Research Journal of
Applied Science
MEDWELL PUBLICATIONS

Spam Classification by using Naive Bayes Algorithm Based on Segmentation

Shahad Suhail Najam and Karim Hashim AL-Saedi
Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq

Key words: Spam, Naive Bayes, information gain, term frequency invers term frequency

Corresponding Author:
Shahad Suhail Najam
Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq

Page No.: 437-447
Volume: 14, Issue 12, 2019
ISSN: 1815-932x
Research Journal of Applied Sciences

Abstract: One of the greatest methods of communication convenient involves using e-mail for personal messages or commercial objective. Being one of the strongest and quick ways of communication, email's publicity has led to increased undesirable spam email. Email spam is one of the main problems of the Internet today and bringing financial damage to companies and individual users. Spam mails can be harmful as they may contain malware and links to phishing Web sites. So, necessary to separate spam from mail messages into a separate folder. Filter classification can be classified in two techniques-learning method based on machine learning techniques and non-machine techniques. Most popular machine learning techniques due to the high accuracy and athletic support. Machine learning techniques include Naive Bayes and support vector machine learning and decision tree, etc. while non-machine learning techniques, black and white list, signatures and verify email address and mail header checking, etc. In this study utilize one of mechanism learning techniques is Naive Bayes algorithm and for extract features from dataset used Term Frequency Invers Term Frequency (TFIDF) method. For reduce dimensionality of feature space use Information Gain (IG) method.

INTRODUCTION

Spam is unsolicited junk mail sent over the internet. Nowadays spam appears as a new threat on the internet at the email system (Sharaff *et al*., 2015). A Spam email has become an important problem on the internet that the whole industry and individuals are suffering from the effects of this problem. Besides costing recipients additional email management time, spam emails lead to consumption of computing and network resources, on the other side, the network performance indicates to network security problems In other words it has a direct effect on the availability of email (Kagstrom, 2005). People who send unwanted or unsolicited message over the internet are called Spammers. These are sent by commercial advertisers who may offer suspiciously uncomfortable life style or promote unwanted actions, here, the intention is to make email user spend money. There is another type of spammer who sends a large number of e-mails that overflow the user mailbox or mail server, Here, the intention is to damage the email service to such an extent that users cannot receive genuine mails. This is termed as Denial-of-Service (DOS) attack (Anonymous, 2003). For the reasons stated, the spam filter is one of the most newly important security systems which have been used recently. The Spam filter has become of big significance

and necessity. In generally, spam filter focuses on classifying the emails and deleting spam or throwing spam emails into spam folders. Spam filters use different detection techniques such as blacklists, Whitelist, statistical analysis, signature-based filters and keywords, etc. (Jabbar, 2015). The major purpose of this study is to limit the impact of spam on the email system action, for both protection and economics aspects. This would be achieved over the design of a spam filter that works to save the spam out of the mailbox of the users. In this paper used one of machine learning techniques is Naive Bayes algorithm for spam filter. The term "Naive Bayes Classifier" workbook simple probability based on Bayes theorem with strong independence hypothesis. It assumes that the class-specific feature the presence or absence of Unrelated by advantage presence or absence. "Naive Bayes" algorithm based on conditional probabilities. This algorithm uses Bayes theorem, a formula that calculates the probability calculation of frequency values and combinations of values in historical data. Baye's theorem to calculate the probability of an event occurring when you give the most likely event that happen earlier (Abdalgafore, 2015). Section two displays some of related work which covers the problem of a spam. Section three presents a background on spam. Section four presents the methodology of the proposed system. The evaluation of results is discussed in section five. The last section presents the conclusion.

Related work: There are many kinds of literature on spam email, these are: Awad and Elseuofi (2011), the proposed system used the most popular machine learning methods (Naive Bayesian classification (NB), k-Nearest Neighbors algorithms (kNN), Artificial Neural Network (ANN), Support Vector Machine (SVM), Artificial Immune System (AIS) and Rough Sets (RS) and of their applicability to the problem of spam email classification. In this research Spam Assassin dataset was used which contains 6000 emails with the spam rate 37.04%, divided dataset into training and testing sets, training set consist of 62.96% of the original set while each test set consists of 37.04%. The experiment is performed with the most frequent words in spam email; select 100 of them as features. Table 1 summarizes the results of the six classifiers by selecting the top 100 features.

Table 1: Results of the six classifiers

Algorithm	Spam recall	Spam precision	Accuracy
NB	98.46	99.66	99.46
SVM	95.00	93.12	96.8
KNN	97.14	87.00	96.20
NN	96.92	96.02	96.83
AIS	93.68	97.75	96.23
RS	92.26	98.70	97.42

Table 1 shows the Naïve Bayes method as the most accurate while the artificial immune system and the k-nearest neighbor give us approximately the same lower percentage while in terms of spam precision it can be observed that the Naïve Bayes method has the highest precision among the six algorithms while the k-nearest neighbor has the worst precision percentage and surprisingly the rough sets method has a very competitive percent and finally find that the recall is the less percentage among the six classifiers while the Naive Bayes still has the highest performance but considered low when compared to precision and accuracy while the rough sets has the worst performance.

