Spam detection in email through comparison of different classifiers

PG Scholar Tejal Rajesh Girase, Mr. Kailash Patidar, Mr. Rishi Kushwaha, Mr. Manoj Yadav
SOE, SSSUTMS, Sehore Bhopal
tej10rajput@gmail.com

Abstract:-

The increasing volume of unsolicited bulk e-mail (also known as spam) has generated a need for reliable anti-spam filters. Machine learning techniques now days used to automatically filter the spam e-mail in a very successful rate. In this paper we review some of the most popular machine learning methods (Naïve Bayesian classification, SVMs, Logistic Regression, R-Algorithm) and of their applicability to the problem of spam Email classification. Descriptions of the algorithms are presented, and the comparison of their performance on the Ling Spam corpus and Anron Dataset is presented.

Keywords: email, spam, SVM, Naive Bayes, dataset

1. Introduction

Recently unsolicited commercial / bulk e-mail also known as spam, become a big trouble over the internet. Spam is waste of time, storage space and communication bandwidth. The problem of spam e-mail has been increasing for years. In recent statistics, 40% of all emails are spam which about 15.4 billion email per day and that cost internet users about $355 million per year. Automatic e-mail filtering seems to be the most effective method for countering spam at the moment and a tight competition between spammers and spam-filtering methods is going on. Only several years ago most of the spam could be reliably dealt with by blocking e-mails coming from certain addresses or filtering out messages with certain subject lines. Spammers began to use several tricky methods to overcome the filtering methods like using random sender addresses and/or append random characters to the beginning or the end of the message subject line [1]. Knowledge engineering and machine learning are the two general approaches used in e-mail filtering. In knowledge engineering approach a set of rules has to be specified according to which emails are categorized as spam or ham. A set of such rules should be created either by the user of the filter, or by some other authority (e.g. the software company that provides a particular rule-based spam-filtering tool). By applying this method, no promising results shows because the rules must be constantly updated and maintained, which is a waste of time and it is not convenient for most users. Machine learning approach is more efficient than knowledge engineering approach; it does not require specifying any rules [3]. Instead, a set of training samples, these samples is a set of pre classified e-mail messages. A specific algorithm is then used to learn the classification rules from these e-mail messages. Machine learning approach has been widely studied and there are lots of algorithms can be used in e-mail filtering.

2. Machine Learning in E-Mail Classification

Machine learning field is a subfield from the broad field of artificial intelligence, this aims to make machines able to learn like human. Learning here means understood, observe and represent information about some statistical phenomenon. In unsupervised learning one tries to uncover hidden regularities (clusters) or to detect anomalies in the data like spam messages or network intrusion. In e-mail filtering task some features could be the bag of words or the subject line analysis. Thus, the input to e-mail classification task can be viewed as a two dimensional matrix, whose axes are the messages and the features. E-mail classification tasks are often divided into several sub-tasks. First, Data collection and representation are mostly problem specific (i.e. e-mail messages), second, e-mail feature selection and feature reduction attempt to reduce the dimensionality (i.e. the number of features) for the remaining steps of the task. Finally, the e-mail classification phase of the process finds the actual mapping between training

2.1 Naïve Bayes Classifeir:

In 1998 the Naïve Bayes classifier (figure 1) was proposed for spam recognition. Bayesian classifier is working on the dependent events and the probability of an event occurring in the future that can be detected from the previous occurring of the same event [2]. This technique can be used to classify spam e-mails; words probabilities play the main rule here. If some words occur often in spam but not in ham, then this incoming e-mail is probably spammed. Naïve bayes classifier technique has become a very popular method in mail filtering software. Bayesian filter should be trained to work effectively. Every word has certain probability of occurring in spam or ham email in its database. If the total of words probabilities exceeds a certain limit, the filter will mark the e-mail to either category. Naïve bayes classifier technique has become a very popular method in mail filtering software. Here, only two categories are necessary: spam or ham. Almost all the statistic-based spam filters use Bayesian probability calculation to combine individual token's statistics to an overall score [1], and make filtering decision based on the score.

