Volume 4, Issue 6, June 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

E-Mail Spam Detection Using NLP

Er. Sugandha Sharma
Assistant Professor (CGC, Gharuan)
Computer Science & Engineering/PTU
India

Er. Seema Rani
Research Scholar (CGC, Gharuan)
Computer Science & Engineering/PTU
India

Abstract— ***Email is the "killer network application". Email is ubiquitous and pervasive. In a relatively short time frame, the Internet has become irrevocably and deeply entrenched in our modern society primarily due to the power of its communication substrate linking people and organizations around the globe. Much work on email technology has focused on making email easy to use, permitting a wide variety of information and information types to be conveniently, reliably, and efficiently sent throughout the Internet. However, the analysis of the vast storehouse of email content accumulated or produced by individual users has received relatively little attention other than for specific tasks such as Spam Detection. In this paper we represent new frame work for e-mail detection and comparison study of different machine learning technique. We have observed best result on SVM by this frame work.***

Keywords*— *Spam, email spam, Machine Learning, Machine Leaning Algorithms, Text Categorization, Vector Space Model

I. INTRODUCTION

Spam is flooding the Internet with many copies of the same message, in an attempt to force the message on people who would not otherwise choose to receive it. Most spam is commercial advertising, often for dubious products, get-rich-quick schemes, or quasi-legal services. Spam costs the sender very little to send -- most of the costs are paid for by the recipient or the carriers rather than by the sender.

Spam is an unwanted communication intended to be delivered to an indiscriminate target, directly or indirectly, notwithstanding measures to prevent its delivery. Spam filter is an automated technique to identify spam for the purpose of preventing its delivery. [1] The motivation behind spam is to have information delivered to the recipient that contains a *payload* such as advertising for a (likely worthless, illegal, or non-existent) product, bait for a fraud scheme, promotion of a cause, or computer malware designed to hijack the recipient's computer. Because it is so cheap to send information, only a very small fraction of targeted recipients — perhaps one in ten thousand or fewer — need to receive and respond to the payload for spam to be profitable to its sender. [2]

There are two main types of spam, and they have different effects on Internet users. Cancellable Usenet spam is a single message sent to 20 or more Usenet newsgroups. (Through long experience, Usenet users have found that any message posted to so many newsgroups is often not relevant to most or all of them.) Usenet spam is aimed at "lurkers", people who read newsgroups but rarely or never post and give their address away. Usenet spam robs users of the utility of the newsgroups by overwhelming them with a barrage of advertising or other irrelevant posts. Furthermore, Usenet spam subverts the ability of system administrators and owners to manage the topics they accept on their systems.

Email spam, also known as junk email or unsolicited bulk email (UBE), is a subset of electronic spam involving nearly identical messages sent to numerous recipients by email. Clicking on links in spam email may send users to phishing web sites or sites that are hosting malware. Spam email may also include malware as scripts or other executable file attachments. E-mail is a good, quick and a low cost communication approach. So spammers always opt to send spam through e-mail. Today every second user has an E-mail, and they are faced with spam problem consequently. E-mail Spam is non-requested information sent to the E-mail boxes. Email spam targets individual users with direct mail messages. Email spam lists are often created by scanning Usenet postings, stealing Internet mailing lists, or searching the Web for addresses. Email spams typically cost users money out-of-pocket to receive. Many people - anyone with measured phone service - read or receive their mail while the meter is running, so to speak. Spam costs them additional money. On top of that, it costs money for ISPs and online services to transmit spam, and these costs are transmitted directly to subscribers.

One particularly nasty variant of email spam is sending spam to mailing lists (public or private email discussion forums.) Because many mailing lists limit activity to their subscribers, spammers will use automated tools to subscribe to as many mailing lists as possible, so that they can grab the lists of addresses, or use the mailing list as a direct target for their attacks.

Spam may be a massive drawback for users and for ISPs. The causes are growth of value of electronic communications on the one hand and improvement of spam sending technology on the other hand. By spam reports of Symantec in 2013, the typical world spam rate for the year was 89.1%, with a rise of 1.4% compared with 2012. The proportion of spam sent from botnets was a lot of higher for 2013, accounting for roughly 88.2% of all spam. Despite several makes an