Chakraborty and Patel, they discussed how email became one of the most common methods of communication among individuals because of its cheapness and speed. They proposed using hybrid system that consists of (ANN) and (NB) as a spam filtering system to enhance the spam filter.

Kumar and Kumar, this paper proposed malicious mail detection system through the (Supervised learning) classification approach. This proposed to handle the spam and phishing emails based on the Meta-data characteristics of email. This proposed present filtering and detection techniques are performing well under detection of targeted malicious mails detection. Next another step is present for detection of persistent and recipient oriented attacks. The result from this work with an efficient probability based supervised learning approach by classifying the testing dataset.

Divya and Kumaresan, (2014), they presented a spam classifier using machine learning algorithms including NB, SVM and kNN was also proposed. The dataset used was spam assassin which contains 6000 emails 3776 of which for training and 2224 emails for testing. The numbers of features used was 100 features. In addition, to the body of an email message, the classification based on other fields of the email such as the subject and the form. The performance evaluation recorded for the three classifiers was: (NB: Accuracy = 99.46), (SVM: Accuracy = 96.90), (kNN: Accuracy = 96.20). On the contrary of the previous studies, NB gave a satisfying performance among the other learning methods.

Yang *et al*. (2015), proposed comparison between using both association rule and Naive Bayes classifier algorithms and just using Naive Bayes classifier for purpose of spam filtering. In this study, applied association rules and Naive Bayes on Enron-spam dataset. It proposed to use map reduce program to handle the amount of words. Map reduce approach is to use <key, value> pairs and the groups that will be received in the reduce function will be grouped by the key:

$$\text{Map}:<K_1, V_1> \rightarrow \text{list} < K_2, V_2 >$$

$$\text{Reduce}:<K_2, \text{list (V2)} > \rightarrow \text{list } (K_3, V_3)$$

To enhance implementation to combine Naive Bayes classifier and A Priori algorithm, the purpose for the enhancing implementation is to improve ham precision rate.

Background

Electronic mail: Electronic mail (email) is the most popularly used for sending messages between users on the Internet (or any other computer network) (Wanroij, 2000). The next section will present the two main parts of email (email structure, email threats).

Email structure: The basic form for email commonly consists of the following two parts: section header contains the e-mail address of the sender and the recipient's email address, the subject or the timestamps that appear when the message was sent by intermediary servers to transport agents (MTAs) which operates as a mail sorting office. An email containing at least three headers:

- From: email address of the sender
- To: The recipient's email address
- Date: date when the e-mail has been sent

May contain following optional fields:

- Received: a variety of information about the broker's servers and date when the message has been processed
- Reply-To: reply-to address.
- Subject: subject of the message
- Message-ID: A unique identification for the message

Email content includes main text consists of text, images and other multimedia data, separated by a line break. An e-mail message consists of lines of 7-bit US-ASCII characters are viewable. Each line in most 76 characters, for compatibility reasons, and ends with CRLF (\r\n) (Malarvizhi and Saraswathi, 2013).

General characteristics of spam: Email spam, also known as Unsolicited Bulk Email (UBE) or spam or unsolicited commercial e-mail (UCE) is the practice of sending unsolicited emails, frequently with commercial content, a large sum Random group of recipients. Spam on the internet because the transaction cost of electronic communications is radically less than any form of alternative forms of communication (Blanzieri and Bryl, 2008). E-mail spam is one of the main problems of the internet today, gets financial damage to companies and individual users. More accurately, the reason spam traffic abuse and storage space and computing power; spam makes users through search and additional sorting e-mail, not just wasting their time and cause loss of work productivity but also chagrined as many claims that violated privacy rights Their own; in the end, the spam cause legal problems with pornographic ads, pyramid schemes, etc. (Youn and McLeod, 2007).

Spam different kinds: There are several kinds of spam email as following: phishing, is fishing for sensitive information (passwords, credit card numbers, etc.) official requests tradition of reliable references such as banks and manage the server or service provider. This activity can be done using the company's characteristics: such as the company's logo and similar fonts and colors.

Sometimes also use a huge attack from spam to disable mail server. Junk mail, cluster mailings from legitimate companies that are undesirable. Offensive spam and pornography, mass mailings of ads adult or pornographic images. Virus spam, cluster mailings that that contains malicious script viruses and Trojans.

Online pharmacy spam which is the spam that promotes different versions of pills of antidepressant that can be bought online. Pirate software spam, offers pirate software that is usually much cheaper than the official software prices. Penny stock spam which is a stock-encouraging spam that encourages people to purchase cheap

Bottom line, the sender of the spam message relay following tasks: announcing some goods or services or ideas, to receive their information users, to deliver malware, or cause temporary suspension of mail server (Tanta-ngai *et al*., 2003; Youn, 2014; Anonymous, 2003).

