging attribute parts and theories. This approach can capture dependencies at a lower computational cost with minimum errors. In this approach, the classification algorithm is being executed several times with a different subset for each iteration. The better end-point in the features subset is selected as the learning model in the training and testing process's shortlisted features. Then, the Random Forest (RF) algorithm is being used for classifying the best subset feature as either spam or non-spam.

This study proposes a novel framework that identifies tweets spam using machine learning that divides the methods into three sections: pre-processing, feature selection, and classification. According to the minimum Redundancy and

Maximum Relevance (mRMR) method, the study improves the feature selection version. The method is used in real-world applications to select the relevance features for the classification process [15]. The feature extraction is based on the Mutual Information (MI) method, which efficiently handles the multivariate feature extraction and the credit scoring subset feature from the predefined process. This makes it possible for the extraction method to handle the massive calculation of high dimension joint probability by selecting joint relevance or joint redundancy between the predictive related features output [16]. The MI efficiently handle numerical data with noise values and allow a tolerance of errors, induce certain and uncertain decision rules, with data evolving due to its dynamic characteristics. The RMR algorithm can also select new tweets features and remove redundant and irrelevant features from the tweets dataset [17]. For improving the classification efficiency, the process of detecting spam is done by an RF algorithm. Therefore, the potential meaningful knowledge may alter overtime accordingly.

It is a vital issue where the existing twitter spam detection finds it difficult to detect spam accurately, minimize computational cost, and shorten the execution time [18]. The advantage of the feature selection methods is stability in finding relevant subsets of tweet features. The most critical instability is that when the feature selection is applied for pattern recognition accuracy with high-speed up. A previous study proposed feature selection based on the filter method, lightweight statistical, and kNN algorithm chosen to classified spam and non-spam. The existing method's poor evaluation results showed instability classification accuracy. A more considerable amount of data caused over-fitting, the longest time spent in training and testing data for detecting spam. Therefore, the application demanded lower time detection and high-performance accuracy of spam detection. The main contributions of this study are summarized as follows:

1. Introducing a novel framework to identify tweets spam using pre-processing, feature selection, and classification methods.
2. Data pre-processing methods were pre-processed with tokenizing and stemming the words to remove duplication and repetition, thereby achieving pure data. Normalization standardization transforms raw values of features, either positive or negative, and extracting initial tweets features.
3. Presenting a new number of tweets features to improve the performance of the selected classifiers.
4. MI and mRMR algorithms are employed as feature selection methods for dimensionality reduction to further improve performance accuracy and computation time of the RF classifier algorithm.
5. Classifying spam and non-spam. Time measure value of classification algorithms are computed, and run-time is also recorded during experiments.
6. Comparing performance metrics between the proposed framework where the framework achieved a higher classification accuracy and lower computation time.

The rest of the paper is organized as follows. Section II reviews the existing literature on feature selection and classification in twitter spam detection. Section III describes the methodology, including a description of feature selection methods and classifiers used in the study. In Section IV, the description of the utilized datasets for experimental purposes is given. Section V discusses how the experiment is being carried out and results achieved, and finally, Section VI concludes the study.

II. LITERATURE REVIEW

A progression of occurrences has been demonstrated that the security dangers brought by Twitter spam can reach a long way past the online networking stage to affect the present reality. A lot of recent approaches classify Twitter spam to moderate the threat, and promising results are testified. The following are the studies that have been carried out for spam detection and solutions.

The previous studies developed detecting spam methods in online social networks such as URL blacklisting, spam tricks, and crowdsourcing for manual classification. Those methods have a training cost and consumable memory space. The authors in [19]presented AdaGraph as a novel graph-based method to detect spam. AdaGraph used graph clustering technology to analyze user behavior to detect spam in large OSNs effectively. Adagraph continuously updates the detected communities to comply with users' dynamic interactions and activities. According to extensive experiments using the Twitter dataset, AdaGraph detects spam with 92.3% accuracy. Besides, the false detection rate of AdaGraph is less than 0.3%, less than half

of the rate achieved by state-of-the-art approaches.

A new hybrid approach has been introduced by [20] to detect the streaming of Twitter spam in real-time using the combination of Particle Swarm Optimization, classifier using a Decision tree, and Genetic algorithms to identify spam tweets. They compared their results with other hybrid algorithms in which a better detection rate showed that PSG-DT results predicted non-spam ratio from 90 to 97%, and spam include 80–90%. The method was useful to detect spam. The authors suggested improving the performance based on the classifier for streaming spam tweets collected daily and analyzing the detection rate.

Researchers [21] proposed a generalized spammer detection framework called Multi-View Learning for Social Spammer Detection (MVSD), utilizing multiple view information of users and network information to solve the challenge of existing approaches not correctly identified spammers. A real-world Twitter data test results show that the proposed method significantly performs better than existing methods.

Another [22] research proposed a semi-supervised spam detection framework, called S3D. S3D used feature 4 lightweight to recognize tweets spam in real-time and update the models periodically in batch mode. The experiment results demonstrate the effectiveness of detecting spam. The tested proposed framework found that the labeled clusters method is useful in identifying new spamming patterns.

The previous approaches limitation uses the features from a fixed time point to detect spammers, without considering temporal factors. [23] proposed dynamic indicators to quantify changes of User activity and User's temporal evolution patterns. The methods based on detecting the similarity between spammers' used Clustering algorithms (Kullback-Leibler divergence) and monitoring machine learning Spammers in online social networks. Results show that our approach can efficiently distinguish the difference between spammers and legitimate users regarding temporal evolution patterns.

Authors in [5] have investigated the class imbalance problem in machine learning-based Twitter spam detection. However, the current system implies that machine learning techniques have shown that detection efficiency can be severely affected by the imbalanced distribution of spam tweets and non-spam tweets, which is commonly seen in real-time Twitter data sets. They found the solution to the problem by proposed FOS, a fuzzy-based oversampling method that generates synthetic data samples from limited observed samples based on fuzzy-based information decomposition. The researcher developed an assembled learning approach that learns more accurate classifiers from imbalanced data in three steps. In the first step, the class distribution in the imbalanced data set is adjusted using random over-, undersampling, and FOS techniques. In the second step, a classification model is constructed upon each of the redistributed datasets. In the third step, a majority voting scheme is introduced to combine all the classification models' predictions. The experimental results show that the proposed approach can improve the spam detection performance on imbalanced Twitter datasets with a range of imbalance degrees. The work can be extended with the synthetic data generation scheme to incorporate correlations among features.

III. METHODS/ MATERIALS

This section proposes a novel framework of Tweet spam identification methods, as shown in Figure 1.

A. Data collecting

The Stanford Twitter sentiment corpus dataset is introduced and contains 1,600,000 tweets extracted using the Twitter API. The tweets have been tagged negative 0, positive 4. This article gathered 4435 sizes of the dataset and about 1878 row tweet IDs from Twitter API. We extracted all 36 features by analyzing the content of tweet IDs and tweets. The tweets are subsequently processed to reduce noise in tweets and create irrelevant tweets to maximum possible feature extraction and feature selection.

Tweet Features Attribute (dependent variable) has both positive and negative values. The classification algorithm for data sets helps identify spam or non-spam, helping social media sites find spammers before they cause disaster.

B. Pre-processing

Data pre-processing mainly entails cleaning, scaling, transforming data, and saves time. Data pre-processing can be categorized into data cleaning, data integration, data reduction, and data. In this step, we solve the problem of missing values, inconsistent values, and infinite values and get the quality data for the selection and classification of characteristics. For helping the classifier understand the data and building the best possible model, Java JDK, OpenNLP, and the Ling-Pipe library from the natural language processing pipeline were used.

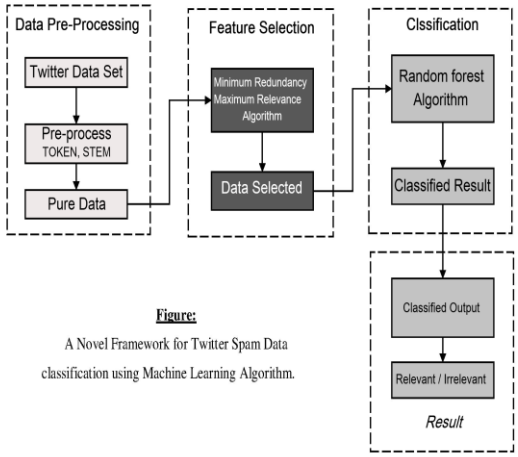

Figure:
A Novel Framework for Twitter Spam Data classification using Machine Learning Algorithm.