attempt to disrupt botnet activities throughout 2013, by the top of the year the overall variety of active bots came back to roughly an equivalent variety as at the top of 2012, with just about 5 million spam-sending botnets in use worldwide. [3] Spam messages cause lower productivity; occupy space in mail boxes; extend viruses, Trojans, and materials containing potentially harmful information for an explicit class of users, destroy stability of mail servers, and as a result users pay a plenty of time for sorting incoming mail and deleting undesirable correspondence. In step with a report from Ferris Analysis, the worldwide add of losses from spam created regarding 130 billion dollars, and within the USA, forty two billion in 2012. [4] Besides expenses for acquisition, installation, and repair of protective means, users are compelled to pay the extra expenses connected with an associated degree of the post traffic, failures of servers, and productivity loss. Thus we are able do such conclusion: spam is not solely an irritating factor, however a direct threat to the business. Considering the beautiful amount of spam messages coming to E-mail boxes, it is possible to assume that spammers don't operate alone; it's world, organized, making the virtual social networks. They attack mails of users, whole firms, and even states.

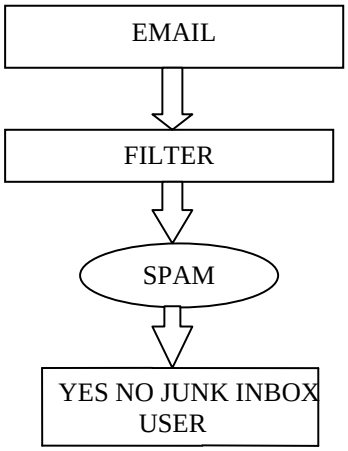

Fig No. 1 Block diagram of spam filter [28]

a. Machine learning:- Machine learning, is a branch of artificial intelligence, which is concerned with the construction and study of systems that can learn from data. For example, a machine learning system could be trained on email messages to learn to distinguish between spam and non-spam messages. After learning, it can then be used to classify new email messages into spam and non-spam folders. [5] The various machine learning algorithms used in our thesis work are:

- SGD Classifier
- Naive Bayes
- Perceptron
- SVM (Support Vector Machine)
- KNN (K Nearest Neighbours)

b. SGD Classifier (SGDC): Stochastic Gradient Descent (SGD):- is a simple yet very efficient approach to discriminative learning of linear classifiers under convex loss functions such as (linear) Support Vector Machines and Logistic Regression. Even though SGD has been around in the machine learning community for a long time, it has received a considerable amount of attention just recently in the context of large-scale learning. [29] SGD has been successfully applied to large-scale and sparse machine learning problems often encountered in text classification and natural language processing. [29]

c. Naïve Bayes (NB) Classifier:-Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem with the "naive" assumption of independence between every pair of features. [21] Given a class variable y and a dependent feature vector $x_1$through x_n, Bayes' theorem states the following relationship:

$$P(y|x_1,\dots.,x_n) = \frac{P(y)P(x_1,\dots x_n|y)}{P(x_1,\dots.,x_n)}$$

Using the naive independence assumption that
for all i, this relationship is simplified to

$$P(y|x_1,\dots,x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1,\dots.,x_n)}$$

d. Perceptron:-In machine learning, the perceptron is an algorithm for supervised classification of an input into one of several possible non-binary outputs. It is a type of linear classifier, i.e. a classification algorithm that makes its

predictions based on a linear predictor function combining a set of weights with the feature vector. The algorithm allows for online learning, in that it processes elements in the training set one at a time.

e. KNN K Means:- The aim of K Means is to partition the objects in such a way that the intra cluster similarity is high but inter cluster similarity is comparatively low. A set of n objects are classified into k clusters by accepting the input parameter k. All the data must be available in advance for the classification. [25]
KNN: Instead of assigning to a test pattern the class label of its closest neighbor, the K Nearest Neighbor classifier finds k nearest neighbors on the basis of Euclidean distance. [26]

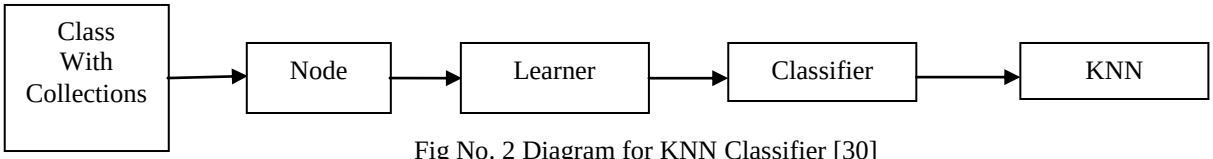

Fig No. 2 Diagram for KNN Classifier [30]

Square root of $((x_2-x_1)^2-(y_2-y_1)^2)$

f. SVM:- In SVM is a new method for the classification of both linear and non-linear data. SVM are supervised learning models and it associated with learning algorithms that analyses data and recognizes patterns. [24] The basic SVM takes a set of input data, for each given input, which has two possible class forms the output making it a non-probabilistic binary linear classifier

II. TEXT CATEGORIZATION

The goal of text categorization is the classification of documents into a fixed number of predefined categories. Each document can be in multiple, exactly one, or no category at all. Using machine learning, the objective is to learn classifiers from examples which perform the category assignments automatically. This is a supervised learning problem. Since categories may overlap, each category is treated as a separate binary classification problem.