Spammer: People who send spam emails to make money from others are called spammers. Also, send the spammers of other species, including newsgroups, instant messages and web publications Board and even exploit the services like Windows messenger get through advertising messages or obnoxious (Youn and McLeod, 2007; Anonymous, 2003). Spammers obtain email addresses by a number of means:

Spammers "harvest" techniques to collect addresses from using the net transfers or DNS entries or Web pages, and common names in specific areas (known as a dictionary attack to guess). "Pending" or search for matching email addresses for some people such as region.

Spammers many use programs called spiders Web to find email addresses on Web pages, although it is possible to trick a spider web by replacing the "@" symbol with another symbol, for example, "#" while posting an e-mail address.

As a result, users have to waste their time value to delete spam. Moreover, because the spam emails can fill quickly file server storage space, it can cause a very serious problem for many sites with thousands of users (Youn, 2014).

Spam filters: Spam filtering is the process used to detect unsolicited emails and spam and blocking access to the user's Inbox. There are two levels can work in spam filter emails that will involve user-level or Enterprise level. Individual users refers to a particular person and works at home and who have been receiving and sending Internet e-mail messages, these users if he wished to learn and install a filter spam messages simply spam filtering system. In spam filter company filter messages during the time entry in the enterprise intranet. Enterprise spam filter, spam filter software installed on the main mail server and is supposed to interact with the Mail Transfer Agent (MTA) which classifies the message at the moment of receipt of Bansal and Bhatia (2017). There are many stages for designing an email filter as following.

- Developing email corpus
- Splitting the content into words and special characters (tokenization)
- Stemming
- Feature selection
- Classification (Beyrami *et al*., 2013)

The filter classification techniques are basically classified into two parts:

- Based on machine learning technique
- Based on Non-Machine Learning Techniques

The machine learning techniques contain Naive Bayes, Support Vector Machine, Neural Network, Decision Tree etc. while the Non-Machine Learning techniques include Bayesian analysis filter, Black/White List, Signatures, Mail Header Checking etc. (Bansal and Bhatia, 2017; Najam and AL-Saedi, 2018).

Feature selection: Select feature can be defined as a process that selects the minimum subset of features of the original group of N features where it is reduced in size to optimal advantage according to some evaluation standard. Feature selection methods are classified into three classes of selection techniques (Fig. 1).

Filter method: It is feature selection using the calculation of weight which may be the relationship between features and class and then choosing features having weight higher than some specific threshold. The algorithms in this category include information gain and Chi-square. Figure 2 shows the process of filter method (Kaoungku *et al*., 2017).

Wrapper method: It is the feature subset selection in which the subset generation and the learning algorithm are wrapped inside the same module. The subset selection steps can be iterative for the best model creation yielding high classification accuracy. Figure 3 shows the process of wrapper method (Lei and Liu, 2004; Xie *et al*., 2009).

Fig. 1: Process of filter method (Kaoungku *et al*., 2017)

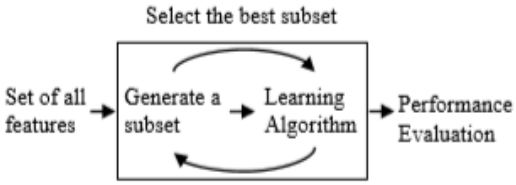

Fig. 2: Process of wrapper method (Kaoungku *et al*., 2017)

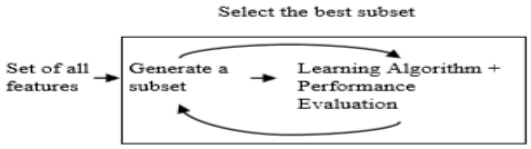

Fig. 3: Process of embedded method (Kaoungku *et al*., 2017)

RootW	Word
strong	stronger
take	taken
bemuse	bemused
rate	rated
value	valuable
item	items
cost	costs
try	tries
world	worldly
modify	modified
record	records
cover	covering
work	works
lead	leader
start	starting
list	listed
artist	artists
remove	removed
modify	modifier
finger	fingered
click	clicking
generate	generator
edit	editor

Fig. 4: Stemming process

Embedded method: It is the feature selection that is part of classification. It is advantage combination for both filter and wrapper methods by selecting features together with creating model. Figure 4 shows the process of embedded method (Xie *et al*., 2009).

Filtering approach as well is classified into two types: supervised methods and unsupervised methods. The supervised methods include many algorithms for features selection such as IG, MI, WF, χ^2 statistic (χ^2) and other algorithms. Unsupervised methods include the Document Frequency (DF), Inverse Document Frequency (IDF), Collection Frequency (CF), Inverse Collection Frequency (ICF) and others (Awad and ELseuofi, 2011; Al-Saedi, 2018).

MATERIALS AND METHODS

The proposed system in this study, titled "spam detection by using Naive Bayes based on segment". The objectives of using this system are to emails classification into spam email or non-spam email. This system consist of three modules; preprocessing modules, training modules, testing modules. Each of these modules contains several sub modules and components.