Figure 1. Proposed Methodology Framework

The NLP purpose is to recognize, read, and decode analysis to interpret human languages in a comprehensible way. Likewise, the most infrequent textual content classification approach analyzes an incoming message and determines whether the incoming message's temper is assessed as positive, negative, or neutral [24]. The human's intellectual capacity can ace a language without much stretch. The lack of clarity and approximately framed qualities of the natural language make NLP hard for machines to execute. Common steps to method text:

1) Original Data

Originally, input the prepared tweets dataset, and extract the data from the excel spreadsheet to produce quality data. Convert text files to tokenization. Consequently, the NLP pipeline, Java code, and OpenNLP are tools useful to decompose sentences at once.

2) Tokenization

The task of tokenization is to break into parts called tokens, while certain characters, such as punctuation marks, are filtered out in the process using a white space delimiter.

3) Stop words removal

This module erases all unusual and redundant statistics inclusive of is, all, too, this, are, can, to, the, etc. These phrases are known as stop words. They are not needed for analysis, not wanted, and so we remove from our datasets. They are not useful for detecting spam, so these may be eliminated. Stop words are amassed and stored in a text file. Because stop words are not required for analysis, so we loaded and removed the stop words from our dataset. Stop words are available on this Website: https://github.com/arc12/Text-Mining-Weak-Signals/wiki/Standard-set-of-english-stopwords.

4) Stemming

Thus, determining the stem word using the Porter stemmer class support to search stem of a word in a document, the same word can be expressed in various forms, for example, "Kill," "kills," "killing." Also, words can be represented in different syntactic categories, which have the same root form and semantically related, such as 'irony,' 'ironic.' The two above scenarios are common for grammatical reasons. For example, 'Am', "is" are transformed into "be"; “dog”, “dogs”, “dog's” are transformed into “dog” .

5) POS Tagging

This method tags each word and designates components of speech to every word and various other tokens. Part-of-speech classifications consist of nouns, verbs, adjectives, prepositions, pronouns, adverbs, combination, and interjection. The original Porter stemmer algorithm contained the most significant 60 suffixes, one contextual rule for preserving or casting off the suffix, and two recording rules under the Porter algorithm. "Progression," "progress," "progressive," and "progressed" convert to a common stem "progress," suggesting that the 4 words share an unusual definition. Instance word has POS tagging (JJ, JJR, JJS, VB, VBD, VBG, VBN, VBP, and VBZ) of an adjective score and verb score. POS tagger parses a sentence or record and tags each term with its speech component and available on website: http://www.ashleybovan.co.uk/words/partsofspeech.html.

https://www.scrapmaker.com/data/wordlists/language/Nouns(5,449).txt

C. Features scaling

The feature scaling is a critical preliminary step; we processed a larger dataset that was completed from the NLP process. Those tweet data were transformed into a new dataset by reduced unwanted features. We applied data normalization and standardization to make the dataset suitable for further processing. Those tweets feature contains some weight much more than the other features. Thus, there may be a risk that heavy features will overshadow the light ones. We used the Standard Deviation method to reduce high dimensional data from extracted original data and normalization to the unit interval. The computed average, standard deviation, minimum, and maximum of selected Tweets features are shown in Table 1.

1) Normalization

The process of building a relational database according to an array of so-called default formats to reduce data redundancy and improve data integrity. The technique aims to convert raw values of features into positive or negative values. Transformed tweets data are divided into different classes considering the presence of that term. Example: Given two ranges of values, such as the unipolar unit interval [0, 1] or the bipolar unit interval [−1, 1]. To transform these particular values, let assume that tweets feature are labeled by features $X_j = (x_{1,j,}x_2,j,\dots.,x_m,M,j)^{\mathrm{T}}$, $x_{i,min}$ and $x_{i,max}$that are the minimal and the maximal value of a tweet feature X_i for all tweets data located in the learning set; so, given tweets

data symbolized a vector of tweets features $\mathrm{X}_j = (\,x_{1,j},x_{2,j,}\,\dots,x_{m,j})^{\mathrm{T}}$, the equivalent unipolar value is computed as follows.

$$a_{i,j} = \frac{x_{i,j} - x_{i,min}}{x_{i,max} - x_{i,min}} \qquad (1)$$

The equivalent bipolar value is given by

$$b_{i,j} = 2 * \frac{x_{i,j} - x_{i,min}}{x_{i,max} - x_{i,min}} - 1 \qquad (2)$$

The example above presents the tweets feature with values shown in Table 1. Numerical features were converted into nominal data value by the minimal, maximal, and mean values; positions of minimal and maximal values are presented into two vector features horizontally and vertically prediction feature. The continuous section will show the results of original parameters and the values of the entire tweets datasets of minimum, maximum, average, and standard deviation, normalized to unipolar and bipolar unit interval values.

Table 1:
Standard Deviation and Normalization results

Name	Type	Minimum	Maximum	Mean	StdDev	Weight
URL	Numeric	0	1	0.193	0.398	0.479
Hashtag	Numeric	0	1	0.096	0.294	0.317
Number_of_forensic	Numeric	6	157	81.586	35.347	0.230
Number_of_Negative_Words	Numeric	1	32	14.73	6.975	0.124
Length_of_Tweets	Numeric	0	1	0.001	0.023	0.22
Number_of_Words Calculation	Numeric	0	3	0.288	0.552	0.502
Number_of_Nouns	Numeric	0	18	4.553	3.008	0.053
Number_of_verbs	Numeric	0	12	3.077	2.183	0.070
Number_of_Adverbs	Numeric	0	13	2.352	1.887	0.084
Number_of_Adjectives	Numeric	0	12	2.193	1.756	0.119

2) Standardization

Standardization is a type of unification method that considers raw qualities themselves and the scattering of values; that is, we utilize the mean value and standard deviation of a given tweet feature. Let the following vector $X_j = (x_{1,j,}x_2,j,\dots.,x_m,M,j)^{\mathrm{T}}$ signify to j,th sample. The following is the calculations to understand a standardization technique,

$$\mu_{i,j} = \frac{x_{i,j} - \bar{x}}{\sigma_i} \qquad (3)$$

Here $x_i$is the mean of the feature $x_i$and σ_i is the standard deviation of this feature

$$\overline{x_i} = \frac{1}{N}\sum_{j=1}^{N} x_{i,j,}$$

$$\sigma_i = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(x_{i,j} - \overline{x_i})^2} \qquad (4)$$

where $x_{i,j}$represents the result value after processing, x represents the original value, $\overline{x_i}$represents the mean of the column features, σ_i represents the standard deviation of the column features, and m represents the dimensions of the attribute.

N is the number of features members in the learning set.

$\sigma_i$is the sample standard deviation

$x_{i,j}$is1..., n number of features associates' sample and $\overline{x_i}$ is the sample Mean.

The above equations are utilized to multiply the test for standard deviation rate, a larger dataset for finding the number of sample, mean, deviation between features value and the variance square root. This will enable accurate performance with speed and produce results that

are more compact, concise and precise over continuous data shown in Figure 2.

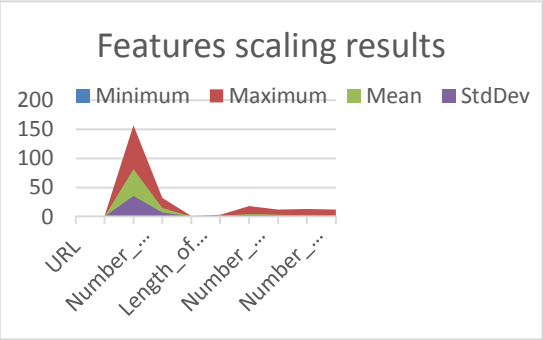

Figure 2: Features scaling results

D. Feature selection/ Feature extraction

1) Mutual information

The mutual information method is a crucial way to learn how mapping a large number of input tweets features to the output class label.
MI is a significant criterion for measuring the correlation of subset tweets features. MI used to search the credit score subset tweets features, which satisfies the criterion "max relevance and min redundancy.