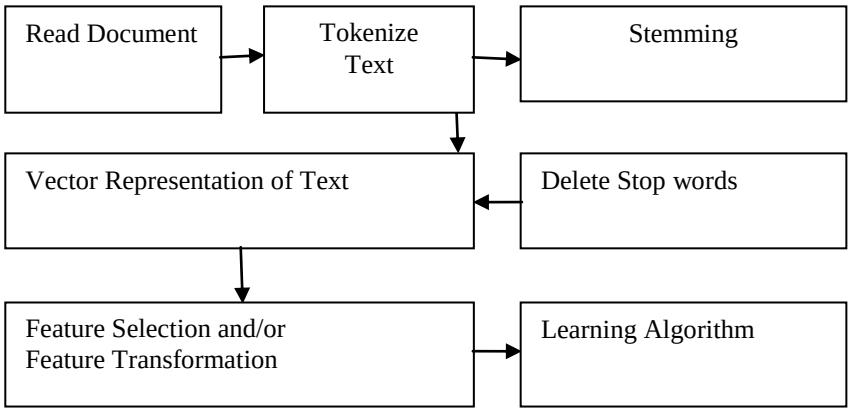

Fig No. 3 Text Classification Process [20]

The first step in text categorization is to transform documents, which typically are strings of characters, into a representation suitable for the learning algorithm and the classification task. Information Retrieval (IR) research suggests that word stems work well as representation units and that their ordering in a document is of minor importance for many tasks. This leads to an attribute-value representation of text. Each distinct word w_i corresponds to a feature with the number of times word wi occurs in the document as its value. To avoid unnecessarily large feature vectors, words are considered as features only if they occur in the training data at least 3 times and if they are not \stop-words" (like \and", \or", etc.). This representation scheme leads to very high-dimensional feature spaces containing 10000 dimensions and more. Many have noted the need for feature selection to make the use of conventional learning methods possible, to improve generalization accuracy, and to avoid \over fitting". Vector representation is used to represent the text in a document.

III. VECTOR SPACE MODEL

The tf–idf weight (term frequency–inverse document frequency) is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document in a collection or corpus. The importance increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. Variations of the tf–idf weighting scheme are often used by search engines to score and rank a document's relevance given a user query.
The *term frequency* in the given document is simply the number of times a given term appears in that document. This count is usually normalized to prevent a bias towards longer documents (which may have a higher term frequency regardless of the actual importance of that term in the document) to give a measure of the importance of the term *t i* within the particular document.

$$tf = \frac{n_i}{\sum_k n_k}$$

With n_i being the number of occurrences of the considered term, and the denominator is the number of occurrences of all terms.

The *inverse document frequency* is a measure of the general importance of the term (it is the logarithm of the number of all documents divided by the number of documents containing the term).

$$idf = log \frac{|D|}{|(d_i \supset t_i)|}$$

With

- |D| : total number of documents in the corpus
- $|(d_i \supset t_i)|$: Number of documents where the term *t i* appears.

Then

$$tfidf = tf.idf$$

A high weight in tf–idf is reached by a high term frequency (in the given document) and a low document frequency of the term in the whole collection of documents; the weight hence tends to filter out common terms.

For each document d_i and keyword k_i "tf-idf" is defined by the weight "w"

.

IV. RELATED WORK

Wan et.al in his paper proposed a spam detection method that uses Sobel operators for edge detection and a multiple filter using Sobel operators and OCR. As spam filters easily catches text, so spam senders uses images to send spam instead of text. Traditional image spam filters have weaknesses in scanning documents and photographs. However, to transmit information, text is always used in image spam. Therefore, in this study, author classifies mail images by the configuration of letters and images. [12]

Borg et.al in his paper presents a method that can use several social networks for detecting spam and a set of metrics for representing OSN data. The paper investigates the impact of using social network data extracted from an E-mail corpus to improve spam detection. The social data model is compared to traditional spam data models by generating and evaluating classifiers from both model types. The results in this paper show that accurate spam detectors can be generated from the low-dimensional social data model alone, however, spam detectors generated from combinations of the traditional and social models were more accurate than the detectors generated from either model in isolation. Online Social Networks (OSNs) contain more and more social information, contributed by users. OSN information may be used to improve spam detection. [6]