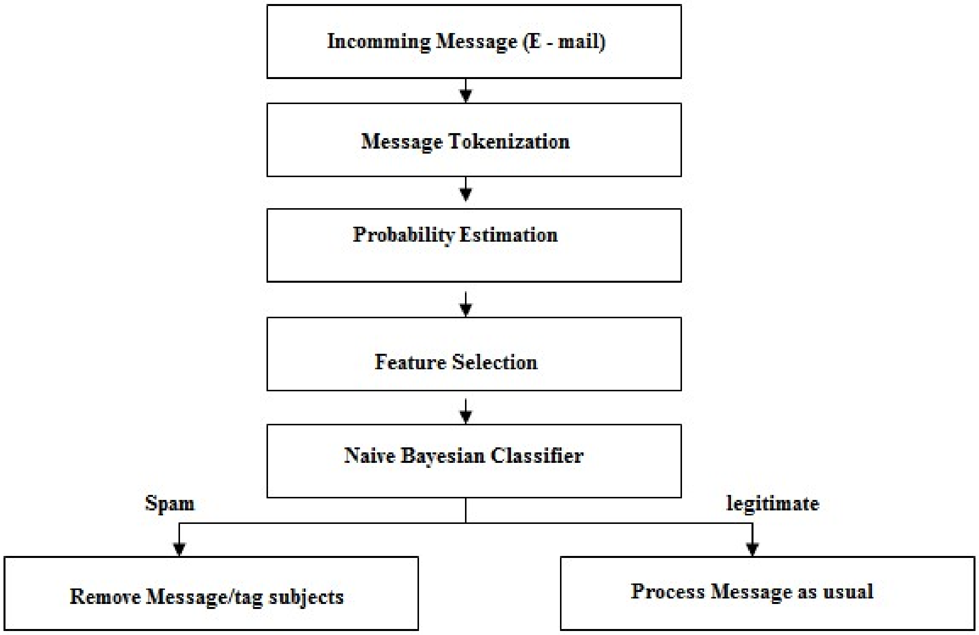

Figure 1: Naïve Bayes Classifier

The statistic we are mostly interested for a token T is its spamminess (spam rating) [4], calculated as follows:

$$S[T] = \frac{C_{Spam}(T)}{C_{Spam}(T) + C_{Ham}(T)}$$

Where CSpam(T) and CHam(T) are the number of spam or ham messages containing token T, respectively. To calculate the possibility for a message M with tokens {T1,......,TN}, one needs to combine the individual token's spamminess to evaluate the overall message spamminess.

2.2. Support Vector Machine

Electronic mail is a key revolution taking place over conventional communication systems due to its fast, convenient, easy, and economical, to use nature. A main bottleneck in electronic communications is the huge diffusion of unwanted, dangerous emails known as spam emails. A key concern is the developing of appropriate filters that can sufficiently capture those emails and get high performance rate. Machine learning (ML) researchers have developed numerous approaches in order to deal with this problem. Within the framework of machine learning, support vector machines (SVM) have prepared a large part to the development of spam email filtering. Based on Support Vector Machine, different scheme have been planned through text classification approaches (TC). A critical problem when using SVM is the selection of