Formally, the mutual information of two discrete random variables, A and B, can be defined as

$$I(A;B) = \sum_{a\epsilon A}\sum_{b\epsilon B} P(a,b)log\left(\frac{P(a,b)}{P(a)P(b)}\right) \quad (5)$$

where *p (a, b)* is the mutual probability function of *A* and *B*, and *p (a)* and *p (b)* are the non-linear probability distribution functions of *A and B*, respectively.
Similarly, in the case of continuous random variables, the summary is replaced by the determined double integral.

$$I(A;B) = \int\int P(a,b)\log\left(\frac{P(a,b)}{P(a)P(b)}\right) dadb \quad (6)$$

where p (a, b) is now the mutual probability density function of *A* and *B*, *p (a)* and *p (b)* are the marginal probability density functions A and B, respectively. MI can be equally stated as

$$I(A;B) = H(A) + H(B) - H(A,B) \quad (7)$$
$$OR$$
$$I(A;B) = H(A) - H(A/B) \quad (8)$$

$$where\ H(a) - P(a)\sum_{a\epsilon A} P(a),\ H(b) = -P(b)\sum_{b\epsilon B} P(b) \quad (9)$$

Signify the entropy of *A* and *B*, $H(a\backslash b) = H(a,b) - H(b)$ signifies the conditional entropy of the involved variables.

The mutual information method selected 14 best features out of 36 tweets features from the pre-processing stage. Those attributes best scored obtained from MI are highly correlated and an increase of dimensionality. Therefore, we presented the mRMR algorithm to decrease redundancy between tweets features and select the best subset from the training model. Minimum redundancy and maximal relevance (mRMR) is a multivariate filter technique, which finds the best tweets featuring *m* with the highest dependency for the classification process.

2) Maximum dependency with minimum redundancy

Feature selection employs a maximum relevance of minimum redundancy and chooses the important tweets feature subset for a given classification task. The tweets feature output will be new features grouping with maximum relevance features and with no duplication. The purpose is to discover maximum dependency and redundancy features between two variables. Thus, we introduced techniques for hypothesis in our research to support enhanced performance accuracy, decreasing over-fitting, and low computational expense and gave high-quality data for classifiers, as shown in the algorithm.

Maximal relevance is a method selecting the features from the tweet's dataset with the mean value of all dependency value between individual featurex_i and the target class *C* label D.

$$\text{Max}\Phi(D,R), \phi = D - R \quad (10)$$

Where *D* means the relevance or dependence and is calculated as

$$maxD(S,C), D = \frac{1}{|s|}\sum_{x_i\epsilon s} I(X_i C) \quad (11)$$

Selecting features based on maximum correlation criteria can bring much redundancy. Therefore, the following minimum redundancy was used to remove irreverence and redundancy tweet data. We use the "maximal-relevance" standard to select relevant features and discard irrelevant features at first.

Minimum redundancy: Thus, the method is applied in the tweets dataset to selected irreverent or duplicated features relationship between features and target class to reduce the dataset's size. Let S* donate the subset tweets feature and $|S|$ is a number of features in S. R means redundancy, it is calculated as

$$minR(s) = \frac{1}{|s|^2}\sum_{x_i x, \epsilon\, s} I(x_j x_i) \qquad (12)$$

where C is the classification target class and $I(x_j x_i)$ is the mutual information between the individual features x_j *and* x_i.

The mRMR feature set is obtained by optimizing the conditions in formulas (11) and (12) simultaneously to select features, Figures that are highly relevant to the standard, no duplicate data can be expressed in equations (13).

This gives a subset of features S^* selected using mRMR,

$$S^* = argmax_{s\subseteq F}[\max D(S,c) - minR)(s)] \qquad (13)$$

Algorithm 1: Minimum Redundancy Maximum Relevance Algorithm

Input: Extracted Tweets Features

// Data-Tweets Dataset

//Features- 36 Tweets Features in Tweets Dataset

Output: Selected Features from Extracted Features// *Set of Tweets features selected from Tweets datasets*

#Applying Minimum Redundancy Maximum Relevance to the Extracted Features

Step 1: Initialize variables

Step 2: Read the Extracted Features

Step 3: Count the number of records

For Feature f_1 in Extracted Features

Do

//Check Relevance between the class label and Feature

R=mutual-Info (f_1, class)

Redundancy=0;

Step 4: Check the Redundancy of the Feature

For Feature f_2 in Extracted Features

Do

Redundancy=Redundancy + mutual Info (f_1, f_2,);

End For

Step 5: *//Store Minimum Redundancy Maximum Relevance Value*

Mvalues [f1] =Relevance-Redundancy

End For

Step 7: Sort the Selected Features to get number-of-selected features

Selected Features=sort(M-values).take (Number-of-Selected-Features)

Step 8: End

Thus, the mRMR algorithm gives the idea to measure the mutual information among two elements, either a given element of this class or a pair of input features through the level of those features presented in Table 2.

Table 2:
Best features selected by mRMR Algorithm

Tweet feature	Selected with mRMR algorithm
F2	Best Feature : Index : 1
F3	Best Feature : Index : 2
F4	Best Feature : Index : 3
F5	Best Feature : Index : 4
F9	Best Feature : Index : 7
F10	Best Feature : Index : 8
F11	Best Feature : Index : 9
F12	Best Feature : Index : 10
F13	Best Feature : Index : 11
F14	Best Feature : Index : 12
Best features	1,2,3,4,7,8,9,10,11,12
Results	1,0,114,20,0,0,1,1,0,0,13

E. Classification

1) Proposed Algorithm

Random forest (RF) is an ensemble learning approach and regression method appropriate for solving classification problems and improving performance accuracy. Random forests have been used effectively to create a spam detection model. The random forest advantages: it generally reduced classification error and higher f-scores compared to decision trees. Performance is high when comparing other algorithms such as SVM, KNN, and Naive Bayes. It provides an efficient mechanism for calculating the approximate value of missing information and preserving accuracy in situations where a considerable percentage of the information is absent. Overfitting is a critical problem that can worsen the results. However, it generates enough trees in the forest for the Random Forest algorithm to help the classifier not over-fit the model. Random forest steps of classification are the following:

Step 1: We used the selected subset tweets features from feature selection containing features having (matrix) to create random samples. We used the selected tweet dataset (n rows and m Columns) to build a new dataset using a data training set.

Step 2: When splitting a node, the best split is found over a randomly selected subset of *M* predictor features instead of all S*predictors, individually at each node. The randomizations used to tweet feature the predictor gives the remaining part of the trees is grown without pruning. The Tweets spams are estimated by grouping the predictions of m trees into the majority voting technique.

Step 3: In the training model on a new dataset sample is used to determine and minimize errors of classification node, the discretization of each

continuous feature used to measure the best node tree using sample variance formula

$$S^2 = \sum \left[(x_i - \bar{x})^2_{(n-1)} \right] \qquad (14)$$

Let S^2 is variance
x_i is tweets feature

$$\sum meaningissum$$

$\bar{x}$ is the mean of the sample
n is the number of features point

Step 4: The result used to divide a node, the best splitting is found by the formula disposed, it is calculated individually at each node. We use cross-validation to determine the test error rate of the subtree.

Step 5: In this algorithm, several trees grow simultaneously, and the final prediction is made by collecting various decisions to obtain the best classification accuracy. Tweet spam is determined by grouping m-tree predictions using majority voting. It shows the values of the classification of spam and non- spam.

In our random forest implementation, we have selected a vector of 6 unplanned features to build each tree in a forest of 100 random trees. The tree grows to its maximum depth as the argument is set to zero, indicating unlimited depth. By using bagging and voting techniques, classification is being done, as described in Algorithm 2. The dataset consists of 4435 records having a number of attribute 36 based on tweets spam detection initial. The dependent variable is spam tweets; it has two values called Spam Tweets – Spammer and Non-Spam Tweets: Non-Spammer results are shown in Table 3. The work done in this paper can hence be used by social media as protection to classify new tricks of spammers as spam or non-spam based on machine learning methods.