Preprocessing module: This first module in the proposal system; its main goal is to remove any redundant data cleansing to remove any kind of noise and fake or missing value of data. This module used to facilitate for classification emails. This module consists of three components. Will explain each component of this module in details. Algorithm1 shows work preprocessing modules.

Algorithm 1; Preprocessing:

Input
(E_1, E_2 … E_N,) //All emails in the dataset ((body of email)
Output
(PE_i) //Preprocessing all emails in dataset
BEGIN
Step1:
For E_i do
1- Temp←T(E_i) // Tokenization
2- Temp←(sw(Temp)) //stop word removal
3- Temp←(s(Temp)) // stemming
4- P_{Ei}←Temp
End for
Step 2:
ForPE_i do;
WF_CP_E←Freq-count (PE_i)// for the total no of frequencys
End for
End

Tokenization: Tokenization is the first component in the preprocessing module. The aim of this process is splitting a stream of text into words, number, symbol and character etc. typically tokenization process is on the email body into series of features. Example (1) shows how tokenization works.

Example (1) sample of body emails: People now the weather or climate in any particular environment can change and affect what people eat and how much of it they are able to eat.

After applying tokenization process on the example above than the output is formed like

Output:

Words:[people<2>, now<1>, the<1>, weather<1>, or<1>, climate <1>, in<1>, any<1>, particular<1>, environment<1>, can<1>, change<1>, and<2>, affect<1>, what<1>, people<1>, eat<2>, how<1>, much<1> ,of<1>, it<1>, they<1>, are<1>, able<1>, to<1>].

The value after each word represented the frequency of a word in the given email.

Stop word removal: Stop words removal is the second component of the preprocessing module. The aim of this process is stop words like "the", "are", "with", "and", "to" etc. these need to be removed because these stop words does not load any useful information for helping to determine whether a mail message belongs to classify or not. This component is very important in the preprocessing module because it has some advantages which will lead to reduce the size of email words. When is applying stop word removal process after the previous component (tokenization) on the same text email which was used in the example (1) the following result will be obtained.

Output: Words: [people, weather, climate, particular, environment, change, affect, people, eat].

Stemming: Stemming is the third component of preprocessing module. The aim of this process is used to extract the root of words, i.e., called as stem/root, the given for example added, adding, additionally all these words come back to the root of the word add. The main objective of stemming is to remove different extensions from the words, this will lead to reduce number of words and reduce storage requirement. In this proposed system, dictionary method for this process and specific database for the dataset is used. Figure 5 displays a sample of stemming process of database from the dataset used.

Training dataset module: Training dataset is the second module of the proposed system. The main goal of this

Class	A	B	C
3	product	code	document
3	contain	loan	congratul
3	send	creat	credit
5	visa	refer	particular

Class	A	B	C
0	0	0	0
0	0.03922371105...	0	0
0	0.04553557259...	0	0
0	0.02770346025...	0	0
0	0.03094995950...	0	0
0	0	0	0
0	0.00365460776...	0	0
0	0.03922371105...	0	0

Fig. 5: Show result from (T_FID_F)

module is to prepare the database which is obtained from the previous module to be easy classification dataset in the testing module. A training set is implemented to build up a system. This module consists of two component and sub module.

Feature extraction: This is the first component of training data set module. The aim of this component is data representation because it is very hard to do calculations with text data. The representation should have to reveal the actual statistics for text data. It should be a representation of the data in a way that even the actual statistics are converted to text data to value. Moreover, it should facilitate the classification and functions also simple enough to implement. In this proposed systems, Term Frequency Invers Document Frequency (T_FID_F) for feature extraction to extract the distinctive features of spam and non-spam based dataset are used. The first part (t, d) is simply used to calculate the number of items of each word appeared in each email. The second part (N, n) is the inverse ratio of emails containing the term t and total number of e-mail messages in the dataset, the implementation is illustrated in the Algorithm 2 and show result in Fig. 5.

Algorithm 2; T_FID_F:

Input
(E_1, E_2, …, E_N) // All email from dataset
Output
F//Extraction features from all emails in dataset based on (T_FID_F)
Begin
Step 1:
For E_i do// Compute for each email from dataset
T_F←(T_w, e)(N_w, e), ..., (T_w, e)/ // Number of times the word appears in email
(N_w, e) //Total number of words in the email
End for
Step 2:
for E_i do// Compute ID_F for each email from dataset
ID_F←Log N/n_i, …., N // Number of emails
n_i//Number of email that contain word
End for
Step 3:
For E_i do// Compute T_FID_F for each email in dataset
T_FID_F←T_F*log N/n_i
End for

Feature selection by using (IG Algorithms): Feature selection is the second component of training module. Feature selection applies to taking only useful subset of features without changing their original forms. Feature selections have several techniques. In this proposed system, using IG algorithms are calculated for every single feature (feature list) weighted regarding the email message in the training dataset. Algorithm 3 outlines the steps of calculating IG per each feature and show result in Fig. 6.