McCord et.al in his paper discuss some user-based and content-based features that are different between spammers and legitimate users. These features are then used to facilitate spam detection. Using the API methods provided by Twitter, they crawled active Twitter users, their followers/following information and their most recent 100 tweets. Then, detection scheme is evaluated based on the suggested user and content-based features. [7]

Zhang et al., 2008 describes a genetic programming approach to feature extraction for a cost-sensitive classification task of spam. The fitness used comprised three objectives: an approximation to the Bayes error, misclassification cost and number of tree nodes used to encode a particular solution. The solution proposed in (Zhang et al., 2008) is the most analogous to the one presented in this paper, since an EA is used for the feature selection. [8]

Dudley et al., 2008 proposed an EA to analyse different configurations for Spam Assassin, a widely-used open source spam filter. Their approach consisted in using an EA to achieve an optimal setup, at a personalized level, for the set of weights that is used to infer if a given message is spam. In this case, the EA minimized the number of false positives and false negatives. [9]

Lee et.al in his paper, for spam detection, planned parameter optimization and has choice to cut back process overheads with guaranteeing high detection rates. In previous papers, either parameter optimization or feature selection are used, however not each. Parameter optimization could be a method that regulates parameters of spam detection models to work out optimum parameters of the detection model. Feature selection could be a method that chooses solely necessary options or feature commenced of all the options. Feature selection allows eliminating orthogonal options to avoid process overheads. [11]

Araujo et.al in his paper, present an efficient spam detection system based on a classifier that combines new link-based features with language-model (LM)-based ones. These features are not only related to quantitative data extracted from the Web pages, but also to qualitative properties, mainly of the page links. He considers, for instance, the ability of a search engine to find, using information provided by the page for a given link, the page that the link actually points at. This can be regarded as indicative of the link reliability. He also checks the coherence between a page and another one pointed at by any of its links. Two pages linked by a hyperlink should be semantically related, by at least a weak contextual relation. Thus, he applies an LM approach to different sources of information from a Web page that belongs to the context of a link, in order to provide high-quality indicators of Web spam. They have specifically applied the Kullback–Leibler divergence on different combinations of these sources of information in order to characterize the relationship between two linked pages. The result is a system that significantly improves the detection of Web spam using fewer features, on two large and public datasets such as WEBSPAM-UK2006 and WEBSPAM-UK2007. [10]

V. PROBLEM DEFINITION & OBJECTIVES

Spam messages cause lower productivity; occupy space in mail boxes; extend viruses, Trojans, and materials containing potentially harmful information for a certain category of users, destroy stability of mail servers, and as a result users spend a lot of time for sorting incoming mail and deleting undesirable correspondence.

According to a report from Ferris Research, the global sum of losses from spam made about 130 billion dollars, and in the USA, 42 billion in 2012. Besides expenses for acquisition, installation, and service of protective means, users are compelled to defray the additional expenses connected with an overload of the post traffic, failures of servers, and productivity loss.

So the conclusion is that: spam is not only an irritating factor, but also a direct threat to the business. Considering the stunning quantity of spam messages coming to E-mail boxes, it is possible to assume that spammers do not operate alone; it is global, organized, creating the virtual social networks. They attack mails of users, whole corporations, and even states. Several methods for spam detection have been used till now. In our thesis work, we represent new frame work for e-mail detection and comparison study of different machine learning technique

.

Objectives:

- To classify messages as spam or non-spam.
- Spam text is highly uncertain, so to make a model in which, to optimize the uncertainty of spam by using NLP approaches.
- Compare this model with existing model.
- Validate our approach.

VI. PROPOSED METHODOLOGY

Step 1: Tokenization & Stemming: -

Tokenization is the process of breaking up the given text into units called tokens. The tokens may be words or number or punctuation mark. Tokenization does this task by locating word boundaries. Ending point of a word and beginning of the next word is called word boundaries. Tokenization is also known as word segmentation.

Stemming usually refers to a crude heuristic process that chops off the ends of words in the hope of achieving this goal correctly most of the time, and often includes the removal of derivational affixes.