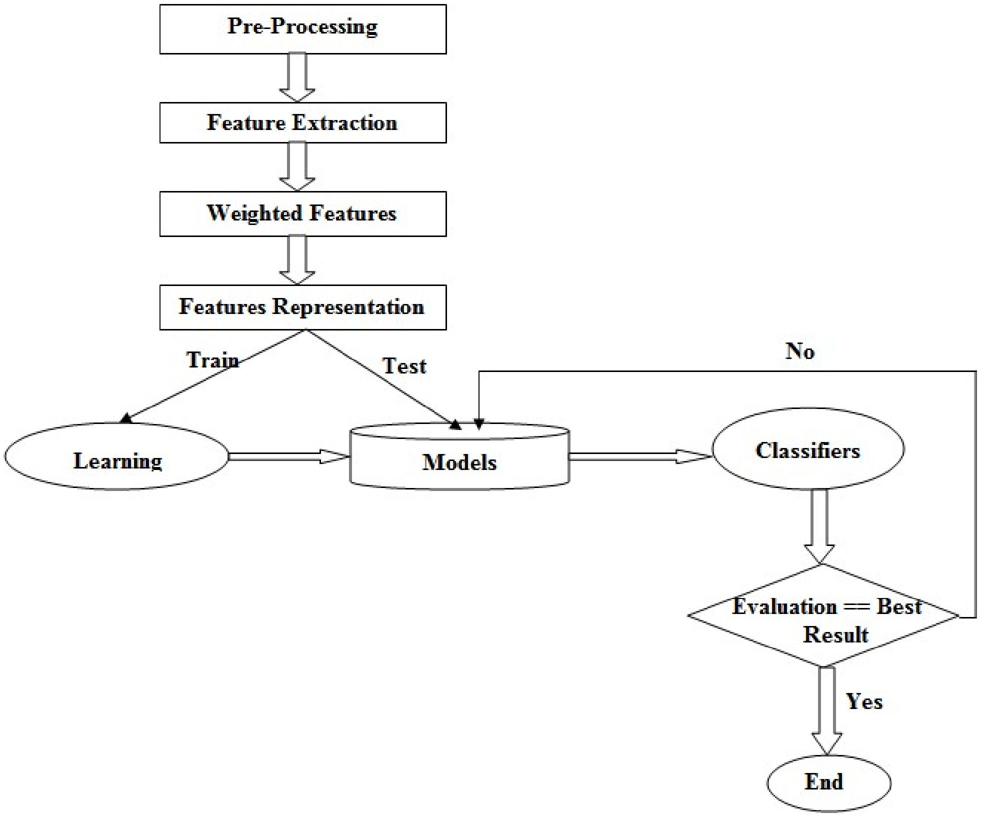

Figure 2: SVM Classifier

kernels as they openly affects the partition of emails in the quality space [5]. Here figure 2; explains the spam filtering using SVM.

3. Literature Survey

They presented hardware architecture of na¨ıve Bayes inference engine for spam control using two class e-mail classification. That can classify more 117 millions features per second given a stream of probabilities as inputs. This work can be extended to investigate proactive spam handling schemes on receiving e-mail servers and spam throttling on network gateways.[1]

The SVM for classification purpose, such system extract email sender behavior data based on global sending distribution, analyze them and assign a value of trust to each IP address sending email message, the Experimental results show that the SVM classifier is effective, accurate and much faster than the Random Forests (RF) Classifier.[3]

Email prioritization (PEP) method that specially focus on analysis of personal social networks to capture user groups and to obtain rich features that represent the social roles from the viewpoint of particular user, as well as they developed a supervised classification framework for modeling personal priorities over email messages, and for predicting importance levels for new messages.[6]

Immune-inspired model named innate and adaptive artificial immune system (IA-AIS) and applied to the problem of identification of unsolicited bulk e-mail messages (SPAM). It integrates entities analogous to macrophages, B and T lymphocytes, modeling both the innate and the adaptive immune systems. An implementation of the algorithm was capable of identifying more than 99% of legitimate or SPAM messages in particular parameter configurations. It was compared to an optimized version of the naive Bayes classifier, which have been attained extremely high correct classification rates. It has been concluded that IA-AIS has a greater ability to identify SPAM messages, although the identification of legitimate messages is not as high as that of the implemented naive Bayes classifier. [7]

Web spam with how to apply email spam detection techniques to identify spam web pages. Alike to the approach to identify spam in emails, web pages are scan for particular features that may categorize them as spam pages such as using keywords stuffing, unrelated popular words, etc.[8] paper represents one more instance of web or else link spam research paper. Blogs, public networks, news or else even e-commerce websites nowadays allow users to issue their comments or feedback. Spammers use such capability to post spam messages during those posts. Therefore spam detection techniques must be also used to permit automatic detection of such posts.

Spam-based categorization scheme of three category. In adding to classic spam and not spam category, a third uncertain category is provide to additional flexibility to the prediction algorithm. Undecided emails should be re-examined and collect more information to be capable then to critic whether they are spam or else not.[9] Here, try to sum up features that can recognize Botnets or spam proxy that are used to throw a huge number of spam emails. Authors look at network interrelated behaviors that can probably identify such spam proxy. [10][11] Evaluate apply uneven set on spam recognition with dissimilar rule execution scheme to get the best matching one. UCI Spam base is use in the investigational study (machine learning repository or repository). Unlike papers discussed the using of special algorithms and also apply the algorithms in special places between email senders along with receivers.

Email messages such as the relations among contacts and messages or else threads of messages. Threads of messages include numerous emails exchange between two or other persons throughout some email messages. Enron dataset is use in this learn similar to a lot of other related research papers in this area wherever it is considered as the major publically existing email messages dataset. For this particular paper, one more small email dataset (CMU) is use. [12] In the region of social networks along with email analysis through the goal of relevant analysis and categorization based on relations among people.