No	WFeatur	Appe. P(WFeatur)	Appe. P(Spam /WFeatur)	Appe. P(Non Spam / WFeatur)	Abse. P(WFeatur)	Abse. P(Spam / WFeatur)	Abse. P(Non Spam / WFeatur)	IG
1	softwar	0.371429	0.615385	0.384615	0.628571	0.318182	0.681818	0.060979
2	file	0.314286	0.181818	0.818182	0.685714	0.541667	0.458333	0.087969
3	licens	0.314286	0.545455	0.454545	0.685714	0.375000	0.625000	0.018350
4	microsoft	0.114286	0.000000	1.000000	0.885714	0.483871	0.516129	0.100179
5	applic	0.371429	0.230769	0.769231	0.628571	0.545455	0.454545	0.070936
6	includ	0.428571	0.333333	0.666667	0.571429	0.500000	0.500000	0.020244
7	other	0.457143	0.437500	0.562500	0.542857	0.421053	0.578947	0.000198

Fig. 6: Show result from (IG)

Algorithm 3; Feature selection by IG:

Input
F// all features extraction from T_FID_F
Output
Features based on IG
BEGIN
Step 1:
For C_i do // Compute probabilities for each class(spam , non -spam)
P(c)←Frequency (c)/N, …, Frequency (c) // number of class (spam, non-spam) email
N // total of email
End for
Step2:
a) For F_i do //compute probabilities appearance each feature in all mail
P (F) ←f_i/N, …., P(F) // probabilities for each features
f_i // appearance of features in all mails
N// total of mails
End for
b) For Fi do //compute probabilities appearance each feature in email class
P(c|F) ←cf_i/Nf_i,, …., cf_i // appearance of feature in each class (spam, non-spam) email
Nf_i // appearance of feature in all mails
End for
Step 3:
a) For F_i do //compute probability absence foreach features in all mails
P(F^-)←f_i^-/N, …, f_i^-// absence offeatures in all mails
N// total of emails
End for
b) For F_i do //compute probability absence for each features in each class email
P(C|F^-|)←cf_i^-/Nf_i^-, …., cf_i^-//absence of feature in each class mails
Nf_i^-// absence of feature in all mails
End for
Step 4:
For F_i do //Compute entropy
a)Total – Entropy← - p (c)*$\log_2$ p(c)
b) Feature_ Entropy← -p (c)*$\log_2$ p(f)
End for
Step 5:
For F_i do //Compute IG
IG←total Entropy-feature-Entropy

Segmentation process for the features:

After applying (IG) algorithm for all features, a number of segmentation of these features will work. The aim of this step is to reduce the complexity when Naive Bayes applied for classification. Segmentation application depended on weighted for feature from applied IG. Each

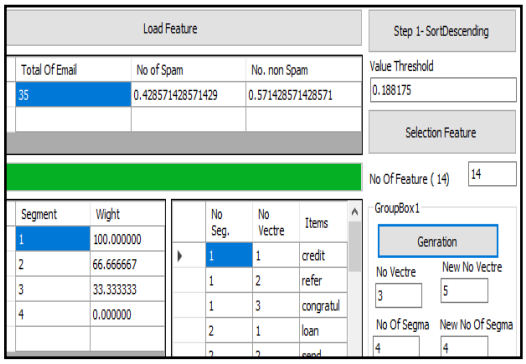

Fig. 7: Offer generation for sample segments and weighted for each segment

segment consists of a specified number of features depending on the threshold which got from IG. If the number of features is larger than the number allocated in each segment, then the number of remaining features is very small and adds these features in the last segment but the number of remaining large feature will add the new segment (this process operates according to certain threshold). The benefit from the process of segmentation is obtained in order to facilitate the work of the algorithm and obtain a high evaluation. The implementation of segmentation is shown in Algorithm 4.

Algorithm 4; Segmentation:

Input:
IG features based on specific threshold ,
N_w //number of features that apply the threshold value
$N_{collected}$ // the numberof specified segment
X_1// the number of features in each segment
X_0// the number of remaining features after segment
Output:
S_{IG} // segment based on IG features
Step1:
$X_1 \leftarrow N_w / N_{collected}$
$X_0 \leftarrow N_w \bmod N_{collected}$
If X_0 <> 0then
If X_0> 3 then
If ($N_{collected}$ * X_1+1) <$N_w$then
Add seg_1
Else
Add features in the least segment
Return S_{IG}
End.

In the segmentation phase it depends on threshold for Information Gain value (IG) and numbers of words for each segment and calculated weighted for each segment as shown in Fig. 7. After that calculated weights for each feature in each segment show this in Fig. 8.

Naive Bayes (NB) module: This is a sub module of training dataset module in this proposed system. This sub module applied NB algorithm belongs to classification algorithms to training dataset. The result of the algorithms

No	Seg	alph	Featur	Wight	Mach
1	1	A	credit	0.0000	
2	1	B	refer	50.0000	
3	1	C	congratul	25.0000	
4	1	D	loan	0.0000	
5	1	E	send	25.0000	True
6	1	F	creat	0.0000	
7	1	G	code	0.0000	

Fig. 8: Weights for sample features of one segment

predicts the incoming email into spam or non-spam. This algorithm includes two phases: training phase and testing test, both of them depend on the email content which is represented by features. Algorithm 5 outline the steps of calculating Naive Bays (NB) algorithm.