Step 2: Vector Model of Text:-

Vector space model or term vector model is an algebraic model for representing text documents (and any objects, in general) as vectors of identifiers, such as, for example, index terms. It is used in information filtering, information retrieval, indexing and relevancy rankings. [20]

Step 3: Feature Selection:-

Feature selection is the process of selecting a subset of the terms occurring in the training set and using only this subset as features in text classification. Feature selection serves two main purposes. First, it makes training and applying a classifier more efficient by decreasing the size of the effective vocabulary. This is of particular importance for classifiers that, unlike NB, are expensive to train. Second, feature selection often increases classification accuracy by eliminating noise features. A *noise feature* is one that, when added to the document representation, increases the classification error on new data.

Step 4: Probabilistic model by learning:-

Statistical analysis tool that estimates, on the basis of past (historical) data, the probability of an event occurring again.

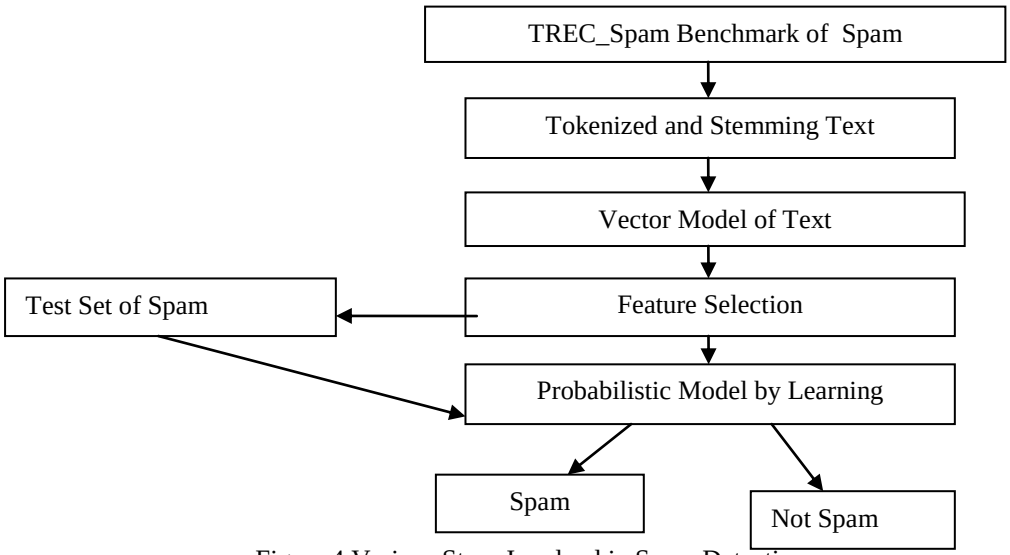

Fig no 4.Various Steps Involved in Spam Detection

TREC: The Text REtrieval Conference (TREC), co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense, was started in 1992 as part of the TIPSTER Text program. Its purpose was to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies.

Step5: Performance measures:

i. **Precision:** In the field of information retrieval, **precision** is the fraction of retrieved documents that are relevant to the find:

$$\text{precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|}$$

Precision takes all retrieved documents into account, but it can also be evaluated at a given cut-off rank, considering only the topmost results returned by the system. This measure is called **precision at n** or **P@n**.

ii. **Recall:** Recall in information retrieval is the fraction of the documents that are relevant to the query that are successfully retrieved.

$$\text{recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents}\}|}$$

For example for text search on a set of documents recall is the number of correct results divided by the number of results that should have been returned.

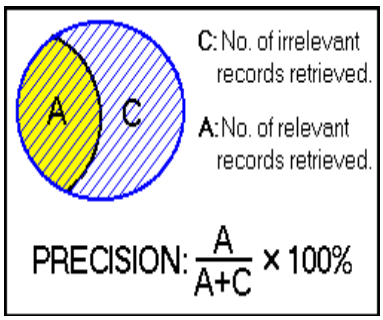

Fig no 5. Precision

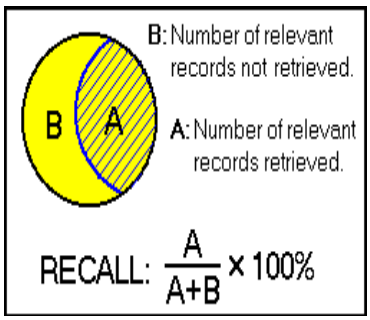

Fig no 6. Recall

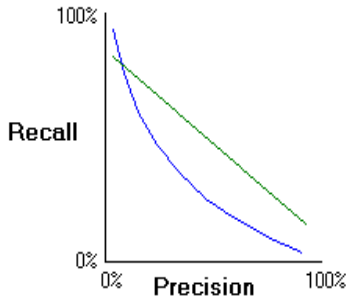

Fig no 7. Graph between Precision &Recall

iii. **F-measure:** In statistical analysis of binary classification, the **F_1 score** (also **F-score** or **F-measure**) is a measure of a test's accuracy. It considers both the precision *p* and the recall *r* of the test to compute the score: *p* is the number of correct results divided by the number of all returned results and *r* is the number of correct results divided by the number of results that should have been returned. The F_1 score can be interpreted as a weighted average of the precision and recall, where an F_1 score reaches its best value at 1 and worst score at 0.
The traditional F-measure or balanced F-score (**F_1 score**) is the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