Real time email categorization and introduce GNUs mail open source used for email folder categorization. The application be developed to parse emails from dissimilar email clients along with perform several data mining analysis with WIKA data mining tool. In email database categorization is also base lying on the time of email messages [13]. The paper use Enron and SRI email datasets designed for the case lessons. Several new categorization method such as: MaxEnt were evaluate within the paper. The key decision to compose in every email categorization papers is what features to choose. Features can be associated to email designate, from or to addresses or else can be interrelated to the content; words, series of words, etc. Natural language processing tricks such as parsing as well as stemming are then concerned to parse email contents along with eliminate any words that may not be related for the classification procedure.

4. Proposed Work

In this work we have shown how classification algorithms work on two different sets. The first set which we had taken is ling spam corpus which is very big data set and it consists of various mails and these mails are classified into train emails and test emails are explain through in given figure 3.

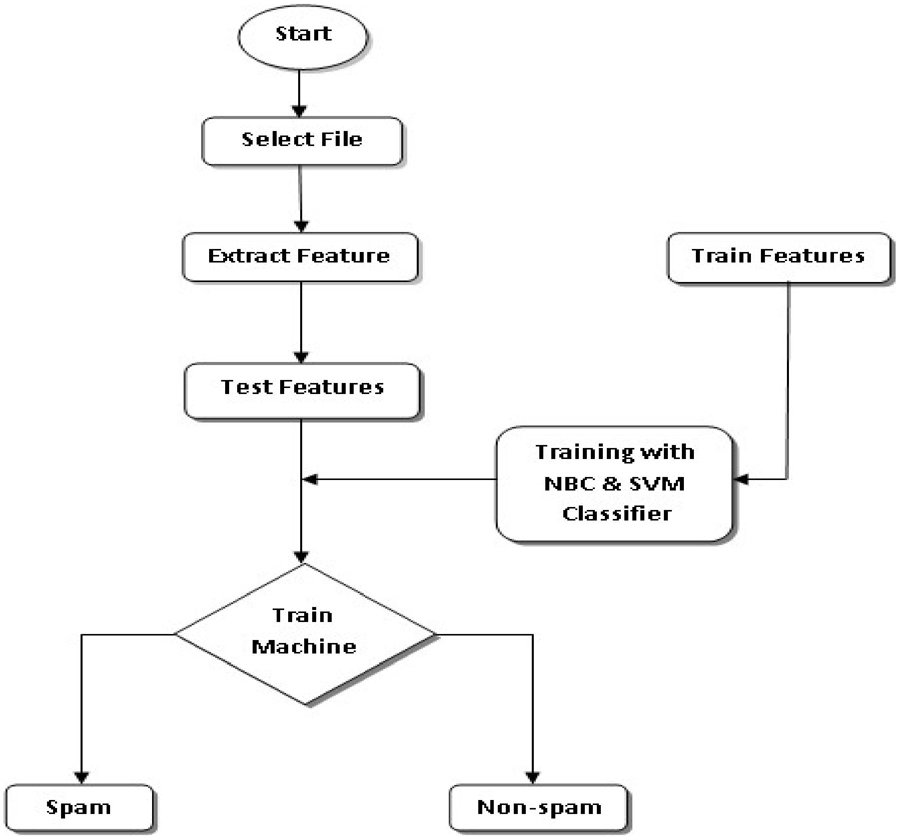

Figure: 3 Proposed method for classification via NBC & SVM

Here we have taken another data set which is anron, this is text file which mainly contains details about fruit with their colors. In this section we will first discuss how ling spam corpus works in steps after that we will move to next data set.

4.1 Classification on Ling Spam corpus and Anron data set

Here we will compare the classification algorithm on the basis of confusion matrix and accuracy. These classification algorithms have been applied on the dataset ling-spam which mainly consists of huge number of mails for training as well as for testing purpose. At the same time we in introduced one more approach that combine the classification algorithm whose accuracy may or may not be more than the previous one, it depends on the dataset and what type of value it contains. The steps involved during this process are as follows:

1. First step is to prepare the data
2. Dictionary will be created for each word
3. Feature extraction i.e. one of the most important process
4. Training the classifier

1. First step is to prepare the data

In this process we have split the downloaded data into training set and test set. Here we have taken ling corpus data set which mainly contains 702 training emails and 260 test mails means we have total of around 962 mails.

a) Removal of stop words – Stop words like “and”, “the”, “of”, etc are very common in all English sentences and are not very meaningful in deciding spam or legitimate status, so these words have been removed from the emails.

b) Lemmatization – It is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item. For example, “include”, “includes,” and “included” would all be represented as “include”. The context of the sentence is also preserved in lemmatization as opposed to stemming (another buzz word in text mining which does not consider meaning of the sentence).