Algorithm 5; NB classifier:

Input: features
Output: classification based on NB , probability (spam , non-spam)
Begin
/* Training */
Spam Probability←No. of spams in training dataset/The total No. of emailsin training dataset
N Spam_ Probability←No. of n spam in training dataset /The total No. of –email in training dataset
For F_l in Training dataset do
Spam_ Probability (F)←Spam _count (F)/Spam count
N Spam_ Probability (F)←N Spam_ count (F)/N Spam_count
End For
/* Testing */
For F_l in Testing dataset do
If value of (F) is Found in training phase then do
Spam _Probability←Spam_Probability *Spam _Probability (F)
Non-Spam_ Probability←Non- Spam_ Probability *Non-Spam _Probability(F)
Else
Get two nearest feature probabilities
Get the average of these features
Spam _Probability←Spam _Probability (F) of the average
Non-Spam_Probability←Non-Spam _Probability (F) of the average
End if
End For
Spam_Probability←Spam _Probability*Spam_ Probability
Non-Spam_Probability←Non-Spam _Probability* Non Spam _Probability
End For
Spam_Probability←Spam_Probability*Spam_ Probability
Non-Spam_ Probability←Non-Spam_ Probability* Non-Spam_ Probability
If Spam_ Probability> Non-Spam_ Probability then do
Class = Spam
Else
Class = Non-Spam
End if
/* Evaluating NB Classifier */
Find Evaluation parameter (True Positive, True Negative, False Positive and False Negative)
Find Accuracy, Error Rate, Precision and Recall.

Training phase: This phase attempts to calculate the following probabilities: Calculate the probability of Spam email class and non-spam email class to the total number of email sample using equation as below:

$$P(\text{spam}) = \frac{\text{No. of spam word}}{\text{No. of words}}$$

$$P(\text{non-spam}) = \frac{\text{No. of spam word-spam word}}{\text{No. of words}}$$

Calculating the P (C_i) probability of certain sample email (X) being in either of two classes. This is done in terms of calculating probability of occurrence of individual feature Fi in either class using equation as below:

$$P\left(\text{word/spam/non}-\text{spam}\right) = \frac{nk+1}{n+\text{vocabulary}}$$

Where:

nk = Number of occurrences of a specific word in spam/non-spam emails
n = Number of words in spam/non-spam emails
Vocabulary = Number of distinct word in all training dataset

Testing phase: Using the probabilities gained through the training phase, test phase process in the email form in the test as follows:

Calculate probabilities of each email sample X (in terms of each value for each features F) for both classes, on the basis of probabilities from training phase.

If certain value for some features F, does not show, for X, during the training phase, then set that value to the average probability of two closest features values of x. For each sample X, find the posterior probabilities using "Bayes theorem" using equation as below:

$$P(c|X) = \frac{P(X|c)P(X)}{P(X)}$$

Decided the class of X based on result from the previous phase (c) The class would be the one with largest probability for x.

Testing phase for each segment: Using the probabilities gained through the training phase, test phase process in the email form in the test as follows: this phase consists of two cases.

Case 1:

- For each segment return spam or non-spam by using NB
- Find for each segment calculated probability for spam and probability for non-spam classified based on spam or non-spam. The implementation of this case in Algorithm 6

Algorithm 6; NB testing for each segment:

Input: Seg, feature
Output: classify
Begin
Step 1:
For each seg_i do
Get w Feature in seg_1 *//saving from training*
wP_0 , P_1 , classify←Call algorithm(3.5) NB classifier (f)
If classify = spam then Sum_ spam +=1
Else
If classify =non-spam *then*Sum_Non- spam+=1
Else
New Email +=1
End for
Step 2:
If Max (Sum_ spam, Sum_ Non- spam) then Return spam
Else If Sum_ spam=Sum_ Non- spam then Return New Email
Return non-spam
End

Case 2:

- For each segment return probability for spam and non-spam based on NB
- Determining max of probability spam and max of probability non-spam
- Determining spam or non-spam. The implementation of this case in Algorithm 7

Algorithm 7; NB testing for each segment:

Input: Seg, feature
Output: classify
Begin
Step 1:
For each seg_i do
Get w features in seg_1 //saving from training
W features, $P_{(0)}$,$P_{(1)}$, classify←Call algorithm (3.5) NB classifier(feature)
ArrayP0 add (P_0)
Array P_1. add (P_1)
End for
Step 2:
X_1←Max (Array P_1)
X_2←Max (Array P_0)
Step3:
IfX_1>X_2 then return spam
Else
If X_2<>X_1 then return New email
Else
Return non spam
End

Dataset: The dataset is used to evaluate the performance of the proposed system in order to validate the proposed system. The dataset used in the proposed system is the Enron dataset which is divided into two parts: spam and non-spam. Table 2 shows the basic dataset statistics to be used. The utilized dataset are discussed in the following sections.