iv. **Accuracy:** In the fields of science, engineering, industry, and statistics, the accuracy of a measurement system is the degree of closeness of measurements of a quantity to that quantity's actual (true) value.

v. **Error rate:** Error rate of a classifier was defined as the percentage of the dataset incorrectly classified by the method. It is the probability of misclassification of a classifier
Error rate= no of incorrectly classified samples/Total no. of samples in the class

VII. RESULTS

In our thesis work 6 different algorithms are introduced in class to implement spam Detection in Python, and their performance is compared. The algorithms we used were:

- SGD Classifier
- Naive Bayes
- Perceptron
- SVM (Support Vector Machine)

- KNN (K Nearest Neighbours)
  We examine the performance of each algorithm in 2 aspects:
- Error rate
- False positive ratio

False positive ratio is interesting because detection out a ham (non-spam) message is a bad thing, worse than letting a spam message get through creation of training set and test set We wrote a Python script to process these messages and create a feature vector out of each message. In the sequel it is described how the feature vectors look like. The script divides the feature vectors into training set and test set, while preserving the ham-spam ratio in each set. Actually, the script randomly creates 90 different pairs of training set and test set as follows:

- We used 9 different "training fractions", i.e. the percentage of training set size out of the entire dataset.
- The training fractions we used were: 0.1, 0.2, ..., 0.9.
- For each training fraction, we randomly created 10 different pairs of training set and test set, so we can examine the performance as an average of 10 runs.

In the following tables the various parameters for various machine learning algorithms has been shown. Here all results are with training fraction of 0.5 except for SVM for which the training fraction is 0.4 (which was the maximum we could test) .We see that the best two algorithms are Naïve Bayes and SVM. The values of F-measure, Accuracy, Precision, Recall, Error rate are given in tables for different machine learning algorithms.

TABLE I. ANALYSIS OF RESULTS AFTER USING NEURAL NETWORK PERCEPTRON

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Error rate
Perceptron	0.04	0.56	83	81.5	88	96	18.5

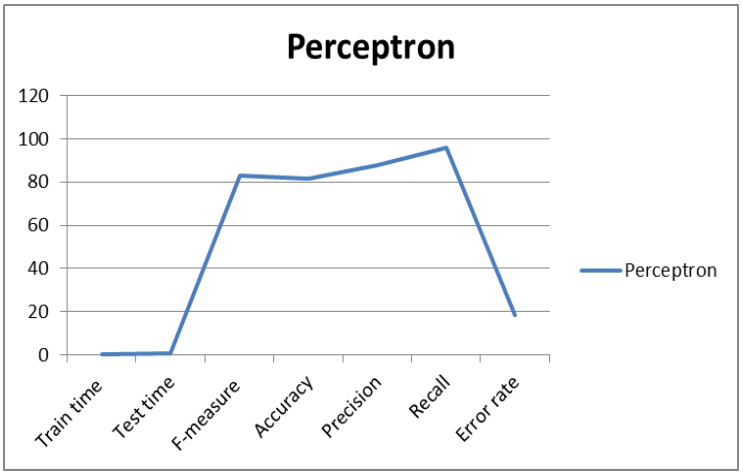

Fig no. 8 A graph showing the analysis of results after using neural network perceptron

TABLE 2 ANALYSIS OF RESULTS AFTER USING K-MEAN

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Error rate
K-mean	0.02	0.571	85	91.2	89	94	8.8