Algorithm 2: Classification using Random Forest Algorithm

Input: Selected Attributes.
Output: Classified Output
Procedure:
Step 1: LetN_F be the number of features
Step 2: Let N_R be the number of Records
Step 3: T_R=70%, T_S=1-T_R *//Split the dataset S* to training and testing set in* 70:30
Step 4: Split the data into training and testing set
Step 5: [Train]_R =N_R (T_R) *// Read the Tweets (Training)*
Step 6: [Test]_R=N_R (T_s) *// Read the Tweets (Testing)*
Step 7: Fun RF ([Train]_R, N_F) *//execute random forest to reduce sample and minimum* error rate
Step 8: Int h = 0
Step 9: For k=I to n Tree *//traverse trees in the forest*
Step 10: Si = sample; hi = RTL(Si,F)
Step 11: H = H + hi *//calculate the best features and indexing for the list*
Step 12: End For
Step 13: return H
Step 14: End Fun
Step 15: function RTL (S,F) //
Step 16: f = subset of f *// the feature with the best value is selected*
Step 17: split the best set of features f *// Aggregate all the nodes' votes in the trees.*
Step 18: return RLT
Step 19: End Fun // *Final tweets feature Classification (Spam and Non-Spam)*

Table 3:
Datasets used for the experiment

Twitter dataset		
Methods	Label	Number of Tweets ID
Existing system	Twitter Spammer	1940
	Twitter Non-spam	1608
	Instance	4435
Proposed system	Twitter Spammer	6548
	Twitter Non-spam	5803
	Instance	13,154

F. Existing system algorithm

1) Naive Bayes for classification

The Naive Bayes algorithm is based on the ML algorithm that implements Bayes' Theorem and belongs to probability classifiers. As the name suggests, the NB algorithm is based on the naive assumption that all features' contribution to its output category is independently performed based on probability theory, which does not explicitly use any representation of the classifier. This stage computes the posterior probability of tweets spam given the broad probability of the sampling twitter by Bayes rule shown in the entire classification process[25].

$$P(S|T) = \frac{P(T|S)P(S)}{P(T)}$$

$$P(S|T) = \frac{P(T|S)\prod_{i=1}^{n} P(w_i \mid S)}{P(T)}$$

And similarly

$$P(N|T) = \frac{P(T|N)P(N)}{P(T)}$$

$$P(N|T) = \frac{P(T)\prod_{i-1}^{n} P(w_i \mid N)}{P(T)}$$

We classify the tweet features by comparing the probability of *P(S/T)*. Thus, the probability of a given tweet feature is classified as spam belonging to the tweets class S. *P(H/T)* is the probability of the given tweets features, classified as non-spam belonging to the tweets class (N).

2) Support Vector Machines (SVMs)

Support Vector Machines (SVMs) are a supervised learning technique used for classification, regression, and outlier detection. SVM has found application in providing results to quadratic programming problems with inequality constraints and linear equality by differentiating different classes. SVM constructs a hyperplane to separate the set of data features having different labels in an immeasurable dimension. The SVM keeps the training error fixed while minimizing the confidence interval. We applied the SVM technique to construct an n-1 dimensional separating hyper-plane to discriminate two classes in an n-dimensional space. A data is viewed as n-dimensional tweet features; thus, two features in the dataset will create a two-dimensional feature field. The distance between the hyper-plane and the nearest data point on each side (called support vectors) is maximized.

3) K-Nearest Neighbor

The nearest neighbor is a flexible, non-linear, and straightforward classification algorithm that does not rely on the assumption that the data is extracted from a given probability distribution. KNN algorithm is one of the simplest classification algorithms; it is the most commonly used in unsupervised learning and supervised learning. Instance-based learning methods are often referred to as "lazy" because the classification process is delayed until new instances arrive. The drawback of the KNN algorithm has a medium speed of training data. Computationally expensive because, to be precise, the entire training phase requires the entire training data when making decisions based on the entire training data set. High memory requirement with sensitive to irrelevant features scale of the information outline[26].

IV. EXPERIMENT AND EVALUATION

A. Experimental setup

We exhibited our experiment on the desktop computer with specification details such as processor (CPU) intel i5, speed 3-4 GHz, RAM 32GB, graphic memory 3GB, window 10 professional x 64- bit. This requirement satisfies our experiment condition. We have used java programming language with other software and tools for the implementation.

B. Datasets

In this study, the open-source dataset from Stanford Twitter sentiment 140 and contains 1.6 million tweets is used for spam detection. The dataset link in http://help.sentiment140.com/for-students. We created a new dataset label positive that indicates spam and negative indicates non-spam in the dataset. Each file inside contains the value of total Tweet ID 4435, User name, tweet messages, and total size 4435 of the tweet dataset. Then we load the tweet dataset for pre-processing using the NLP pipeline, including deleting redundant tokens, stop word, stemming, pre-processed before being used as input in future selection and classifier. In our research, we define tweets that contain malicious URLs as Twitter spam. We can identify if a given URL in which category URL belongs to spam or non-spam tweets.

Table 5:
Number of Tweet features executed in pre-processing

Number of features in Bag-of-words feature set	Total words
Total Number of words in Bag-of-Words feature set Token	11138
Total Number of words in Bag-of-Words feature set of Stop words	5092
Total Number of words in Bag-of-Words feature set after stop-words removal	6044
Total Number of words in Bag-of-Words feature set after stemming	6256

C. Experiment

In this study, we experiment to verify the effectiveness and applicability of the proposed method. The first experiment's primary purpose is to verify the performance of the existing system in detecting spam and performance levels. The second purpose of the experiment is to verify the effectiveness of the proposed method of identifying tweet spam using the features selection to improve performance accuracy and computational time. Also, for reasonable comparison motives, parameters of all methods were set up with values. When the experimental data are analyzed, MI and mRMR algorithm, Random forest and existing four algorithms, their parameters are mentioned earlier. The settings of all conducted experiments are as follows:

1) Data processing

Initially, pre-processing stages were used to eliminate fewer words, combine word forms such as plural and verb links into simultaneous events, predict interesting tweet features from unstructured tweet data. Redundant tweet features have a missing value of 80%-100%, removed words were about 6044 rows, and indexed the features documents' data file size. The total tokens 11,138 rows stop word processing account words in 30%-40% of total word counts is 5092 in the tweeted document.

Table 4:
The number of Tweet features executed from the Tweets Dataset.

AttNO/Method	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13
MI	√	√	√	√	√	√	√	√	√	√	√	√	√
mRMR	x	√	√	x	√	x	x	√	√	√	√	√	√
Existing methods	√	√	√	√	√	√	x	√	√	√	√	√	x

The stemming process reduced the index size by 40-50% words, and similar, usable, punctuation brackets, hyphens, dots, and URL links were removed from the document, as shown in table 4. Our experiment pre-process stage reveals that URLs and hashtag duplicated or bad tweets were removed by stop word, and stemming is applied and successful.

2) Features scaling

Normalization is used for data reduction methods that transform continuous features into discrete features. We can reduce the total data volume of continuous attributes; the Normalization process involves the partition of continuous attribute values into several intervals. Tweets dataset contains the missing value, noise, and process cleaning data by removing noise and missing value at the end categories features to a new dataset and filling the missing value of number features by mean value of corresponding features. Afterward, a new tweet dataset with numerical value transformed, and standardization applies a score to transfer features value into the same unit according to the feature matrix columns. Normalization aims to set a unified standard when calculating the similarity by dot multiplication that can transfer the value of each feature into a unit vector. We achieved a reduction in the total data volume of continuous features.

3) Feature extraction and feature selection

We presented Mutual information and mRMR algorithms for dealing with large datasets in real-time detection spam, unapproachable computational problems. We implemented the feature selection by using Mutual information methods for feature extraction of the optimal tweets feature subset for credit scoring. Afterward, a new tweet dataset was obtained with 14 features. We used the mRMR algorithm to select the best subset of features from the Tweets dataset to build classifiers for all possible subsets of features. The summary of features selected with MI and the mRMR algorithm is described in the table.

4) Classification

The dataset obtained after irrelevant and redundant features were used to perform classification methods using RF and other existing algorithms for comparison. The tweets datasets were split into training and testing set in ratio 70:30, respectively. For the proposed system, 9206 rows are used for training and for testing 3947 rows. The total amount of tweets data 1878 for both systems and the mRMR algorithm are selected to continue for classification. The description of the selected features is shown in Table 2. The models are tested and validated using the NetBeans IDE 8.2 java software by running various novel

frameworks that, in the end, display results for overall performance evaluation of mode.

D. Evaluation metrics

Different evaluation metrics are used to measure the performance of the below-proposed twitter spam detection methods. These include accuracy, recall (sensitivity), precision, and F1 Score.

Accuracy is the capability of an approach to differentiate spam and truthful tweets features correctly. It is measured by calculating the proportion of TP and TN in all cases assessed.

Accuracy= (TP+TN)/(TP+FP+TN+FN)

Recall measures the proportion of correctly identified tweets features. It shows how effective an approach is in detecting tweets spam

Recall=TP/(TP+FN)

Precision measures the number of tweet features classified as spam is correctly predicted.

Precision=TN/(TN+FP)

The F1 score is the weighted average of recall and accuracy.