2. Creating word dictionary

It can be seen that the first line of the mail is subject and the 3rd line contains the body of the email. We will only perform text analytics on the content to detect the spam mails. As a first step, we need to create a dictionary of words and their frequency. For this task, training set of 700 mails is utilized. This python function creates the dictionary for you.

Once the dictionary is created we can add just a few lines of code written below to the above function to remove non-words about which we talked in step 1. I have also removed absurd single characters in the dictionary which are irrelevant here.

3. Feature extraction process

Once the dictionary is ready, we can extract word count vector (our feature here) of 3000 dimensions for each email of training set. Each word count vector contains the frequency of 3000 words in the training file. Of course you might have guessed by now that most of them will be zero. Let us take an example. Suppose we have 500 words in our dictionary. Each word count vector contains the frequency of 500 dictionary words in the training file. Suppose text in training file was “Get the work done, work done” then it will be encoded as [0,0,0,0,0,…….0,0,2,0,0,0,……,0,0,1,0,0,…0,0,1,0,0,……2,0,0,0,0,0]. Here, all the word counts are placed at 296th, 359th, 415th, 495th index of 500 length word count vector and the rest are zero.

4. Training the classifiers

I have trained 5 models here namely Logistic Regression, Naive Bayes classifier, Random forest, Support Vector Machines (SVM) and ensemble method. Logistic regression is one of the most popular machine learning algorithms for binary classification. This is because it is a simple algorithm that performs very well on a wide range of problems. It takes real-valued inputs and makes a prediction as to the probability of the input belonging to the default class. Naive Bayes classifier is a conventional and very popular method for document classification problem. It is a supervised probabilistic classifier based on Bayes theorem assuming independence between every pair of features. We can force the decision trees to be different by limiting the features (rows) that the greedy algorithm can evaluate at each split point when creating the tree. This is called the Random Forest algorithm. SVMs are supervised binary classifiers which are very effective when you have higher number of features. The goal of SVM is to separate some subset of training data from rest called the support vectors (boundary of separating hyper-plane). The decision function of SVM model that predicts the class of the test data is based on support vectors and makes use of a kernel trick. The last one is ensemble method which means combining more than one method.

Once the classifiers are trained, we can check the performance of the models on test-set. We extract word count vector for each mail in test-set and predict its class (ham or spam) with the Logistic regression, NB classifier, Random forest, SVM model and ensemble method.

4.2. Algorithm and its flow chart

Here, we explain the algorithm and its flow chart (figure 4) for data set classification is as follow:

4.2.1. Training Set

The training set consists of a collection samples used as a reference for testing process. For example, in email classification the training sets are predefined ham and spam messages. These training sets undergo a preprocessing procedure before applying method. In document classification, Documents are represented as a function of the vocabulary terms.

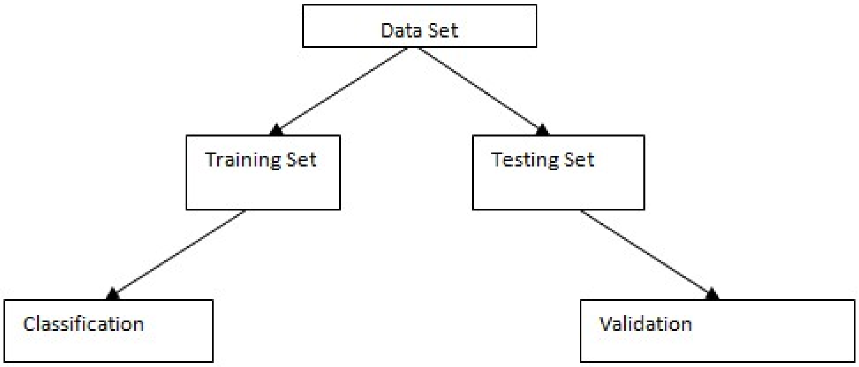

Figure: 4 Data set Classification

Accordingly, the model representation is a d x n matrix where d corresponds to the number of vocabulary words and n to the number of documents.