Division the dataset into two phases: training and testing, training phase have 3620 emails and testing phase have 2551 emails Fig. 9 illustrate this division.

Spam detection evaluation

Accuracy: This is the ratio between the sum of true prediction was divided by the total emails. Table 3 show accuracy result for experiment.

Error rate: This is the ratio between the sum of false prediction was divided by the total E-mails. Table 4 show accuracy result for experiment.

Precision: This is the ratio between the sum of true positive was divided by the total number of positive prediction. Table 5 show precision result for experiment.

Recall: This is the ratio between the sum of true positive was divided by the total number of true positive and false negative prediction. Table 6 show precision result for experiment. Comparisons between previous related works with proposed system in this study Table 7 illustrate compering.

Table 2: Details datasets to be used

Dataset	Totals of emails	Number of spam emails	Number of non-spam emails
Enron dataset	5172	1500 emails	emails

Table 3: Accuracy results on testing sample

Threshold value	No. features	No. segments	Naive Byes (%)
0.013568	100	33	92
0.02162	57	19	94
0.011917	114	16	92

Table 4: Error rate results on testing sample

Threshold value	No. features	No. segments	Naive Byes (%)
0.013568	100	33	93
0.021627	57	19	92
0.011917	114	16	93

Table 5: Precision results on testing sample

Threshold value	No. features	No. segments	Naive Byes (%)
0.013568	100	33	93
0.021627	57	19	92
0.011917	114	16	93

Table 6: Recall results on testing sample

Threshold value	No. features	No. segments	Naive Byes (%)
0.013568	100	33	93
0.021627	57	19	92
0.011917	114	16	93

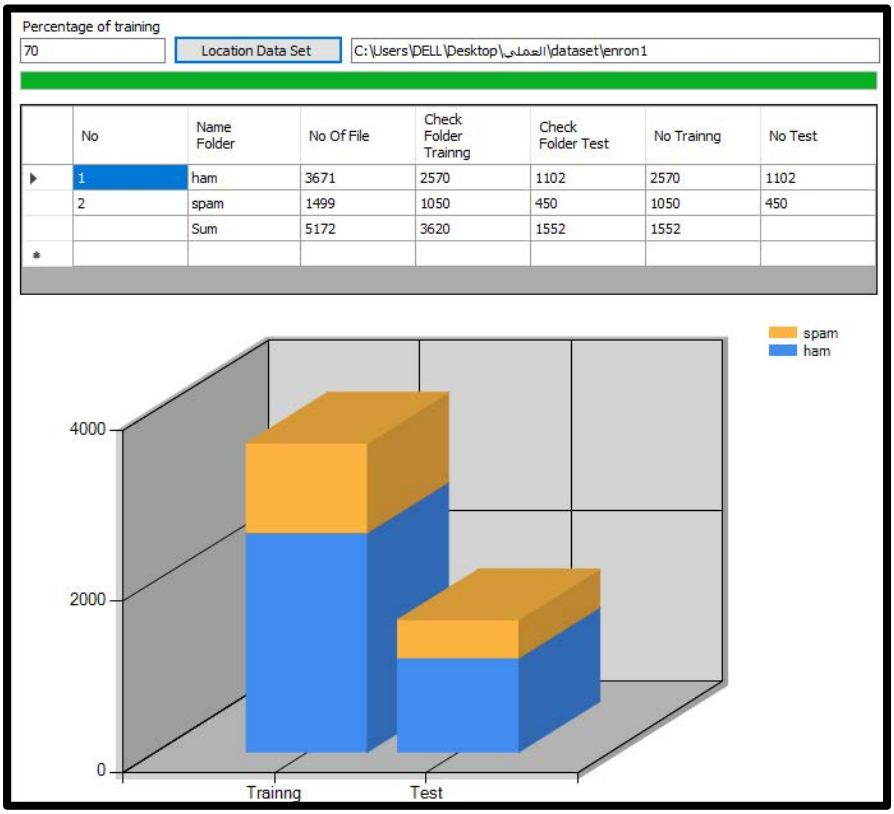

Fig. 9: Results of training and testing for emails

Table 7: Comparing

Researchers	Techniques used	Dataset used	Evaluation measure
Tianda yang, Kai Qian *et al*.	Used association rule and Naive Bayes Using just Naive Bayes Used map reduce approach	Enron dataset	Precision rate 91.96%
S. Divya and T. Kumaresan	NB, SVM, KNN The number of features used was 100 features In addition to the body of an email message, the classification was based on other fields of the email such as subject and the form	The dataset used was spam assassin dataset which contains 6000 emails	The Naive Bayes have a high accuracy from other algorithms = 92.55
Our proposed system	Used Naive Bayes algorithm Used segmentation method for features depended on weights from IG Used Naive Bayes based on segmentation	Enron dataset	Accuracy depended on threshold value, so, threshold (0.011917) have a high accuracy = 93%

CONCLUSION

After building email spam filter for detecting the spam and non-spam emails the following list are concluded: The implementation of the operation of the segment based on the information gain algorithm; this is because the work of information gain based on the selection of features that have high weights values. The implementation of NB algorithm based on segment two new methods is used in the testing phase. Classifiers performance is enhanced with a bigger training sample size.