F1 SCORE=2TP/(2TP+FP+FN)

True-Positive (TP): the number of correctly predicted tweet spam, which means that the real class's value is spam and the value of the predicted class as well as spam.

True-negative (TN): number of truths correctly predicted Tweets features, which means that the value of the real class is true and the value of the predicted class as well as true

False-Positive (FP): number of incorrectly predicted tweets spam, which means the value of the real class is spam, but the value of the predicted class is true.

False-negative (FN): number of incorrectly predicted truths Tweets Feature, which means that the value of the real class is true, but the value of the predicted class is spam.

V. RESULTS AND DISCUSSION

A. Performance stability

Table 6 shows the Tweets dataset's performance comparison using the proposed feature selection and existing methods features selection. The different valuation metrics have been used to examine the performance accuracy. By analyzing the designed graph in Figures 3 and 4, the proposed framework's performance is meaningfully improved by selecting the best features using minimum redundancy maximum relevance (mRMR). The classification is done using Random Forest. Compared with existing classifiers KNN 80% using 13 lightweight feature choices, proposed methods achieved 90% higher than existing systems of 80%, Naïve Bayes 79%, SVM 75% are tested without using the mRMR algorithm. The result of existing algorithms shows dropping in accuracy compared with proposed methods. Random forest algorithms show stability compared with others in terms of detecting tweet spam. In proposed methods, spam was found, as shown in Table 3.

Table 6:
Comparison of Proposed feature selection and Existing features selection

Algorithm	Accuracy
KNN	80.00%
Naïve Bayes	79.00%
SVM	75.00%
RF & mRMR	90.00%

B. Scalability

1) Tweets features

In this section, we focus on the effect of an increase in the size of the training set computational performance of algorithms in terms of training time. We applied features

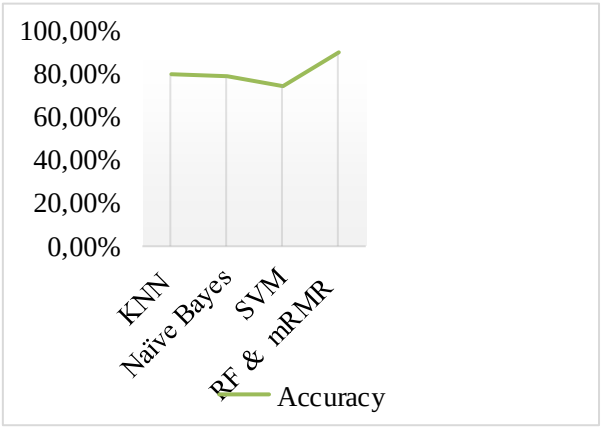

Figure 3: Comparison of proposed methods and existing methods

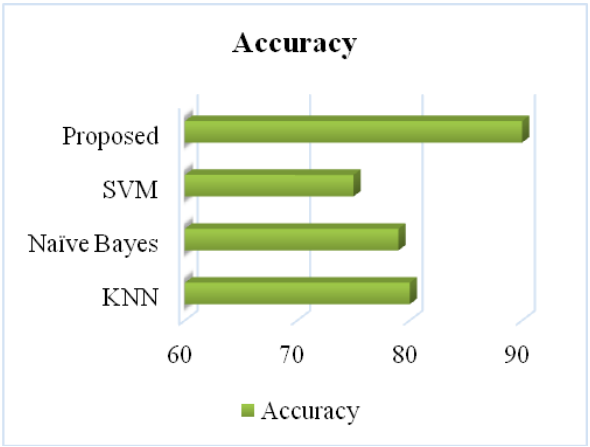

Figure 4: Comparison of proposed methods and existing methods

extraction using the MI algorithm to extract the most relevant subset features that depend on features indexing values of features. We found out that MI may cause problems with dropout feature experience during mutual information algorithms' execution. The results show that mutual information can score a subset of 14 Tweets features, shown in Table 7, but those features presented data with higher dimensionality. Therefore, we applied the mRMR algorithm to reduce the higher dimensional and select a relevant subset for the classification model. The results showed that 10 best features were selected for training the model out of 14 features. Our proposed method is useful to remove high dimensions and select optimal features for the training model. We introduced new features such as the number of Terrorism Words, the number of forensic words that twitter user sent, and those new features added to the selected features set. The results show that the mRMR and MI methods are an optimal feature subset from within the candidate set to determine the best candidate feature for the tweets dataset. It proves that a new feature introduced by such a forensic number is the target feature for spreading spam. Those methods may lead to high classification performance and minimize classifier error by using RF.

Table 7:
Tweets Dataset features

AttNo	Features Name	Explanations	Reference
F1	Tweet ID	The number of tweet Id	
F2	URL	The number of URLs included in this tweet	[27][1][28]
F3	Hash Tag	The number of hashtags included in this tweet	[27][1]
F4	Length of Tweet	The number of length of the tweet	[4][29]
F5	Number_of_Words Calculation	Counting the number of times a word appears in n tweets	[4][29]
F6	Number_of_Digits	The number of digits in this tweet	[30][28]
F7	Number_of_Characters	The number of characters in this tweet	[27][1][28][29]
F8	Number_of_Terrorism_Words	The number of terrorism words	NO
F9	Number_of_Negative_Words	The Number of negative words	[31]
F10	Number_of_Nouns	The number of nouns in this tweet	[31]
F11	Number_of_Verbs	Number of Verbs in these tweets	[31]
F12	Number_of_Adverbs	The number of adverbs in this tweet	[31]
F13	Number_of_Adjectives	Number of the adjectives in this tweet	[32]
F14	Number _of_Forensic words	The number of forensic words this twitter user sent	NO

2) Computation time

We have thought of observing the running time required for executing the whole structure of the random forest and mRMR system based on proposed methods and existing frameworks. The description of the running time result is given in Table 8 and Figures 5 and 6.

Total Execution Time (Random Forest): 0hr 0m 20s

Existing System: -

Total execution time (KNN): 2.022 seconds.

Total execution time (Naïve Bayes): 2.8 seconds.

Total execution time (SVM): 3.8 seconds.

It can be noted that the previous system, the time required for data classification, is very low.

Hence, the system can be used to operate in real time.

Table 8:
Proposed Random Forest Time Spending of Dataset

Algorithm	Time(Seconds)
KNN	114
Naïve Bayes	174
SVM	228
RF & mRMR	20

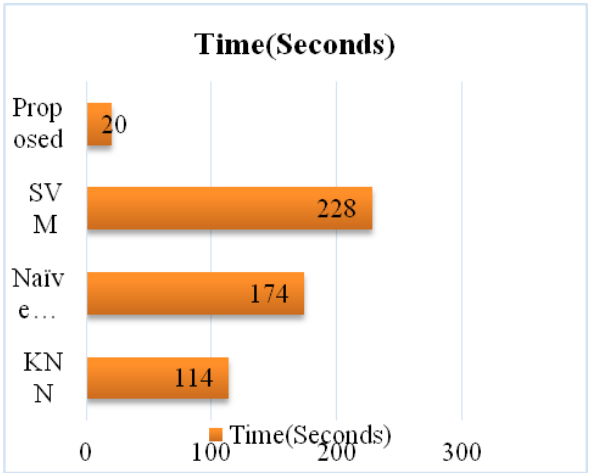

Figure 5: Comparison of proposed methods and existing methods running time

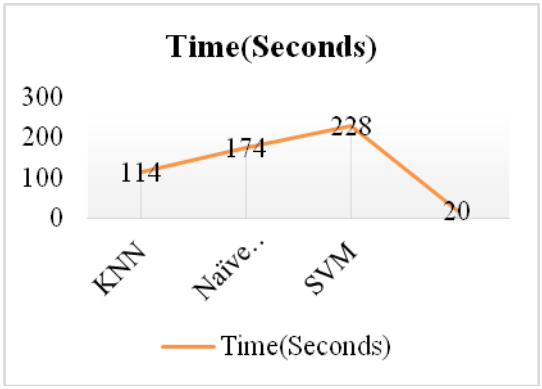

Figure 6: Comparison of spam detection speed

Table 9:
Comparison of Precision, Recall and Accuracy, F-score

Algorithm	Accuracy	Precision	Recall	F-score
Proposed	90	97.87	95.82	96.83
SVM	75	77.21	79.93	79.01
BN	79	80.32	82.42	81.15
KNN	80	83.46	80.11	83.9

C. Discussion

1) Performance stability

The random forest increased performance accuracy based on the experiment outcomes compared to the previous method parameters such as precision, recall, and F-measure using tweets datasets despite the other three algorithms experiencing a significant drop in their measured value. Our proposed methods accuracy show higher (90%) value of the measure compared to existing methods the KNN (80%), BN (79%), SVM (75%); which means the higher the value of measured, the higher the ability to identify between spam and non-spam using tweets dataset. So the proposed method is stable in identifying Tweets spam. Thus, improving performance accuracy random forest proved a robust algorithm in real-time spam detection based on Tweets datasets. We noted that random forests become better in classification tasks with larger numbers of decision trees (100). Thus, evidence that the false identification level is very low for tweet datasets. Therefore, all the best features selected by feature selection are suitable for generation in the classification model. We found out that the existing method is slow in the training model. Adding more training samples might be the risk of causing over-fitting. The experiment revealed is unsteady for an enormous quantity of datasets.