4.2.2. Testing Set

After Finding the PCA basis and projection matrices of each class, incoming messages are tested using Document Reconstruction. The objective of document reconstruction is to assign the new message to the correct.

4.2.3. Classification

Email filtering task depends on document classification approach. When classifying documents, choosing the best performing classifier is an elementary step. Thus extracting the best characterizing features, and correctly classifying incoming messages are key issues. The performance of the system is measured in terms of its accuracy.

4.2.4. Validation

Data set classified in set distinct set first one is training set and second one is testing set. After that this classification, training data set further to proceed for again classification and testing data set is also further to proceed validation to validate the data set.

4.3. Checking Performance

Test-set contains 130 spam emails and 130 non-spam emails. If you have come so far, you will find below results. I have shown the confusion matrix of the test-set for both the models. The diagonal element represents the correctly identified (a.k.a. true identification) mails where as non-diagonal elements represents wrong classification (false identification) of mails.

Algorithm:

1. Download the dataset from website or from inbuilt library

2. Preprocess the data set using preprocessing technique

3. After preprocessing apply the various machine learning classification algorithm like naïve bayes, logistic regression, random forest and SVM on the preprocess dataset.

4. Calculate the accuracy of the classification algorithms

5. Compare all the classification algorithms.

Algorithm

```
1. Initialize :
        train_dir = 'train-mails'
2. dictionary = make_Dictionary(train_dir)
3. Compute train_matrix:
         train_matrix = extract_features(train_dir)
4. Calling all the classifiers:
        model1 = LinearSVC()
        model2 = MultinomialNB()
        model4= LogisticRegression()
        model5= RandomForestClassifier()
5. Now test_dir = 'test-mails'
6. Computing the accuracy of all the classifiers
        'Accuracy Score: ' = metrics.accuracy_score (test_labels( Ham, Spam), result)
        Print ‘Accuracy Score
7. Define the function make_Dictionary
(a)     emails =  listdir(train_dir)
(b)     all words=[ ]
(c)     with open(mail) as m:
                for i, line in enumerate(m):
           if (i == 2):
              words = line.split()
              all_words += words
8. Define the function extract_features
(a)     files = listdir(mail_dir)
(b)     features_matrix = np.zeros((len(files),3000))
```

```
(c)     docID = 0;
(d)     for fil in files:
        with open(fil) as fi:
        for i,line in enumerate(fi):
        if i == 2:
        words = line.split()
```

4.4. Result Analysis and its parameters metrics used

Here, we have used classification algorithms available in different library. First we will compute the confusion matrix after that we will calculate the accuracy by using function metrics. accuracy_score; Firstly we will show the output of Ling Spam Dataset which is given below:

```
According to SVM
[[126   4]
 [  6 124]]
Accuracy Score: 96.1538461538%
According to Naive Bayes
[[129   1]
 [  9 121]]
Accuracy Score: 96.1538461538%
According to logistic regression
[[126   4]
 [  1 129]]
Accuracy Score: 98.0769230769%
According to Random forest
[[124   6]
 [  6 124]]
Accuracy Score: 95.3846153846%
Ensemble Method
[[129   1]
 [ 14 116]]
Accuracy Score: 94.2307692308%
```

Further proceed to show the output of anron Dataset which is given below:

According to logistic regression

Accuracy: 0.9760946395426645

Precision: 0.9761257884520781

Recall: 0.9760946395426645

According to Naive Bayes

Accuracy: 0.9801910766676561

Precision: 0.9805686707018927

Recall: 0.9801910766676561

According to SVM

Accuracy: 0.9743861140002076

Precision: 0.9743269007066634

Recall: 0.9743861140002076

According to Random Forest Classification

Accuracy: 0.9339224028465277

Precision: 0.9282226079845802

Recall: 0.9281132410379612

Method	**Base Methods**	**Proposed Methodology**	
		Data Set-1 (Ling-Spam Corpus)	**Data Set-2 (Enron-email)**
SVM	91 %	96.15 %	97.4%
Naive Bayes	92 %	96.15 %	98.0%
Logistic Regression	-	98.07 %	97.6%
Random Forest	-	95.0%	93.6%
PCA	94.5 %	-	-

Table 1: Comparisons of previous and present result on given data set

Comparative study of based methods to be used in previous paper and proposed methods in given table 1:

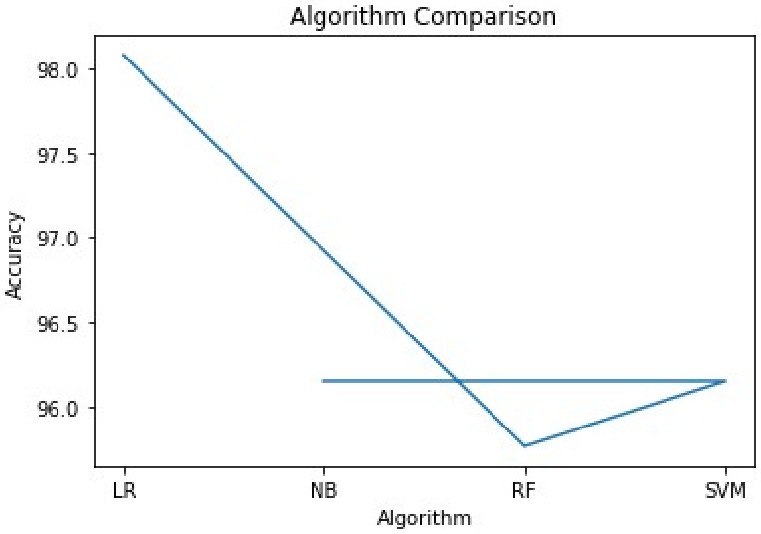

Figure: 5. A Comparative Study of different classifier on ling_spam_corpus Dataset

Correct Classification results for each classification algorithm on ling_spam_corpus Dataset and comparisons to each other with accuracy are shown in Figure 5.

5. Conclusions

In this paper we review some of the most popular machine learning methods and of their applicability to the problem of spam e-mail classification. Descriptions of the algorithms are presented, and the comparison of their performance on the Ling corpus Spam and Anron Dataset is presented, the experiment showing a very promising results specially in the algorithms that is not popular in the commercial e-mail filtering packages, spam recall percentage in the five methods has the accuracy values, while in term of accuracy we can find that the Naïve bayes and SVM methods and Logistic Regression methods has a very satisfying performance among the other methods, more research has to be done to escalate the performance of the Naïve bayes either by hybrid system or by resolve the feature dependence issue in the naïve bayes classifier, or hybrid the Immune by rough sets. Finally hybrid systems look to be the most efficient way to generate a successful anti spam filter nowadays.

Reference

[1] S. Whittaker, V. Bellotti and P. Moody, “Introduction to this special issue on revisiting and reinventing e-mail”, Human-Computer Interaction, 20(1), 1-9, 2005.

[2] G. Santhi, S. M. Wenisch and P. Sengutuvan, “A Content Based Classification of Spam Mails with

Fuzzy Word Ranking", IJCSI International Journal of Computer Science Issues, 10, 48-58, 2013.

[3] I. Koprinska, J. Poon, J. Clark and J. Chan, "Learning to classify e-mail", Information Sciences, 177(10), 2167-2187, 2007.

[4] M. Alazab and R. Broadhurst, "Spam and Criminal Activity", Australian Institute of Criminology, 2014.

[5] R. Amin, J. Ryan, and J. van Dorp, "Detecting Targeted Malicious Email Using Persistent Threat and Recipient Oriented Features", IEEE Secur, Priv. Mag, (99), 1-1, 2012.

[6] X. Carreras and L. Màrquez, "Boosting trees for anti-spam email filtering", Proceedings of 4th International Conference on Recent Advances in Natural Language Processing, 2001.

[7] K. N. Tran, M. Alazab and R. Broadhurst, "Towards a Feature Rich Model for Predicting Spam Emails Containing Malicious Attachments and URLs", 11th Australasian Data Mining Conference, Canberra, 2015.

[8] F. Temitayo, O. Stephen and A. Abimbola, "Hybrid GA-SVM for efficient feature selection in e-mail classification", Computer Engineering and Intelligent Systems, 3(3), 17-28, 2012.

[9] Ending Spam - Bayesian Content Filtering and the Art of Statistical Language Classification by Jonathan A. Zdziarski

[10] E-mail spam, http://en.wikipedia.org/wiki/E-mail_spam

[11] Spam (electronic), http://en.wikipedia.org/wiki/Spam_%28electronic%29

[12] A plan for spam by Paul Graham, http://www.paulgraham.com/spam.html

[13] Az embereket egyre kevésbé zavarja a spam,