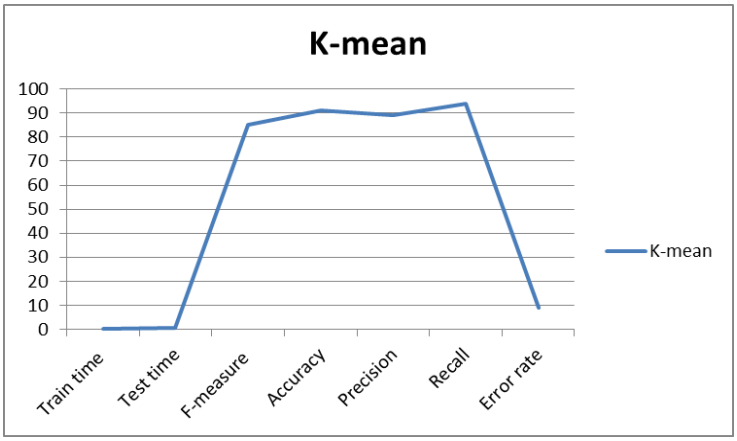

Fig no. 9 A graph showing the analysis of results after using K-mean

TABLE 3: ANALYSIS OF RESULTS AFTER USING NAIVE BAYES

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Dimensionality	Density	Error rate
Naïve Bayes	0.018	0.005	89	92.4	92	89	33810	1	7.6

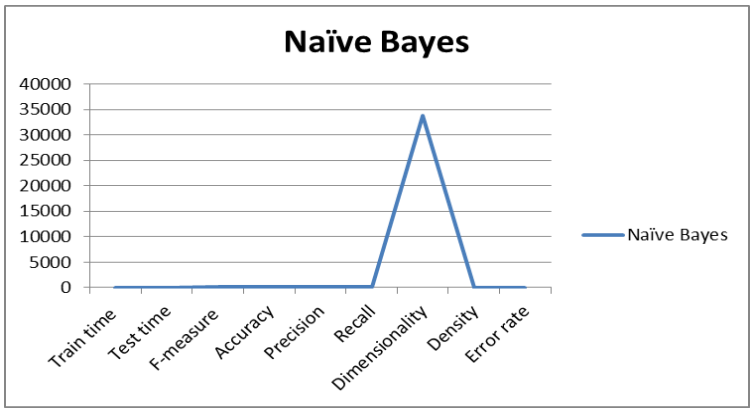

Fig no. 10 A graph showing analysis of results after using Naïve Bayes Classifier

TABLE 4: ANALYSIS OF RESULTS AFTER USING SGD CLASSIFIER

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Dimensionality	Density	Error rate
SGD Classifier	0.371	0.004	90	93.1	97	88	33810	0.666	6.9

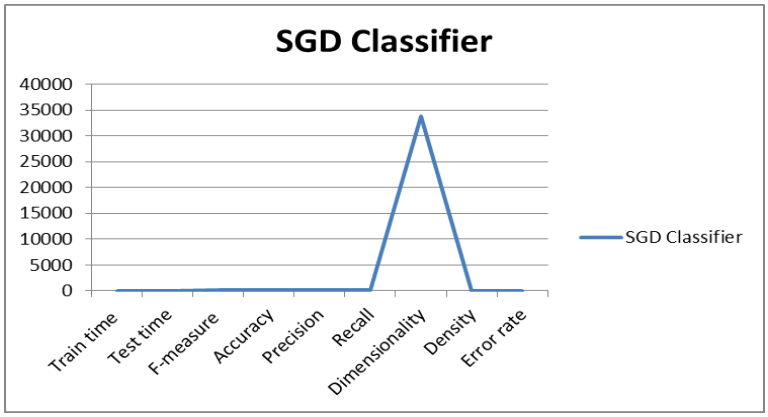

Fig no. 11 A graph showing analysis of results after using SGD Classifier

TABLE 5: ANALYSIS OF RESULTS AFTER USING SVM

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Dimensionality	Density	Error rate
SVM	0.56	0.04	92	95	96	95	534	0.89	5

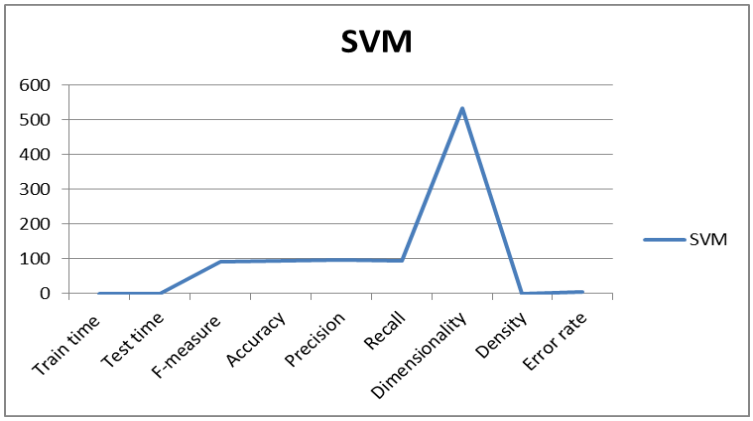

Fig No. 12 A graph showing analysis of results after using SVM

TABLE 6: ANALYSIS OF RESULTS AFTER USING LINEAR SVC

Classifier	Train time	Test time	F-measure	Accuracy	Precision	Recall	Dimensionality	Density	Error rate
Linear SVC	0.59	0.879	87	91.9	93	90	561	0.99	7.9