2) Scalability

a) Feature selection stability

In this study, we focus on feature selection, which multidimensional tweet data. Let S^* be the number of features in the tweet dataset, 10-fold cross-validation is employed. As a result, we created 10 times randomly selecting 70% testing and 30% training data in each time's rows and taking all S* tweets features, the total amount of features, m features are shortlisted by the mRMR algorithm. Furthermore, S^* and M-value are used for classification purposes. Our model proved efficiency and stability in reduced high dimension features and selected the classification model's best features sets. The method achieved these findings on higher performance measures compared with existing methods [18]. The best feature sets are selected with low complexity and low error rates. We discovered that some features are a high spam rate, such as a number of forensic; the number of words calculation and URL are selected with minimum complexity with less complexity and high accuracy than the original feature space. We used less essential features to train the model so that models can be trained faster. The best feature set which has been used to build the final model is shown in Table 9.

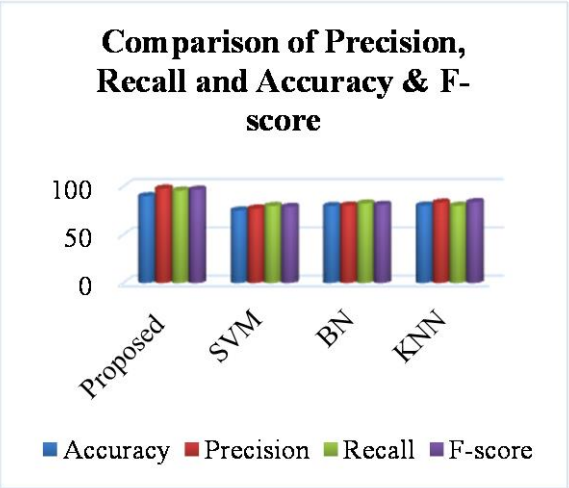

Figure 7: Comparison of Precision, Recall and Accuracy, F-score

b) Time complexity

The experiment is supported on processor (CPU) intel i5, speed 3-4 GHz, RAM 32GB window 10 professional 64-bit, KNN algorithm was overlapping data and took the longest time to detect spam compared to other algorithms. Existing dataset having highly dimensional data creates difficulty in run-time is putting most time into advised system dataset. In our proposed system, KNN algorithm results show improvement compared to NB and SVM algorithms. We discover that previous techniques of spam detection took a long time to detect spam without mRMR methods. We applied the feature selection algorithm and classification algorithm and observed that the classification algorithm detects spam with less complexity and high accuracy. Random forest algorithms used minimum running time 0hr 0m 20s to complete detecting spam compared with all existing algorithms. At the same time, KNN took 2.022 seconds to complete, SVM based algorithms are the most time consuming with 3.8 seconds, followed by BN 2.8 second. We achieved resolving time computation. The Random Forest algorithm took long hours for a larger dataset in a previous study, while our proposed techniques take a short time to complete the spam detection.

D. Limitations

There are several difficulties encountered during this research while trying to identify Tweets spam from Tweet datasets. In spam detection jobs, it is difficult to identify Spammer's pattern based on previous historical data because spammers always change the way of committing scam and use new ideas and strategies such as the number of terrorist words and tweets length features. The solution part is to recognize the best features that can detect spammers required to improve feature selection in wrapping methods. Overlapping data, we discovered that sometime the legitimate data might be thought of as spam and vise-versa. Tweet spam may look authorized on counting for false negative. The performance measure is limited across the different models, and there is an essential improvement over the model selected. Furthermore, we discovered that spammers concentrated on using the number of forensic words, the number of length of the tweet, number of terrorism words, number of characters, number of digits, number of URLs, and words to spread spam. In our experiment, the proposed method finds difficulty identifying tweets ID spam on those feature subsets, number of terrorism words, number of characters feature subsets.

VI. Conclusion

This paper proposed a novel framework for identifying twitter spam data using machine learning algorithms to address the feature selection issue. The proposed methods are categorized into three sections: pre-processing, feature selection, and classification. The experimental results on a series of multilevel datasets demonstrated that our novel framework could effectively identify the best feature subset and minimize larger dataset size errors. Furthermore, we compared with the existing methods of twitter spam detection versions such as Naïve Bayes, SVM, KNN to show the importance of feature selection results. In classification, our experiments' results show the identification of spam and non-spam for real-time running classification jobs. A novel framework method seems to have the best of all performance accuracy and running time. Existing methods showed a long time for training data, overlapping, and performance measures are lower, and some features were dropped during the selection period. The constraints of our proposed methods are discussed insection D. The contribution of this research can: (1) reduce dimension from tweets datasets, (2) identify tweet spam in real-time, (3) increase speed up in detecting spam, (4) select the best features for learning model and (5) enhancing efficiency accuracy tasks. In future research, we plan to improve the performance measures using different new features streaming of spam tweets active daily on social networks. The imbalance dataset issue is a problem worthy of further study in the future. Hence, incoming new tweets, often unavailable on the twitter database, become

suitable in real-time spam detection. Studying the effect of feature selection engineering methods and being deprived of memory space also need to be addressed.

ACKNOWLEDGMENT

The author would like to thank Dr. Amir Rizaan Abdul Rahiman, Prof. Dr. MohdTaufik B Abdullah, and Dr. Abdullah bin Muhammed for the fruitful discussions and exchange of ideas about a multitude of aspects related to social media, spam content detection, and the motives of spammers. Also, sincere appreciation goes to University Putra Malaysia for its support in this research work.

REFERENCES

[1] ALARIFI, A., ALSALEH, M. and AL-SALMAN, A. M. (2016) Twitter turing test: Identifying social machines. *Information Sciences*, 372, pp. 332–346. doi: 10.1016/j.ins.2016.08.036.

[2] GUPTA, H., JAMAL, M. S., MADISETTY, S. and DESARKAR, M. S. (2018) A framework for real-time spam detection in Twitter. In: P*roceedings of 10th International Conference on Communication Systems & Networks, Bengaluru, January 3-7, 2018*. Piscataway, New Jersey: IEEE, pp. 380–383. doi: 10.1109/COMSNETS.2018.8328222.

[3] MILLER, Z., DICKINSON, B. DEITRICK, W., HU, W. and WANG, A. H. (2014) Twitter spammer detection using data stream clustering. *Information Sciences*, 260, pp. 64–73, , doi: 10.1016/j.ins.2013.11.016.

[4] ADEWOLE, K. S., ANUAR, N. B., KAMSIN, A. and SANGAIAH, A. K. (2019) SMSAD: A framework for spam message and spam account detection. *Multimedia Tools and Applications*, 78(4), pp. 3925–3960. doi: 10.1007/s11042-017-5018-x.

[5] LIU, S., WANG, Y., ZHANG, J., C. CHEN, and XIANG, Y. (2017) Addressing the class imbalance problem in Twitter spam detection using ensemble learning. *Computer Security*, 69, pp. 35–49. doi: 10.1016/j.cose.2016.12.004.

[6] ZHENG, X., ZENG, Z., CHEN, Z., YU, Y. and RONG, C. (2015) Detecting spammers on social networks. *Neurocomputing*, 159(1), pp. 27–34, doi: 10.1016/j.neucom.2015.02.047.

[7] KAUR, R., SINGH, S. and KUMAR, H. (2018) Rise of spam and compromised accounts in online social networks: A state-of-the-art review of different combating approaches. *Journal of Network and Computer Applications*, 112, pp. 53–88. doi: 10.1016/j.jnca.2018.03.015.

[8] LABANI, M., MORADI, F. AHMADIZAR, P. and JALILI, M. (2017) A novel multivariate filter method for feature selection in text classification problems. *Engineering Applications of Artificial Intelligence*, 70, pp. 25–37. doi: 10.1016/j.engappai.2017.12.014.