SUGGESTIONS

The proposed work can be extended to deal with new features such as attachments and images included in the email messages. Design and implement of online system to work on the server instead of the offline system. Any new incoming email message can be automatically classified. Design and implement of mobile Application to classify incoming a new SMS automatically.

REFERENCES

Abdalgafore, H.A., 2015. A hybrid spam E-mail filtering system using ant colony optimization and Naive Bayesian. MSc Thesis, University of Technology Sydney, Ultimo, Australia.

Al-Saedi, K.H., 2018. Toward mining salient attribute based on developing principle component analysis algorithm. J. Eng. Appl. Sci., 13: 5890-5896.

Anonymous, 2003. Spam: A security issue. Cipher Trust, Nuremberg, Germany.

Awad, W.A. and S.M. ELseuofi, 2011. Machine learning methods for E-mail classification. Intl. J. Comput. Appl., 16: 39-45.

Bansal, A. and P.K. Bhatia, 2017. A survey of various machine learning algorithms on email spamming. Intl. J. Adv. Electron. Comput. Sci., 4: 82-87.

Beyrami, N.T., M. Razaa and F. Derakhshi, 2013. The feature selection and dimensionality reeducation methods for email classification. J. Basic. Appl. Sci. Res., 3: 633-636.

Blanzieri, E. and A. Bryl, 2008. A survey of learning-based techniques of email spam filtering. Artif. Intell. Rev., 29: 63-92.

Chakraborty, N. and A. Patel, 2012. Email spam filter using Bayesian neural networks. Intl. J. Adv. Comput. Res., 2: 65-69.

Jabbar, S.F., 2015. A spam email classifier based on naive Bayesian approach. MSc Thesis, Al-Mustansiriyah University, ?Baghdad, Iraq.

Kagstrom, J., 2005. Improving naive bayesian spam filtering. Master Thesis, Department for Information Technology and Media, Mid Sweden University, Sweden.

Kaoungku, N., K. Suksut, R. Chanklan, K. Kerdprasop and N. Kerdprasop, 2017. Data classification based on feature selection with association rule mining. Proceedings of the International Multi Conference on Engineers and Computer Scientists Vol. 1, March 15-17, 2017, IMECS, Hong Kong, ISBN:978-988-14047-3-2, pp: 1-6.

Lei, Y. and H. Liu, 2004. Efficient feature selection via analysis of relevance and redundancy. J. Mach. Learn. Res., 5: 1205-1224.

Malarvizhi, R. and K. Saraswathi, 2013. Content-based spam filtering and detection algorithms-an efficient analysis and comparison. Intl. J. Eng. Trends Technol., 4: 4237-4242.

Najam, S.S. and K.H. AL-Saedi, 2018. Spam classification by using association rule algorithm based on segmentation. Intl. J. Eng. Technol., 7: 2760-2765.

Naseriparsa, M., A.M. Bidgoli and T. Varaee, 2014. A hybrid feature selection method to improve performance of a group of classification algorithms. Comput. Sci., 69: 28-36.

Sharaff, A., N.K. Nagwani and K. Swami, 2015. Impact of feature selection technique on email classification. Intl. J. Knowl. Eng., 1: 59-63.

Tanta-ngai, H., T. Abou-Assaleh, S. Jiampojamarn and N. Cercone, 2003. Secure mail transfer protocol (SecMTP). Proceedings of the International Conference on Advances in the Internet Processing Systems and Interdisciplinary Research IPSI-2003, October 5-11, 2003, Sveti Stefan, Montenegro, Budva, Montenegro, pp: 1-6.

Wanroij, R.V., 2000. Secure electronic mail. MSc Thesis, University of Twente, Enschede, Netherlands.

Xie, J., J. Wu and Q. Qian, 2009. Feature selection algorithm based on association rules mining method. Proceedings of the 2009 8th IEEE/ACIS International Conference on Computer and Information Science, June 1-3, 2009, IEEE, Shanghai, China, ISBN:978-0-7695-3641-5, pp: 357-362.

Yang, T., K. Qian, D.C.T. Lo, K. Al Nasr and Y. Qian, 2015. Spam filtering using association rules and Naive Bayes clas-sifier. Proceedings of the 2015 IEEE International Conference on Progress in Informatics and Computing (PIC), December 18-20, 2015, IEEE, Nanjing, China, ISBN:978-1-4673-8086-7, pp: 638-642.

Youn, S. and D. McLeod, 2007. Spam email classification using an adaptive ontology. J. Software, 2: 43-55.

Youn, S., 2014. SPONGY (SPam ONtoloGY): Email classification using two-level dynamic ontology. Sci. World J., 2014: 1-11.