[9] HOSSEINI, E. S. and MOATTAR, M. H. (2019) Evolutionary feature subsets selection based on interaction information for high dimensional imbalanced data classification," *Applied Soft Computing*, 82, p. 105581. doi: 10.1016/j.asoc.2019.105581.

[10] CAI, J., LUO, J., WANG, S. and YANG, S. (2018) Feature selection in machine learning: A new perspective. *Neurocomputing*, 300, pp. 70–79, doi: 10.1016/j.neucom.2017.11.077.

[11] ALNUAIMI, N., MASUD, M. M., SERHANI, M. A. and ZAKI, N. (2019) Streaming feature selection algorithms for big data: A survey. *Applied Computing and Informatics*, 16(1/2), pp.1-23. doi: 10.1016/j.aci.2019.01.001.

[12] AL-ZOUBI, A. M., FARIS, H., ALQATAWNA, J. and HASSONAH, M. A. (2018) Evolving Support Vector Machines using Whale Optimization Algorithm for spam profiles detection on online social networks in different lingual contexts. *Knowledge-Based Systems*, 153, pp. 91–104. doi: 10.1016/j.knosys.2018.04.025.

[13] SUCHETHA, N. K., NIKHIL, A. and HRUDYA, P. (2019) Comparing the wrapper feature selection evaluators on twitter sentiment classification. In: *Proceedings of 2019 2nd International Conference on Computational Intelligence in Data Science (ICCIDS). Chennai, October 24-26, 2019.* Piscataway, New Jersey: IEEE. doi: 10.1109/ICCIDS.2019.8862033.

[14] UÇAR, M. K. (2020) Classification

performance-based feature selection algorithm for machine learning: P-Score. *IRBM*, 41(4), pp. 229-239, doi: 10.1016/j.irbm.2020.01.006.

[15] SULAIMAN M. A. and LABADIN, J. (2015) Feature selection based on mutual information. In: Proceedings of *2015 9th International Conference on IT in Asia (CITA 2015), Sarawak, Kuching, Malaysia, August 4-5, 2015.* Piscataway, New Jersey: IEEE. pp. 1–6. doi: 10.1109/cita.2015.7349827.

[16] ZHOU, H., WANG, X. and ZHANG, Y. (2020) Feature selection based on weighted conditional mutual information. *Applied Computing and Informatics*, 16(1/2), ahead-of-print. doi: 10.1016/j.aci.2019.12.003.

[17] AMIRI, F., REZAEI M., YOUSEFI, C., LUCAS, C., SHAKERY, A. and YAZDANI, N. (2011) MUTUAL information-based feature selection for intrusion detection systems. *Journal of Network and Computer Applications* 34(4), pp. 1184–1199. doi: 10.1016/j.jnca.2011.01.002.

[18] LIN, G., SUN, N., NEPAL, S., ZHANG, J., XIANG, Y. and HASSAN, H. (2017) Statistical Twitter spam detection demystified: performance, stability and scalability. *IEEE Access*, 5, pp. 11142–11154. doi: 10.1109/ACCESS.2017.2710540.

[19] AMIRA, S. and GIRDZIJAUSKAS, S. (2017) AdaGraph : Adaptive Graph-Based Algorithms. In: *Proceedings of 5th International Conference (NETYS 2017), Marrakech, Morocco, May 17-19, 2017.* Berlin: Springer, pp. 338–354. doi: 10.1007/978-3-319-59647-1.

[20] MURUGAN, N. S. and DEVI, G. U. (2018) Detecting Streaming of Twitter Spam Using Hybrid Method. *Wireless Personal Communication*, 103(2), pp. 1353–1374. doi: 10.1007/s11277-018-5513-z.

[21] SHEN, H., MA, F., X. ZHANG, L. ZONG, X. LIU, AND W. LIANG (2016) Discovering social spammers from multiple views. *Neurocomputing*, 225, pp. 49–57. doi: 10.1016/j.neucom.2016.11.013.

[22] SEDHAI, S. and SUN, A. (2018) Semi-Supervised Spam Detection in Twitter Stream. *IEEE Transactions on Computational Socia Systems*, 5(1), pp. 169–175. doi: 10.1109/TCSS.2017.2773581.

[23] FU, Q., FENG, B. , GUO, D. and LI, Q. (2018) Combating the evolving spammers in online social networks. *Computers and Security*, 72, pp. 60–73. doi: 10.1016/j.cose.2017.08.014.

[24] FERRARIO, A. and NAEGELIN, M. (March 1, 2020). The art of natural language processing: classical, modern and contemporary approaches to text document classification. *SSRN Electronic Journal*, Available from: https://ssrn.com/abstract=3547887). doi: 10.2139/ssrn.3547887.

[25] ANITHA, P. U., RAO, C. V. G. and BABU, S. (2018) Email spam classification using neighbor probability based Naïve Bayes algorithm. In: *Proceedings of 7th International Conference on Communaction System Network Technologies (CSNT 2017), Nagpur, India, November 11-13,* Piscataway, New Jersey: IEEE, pp. 350–355. doi: 10.1109/CSNT.2017.8418565.

[26] KARAKAŞLI, M. S., AYDIN, M. A., YARKAN, S. and BOYACI, A. (2019) Dynamic feature selection for spam detection in Twitter. In: BOYACI, A., EKTI, A., AYDIN, M., YARKAN, S. (eds) I*nternational Telecommunications Conference. Lecture Notes in Electrical Engineering*, 504. Springer, Singapore, pp. 239–250. doi: 10.1007/978-981-13-0408-8_20.

[27] WASHHA, M., QAROUSH, A., MEZGHANI, M. and SEDES, F. (2019) Unsupervised collective-based framework for dynamic retraining of supervised real-time spam tweets detection model. *Expert Systems with Applications*, 135, pp. 129–152. doi: 10.1016/j.eswa.2019.05.052.

[28] WU, T., WEN, S., XIANG, Y. and ZHOU, W. (2018) Twitter spam detection: Survey of new approaches and comparative study, *Computers and Security*, 76, pp. 265–284. doi: 10.1016/j.cose.2017.11.013.

[29] BROCARDO, M. L., TRAORE, I. and WOUNGANG, I. (2015) Authorship verification of e-mail and tweet messages applied for continuous authentication.

Journal of Computer and System Sciences, 81(8), pp. 1429–1440. doi: 10.1016/j.jcss.2014.12.019.
[30] CHEN, Ch., ZHANG, J., XIE, Y., XIANG, Y., ZHOU, W., HASSAN, M., ALELAIWI, A., and ALRUBAIAN, M. (2016). A performance evaluation of machine learning-based streaming spam Tweets detection. *IEEE Transactions on Computational Social Systems*, 2(3), pp. 65–76. doi: 10.1109/TCSS.2016.2516039.
[31] AL-AYYOUB, M., JARARWEH, Y., RABAB'AH, A. and ALDWAIRI, M. (2017) Feature extraction and selection for Arabic tweets authorship authentication. *Journal of Ambient Intelligence and Humanized Computing*, 8(3), pp. 383–393. doi: 10.1007/s12652-017-0452-1.
[32] TALHA, A. and KARA, R. (2017) A survey of spam detection methods on Twitter. *International Journal of Advanced Computer Science and Applications*, 8(3), pp. 29–38. doi: 10.14569/ijacsa.2017.080305.

参考文:

[1] ALARIFI，A.，ALSALEH，M.和AL-SALMAN，A. M.（2016）Twitter圖靈測試：識別社交機器。信息科學，37(2)，第332–346頁。 doi：10.1016 / j.ins.2016.08.036。
[2] GUPTA，H.，JAMAL，M。S.，MADISETTY，S. 和 DESARKAR，M. S.（2018）種 Twitter中實時垃圾郵件檢測框架。 於：第十屆國際通信系統與網絡國際會議論文集，班加羅爾，2018年1月3日至7日。新澤西州皮斯卡塔維：IEEE，第380–383頁。 doi：10.1109 / COMSNETS.2018.8328222。
[3] MILLER，Z.，DICKINSON，B. DEITRICK，W.，HU，W。和 WANG，A.H。（2014）使用數據流聚類的Twitter垃圾郵件發送者檢測。情報科學，260，第64-73頁, doi：10.1016 / j.ins.2013.11.016。
[4] ADEWOLE，K. S.，ANUAR，N. B.，KAMSIN，A. 和 SANGAIAH，A. K.（2019）SMSAD：垃圾郵件和垃圾郵件帳戶檢測的框架。多媒體工具和應用，78（4），第3925-3960頁。 doi：10.1007 / s11042-017-5018-x。
[5] LIU S.，WANG，Y.，ZHANG，J.，C. CHEN，XIANG，Y.（2017）使用集成學習解決Twitter垃圾郵件檢測中的類不平衡問題。計算機安全， 69，第35-49頁。 doi：10.1016 / j.cose.2016.12.004。
[6] ZHENG X.，ZENG Z.，CHEN，Z.，YU Y. 和 RONG C.（2015）在社交網絡上檢測垃圾郵件發送者。神經計算，159（1），第27-34頁，doi：10.1016 / j.neucom.2015.02.047。
[7] KAUR，R.，SINGH，S. 和 KUMAR，H.（2018），在線社交網絡中垃圾郵件的增多和受感染帳戶的崛起：不同打擊方式的最新回顧. 網絡與計算機應用學報 , 112，第53–88頁。 doi：10.1016 / j.jnca.2018.03.015。
[8] LABANI，M.，MORADI，F。AHMADIZAR，P. 和 JALILI，M.（2017）一種用於文本分類問題中特徵選擇的新型多元過濾方法。人工智能的工程應用，70，第25-37頁。 doi：10.1016 / j.engappai.2017.12.014。
[9] HOSSEINI，E. S. 和 MOATTAR，M. H.（2019），基於交互信息的進化特徵子集選擇，用於高維不平衡數據分類，“應用軟件計算”，82，第 105581. doi：10.1016 / j.asoc.2019.105581。
[10] CAI，J.，LUO，J.，WANG，S. 和 YANG，S.（2018）機器學習中的特徵選擇：一個新的視角。神經計算，300，第70-79頁，doi：10.1016 / j.neucom.2017.11.077。
[11] N. ALNUAIMI，MASUD，M。M.，SERHANI，M。A.和ZAKI，N。（2019）大數據的流特徵選擇算法：一項調查。應用計

算與信息學，16（1/2），第1-23頁。
doi：10.1016 / j.aci.2019.01.001。
[12] AL-ZOUBI，A. M.，FARIS，H.，ALQATAWNA，J. 和 HASSONAH，M. A.（2018）使用鯨魚優化算法的演進支持向量機，用於在不同語言環境下在線社交網絡上的垃圾郵件配置文件檢測。知識系統，第153頁，第91-104頁。 doi：10.1016 / j.knosys.2018.04.025。
[13] SUCHETHA，N. K.，NIKHIL，A. 和 HRUDYA，P.（2019）比較Twitter情感分類上的包裝器特徵選擇評估器。於：2019年第二屆數據科學計算智能國際會議論文集（ICCIDS）。金奈，2019年10月24日至26日。新澤西州皮斯卡塔維：IEEE。 doi：10.1109 / ICCIDS.2019.8862033。
[14] UÇAR，M。K.（2020），基於分類性能的機器學習特徵選擇算法：P-Score。 IRBM，41（4），第229-239頁，doi：10.1016 / j.irbm.2020.01.006。
[15] SULAIMAN M. A. 和 LABADIN，J.（2015）基於互信息的特徵選擇。於：2015年第9屆亞洲IT國際會議論文集（CITA 2015），砂拉越，馬來西亞古晉，2015年8月4日至5日。新澤西州皮斯卡塔維：IEEE。第1-6頁。 doi：10.1109 / cita.2015.7349827。
[16] ZHOU H.，WANG X. 和 ZHANG，Y.（2020）基於加權條件互信息的特徵選擇。《應用計算和信息學》，16（1/2），提前發布。 doi：10.1016 / j.aci.2019.12.003。
[17] AMIRI，F.，REZAEI M.，YOSEEFI，C.，LUCAS，C.，SHAKERY，A。和 NAZDANI，（2011）針對入侵檢測系統的基於互信息的特徵選擇。網絡與計算機應用學報 34（4），第1184-1199頁。 doi：10.1016 / j.jnca.2011.01.002。
[18] LIN G.，SUN，N.，NEPAL，S. ZHANG，J.，XIANG，Y. 和 HASSAN，H.（2017）神秘的統計Twitter垃圾郵件檢測：性能，穩定性和可伸縮性。IEEE訪問，第5頁，第11142-11154頁。 doi：10.1109 / ACCESS.2017.2710540。
[19] AMIRA，S. 和 GIRDZIJAUSKAS，S.（2017）AdaGraph：基於自適應圖的算法。在：第五屆國際會議論文集（NETYS 2017），摩洛哥馬拉喀什，2017年5月17日至19日。柏林：施普林格，第338–354頁。 doi：10.1007 / 978-3-319-59647-1。
[20] MURUGAN，N。S. 和 DEVI，G。U.（2018）使用混合方法檢測Twitter垃圾郵件流。無線個人通信，103（2），第1353–1374頁。 doi：10.1007 / s11277-018-5513-z。
[21] SHEN，H.，MA，F.，X. ZHANG，L. ZONG，X. LIU，W. LIANG（2016）從多個角度發現社會垃圾郵件發送者。神經計算，225，第49-57頁。 doi：10.1016 / j.neucom.2016.11.013。
[22] SEDHAI，S. 和 SUN，A.（2018）Twitter Strea中的半監督垃圾郵件檢測
[26] KARAKAŞLI，M. S.，AYDIN，M. A.，YARKAN，S. 和 BOYACI，A.（2019）動態特徵選擇，用於Twitter中的垃圾郵件檢測。在：BOYACI，A.，EKTI，A.，AYDIN，M.，YARKAN，S.（eds）國際電信會議上。電氣工程講義，504。施普林格，新加坡，第239–250頁。 doi：10.1007 / 978-981-13-0408-8_20。
[27] WASHHA，M.，QAROUSH，A.，MEZGHANI，M.和SEDES，F.（2019）基於監督的基於集體的框架，用於對監督的實時垃圾郵件推文檢測模型進行動態再訓練。專家系統及其應用，135，第129-152頁。 doi：10.1016 / j.eswa.2019.05.052。

[28] WU，T.，WEN，S.，XIANG，Y. 和 ZHOU，W.（2018）Twitter 垃圾郵件檢測：新方法的調查和比較研究，《計算機與安全》，76，第265-284頁。 doi：10.1016 / j.cose.2017.11.013。

[29] BROCARDO，M. L.，TRAORE，I .和 WOUNGANG，I.（2015）電子郵件和推文消息的作者身份驗證，用於連續認證。計算機與系統科學學報，81（8），第1429–1440頁. doi：10.1016 / j.jcss.2014.12.019。

[30] CHEN, Ch., ZHANG, J., XIE, Y., XIANG, Y., ZHOU, W., HASSAN, M., ALELAIWI, A., 和 ALRUBAIAN, M.（2016）。基於機器學習的流式垃圾郵件Tweets檢測的性能評估。IEEE計算社會系統交易，2（3），第65–76頁。 doi：10.1109 / TCSS.2016.2516039。

[31] AL-AYYOUB，M.，JARARWEH，Y.，RABAB'AH，A。和 ALDWAIRI，M。（2017）阿拉伯語推文作者身份認證的特徵提取和選擇。環境智能與人性化計算雜誌，8（3），第383–393頁。 doi：10.1007 / s12652-017-0452-1。

[32] TALHA，A. 和 KARA，R.（2017）對Twitter上垃圾郵件檢測方法的調查。國際高級計算機科學與應用學報，8（3），第29-38頁。 doi：10.14569 / ijacsa.2017.080305。

Appendix:

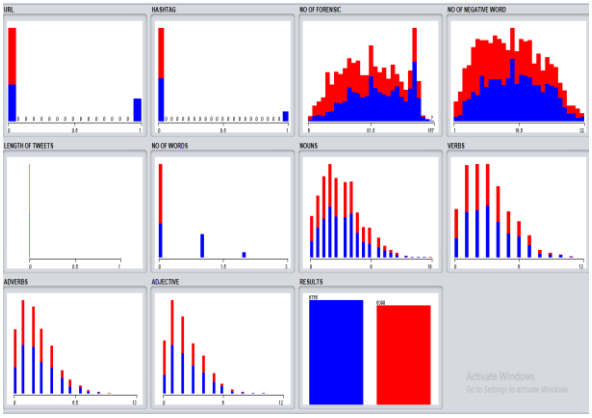

Figure 8: Tweets Dataset