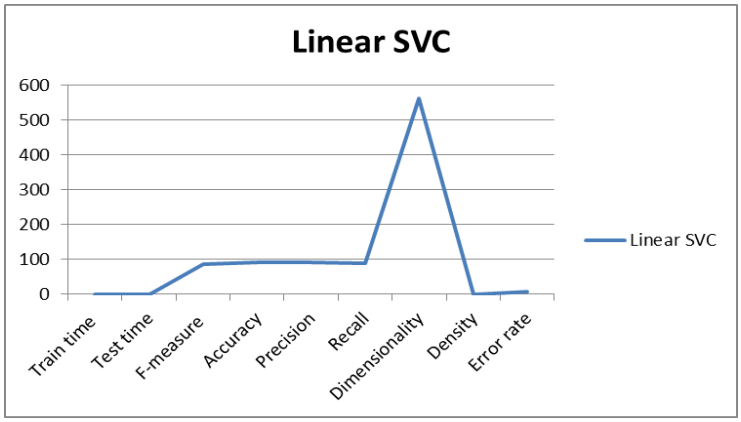

Fig no. 13 A graph showing analysis of results after using Naïve Bayes Classifier

TABLE 7 ANALYSIS OF RESULTS OF ALL CLASSIFIER

Classifier	**Perceptron**	**K-mean**	**NB**	**SGDC**	**LSVC**	**SVM**
Train time	0.04	0.02	0.018	0.371	0.59	0.56
Test time	0.56	0.571	0.005	0.004	0.879	0.04
F-measure	83	85	89	90	87	92
Accuracy	81.5	91.2	92.4	93.1	91.9	95
Precision	88	89	92	97	93	96
Recall	96	94	89	88	90	95
Dimensionality	NA	NA	33810	33810	561	534
Density	NA	NA	1	0.666	0.99	0.89
Error rate	18.5	8.8	7.6	6.9	7.9	5

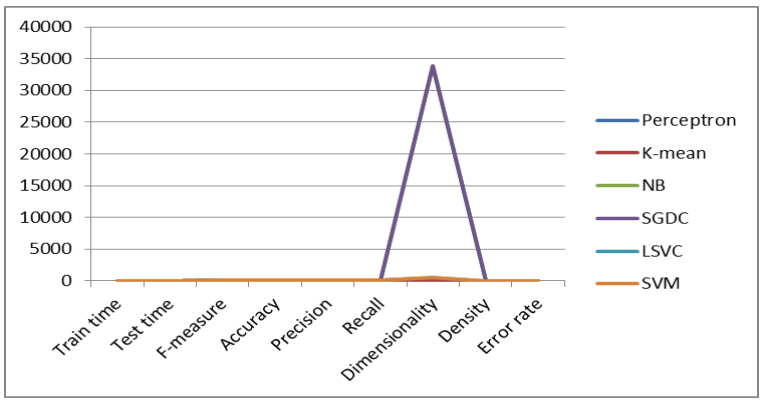

Fig No. 14 A graph showing analysis of results of all the classifier methods

VIII. CONCLUSIONS

Above tables show the best results of each algorithm using its optimal settings. Here all results are with training fraction of 0.5 except for SVM for which the training fraction is 0.4 (which was the maximum we could test). We see that the best two algorithms are Naïve Bayes and SVM. Considering the importance of low false positive ratio in spam filtering problem, it is understandable why Naïve Bayes is most widely used in real life. Adding the facts that Naïve Bayes is much easier to implement and has much lower running-time, it becomes clear why Naïve Bayes makes a natural choice.

ACKNOWLEDGMENT

This research paper is made possible through the help and support from my thesis guide.

First and foremost, I would like to thank Er. Sugandha Sharma for her most support and encouragement, she kindly read my paper and offered invaluable detailed advices on grammar, organization, and the theme of the paper.

Finally, I sincerely thank to my college HOD, teachers and friends. The product of this research paper would not be possible without all of them.

REFERENCES

[1] Gordon V. Cormack, David R. Cheriton, "Email Spam Filtering: A Systematic Review", Foundations and Trends ®in Information Retrieval Vol. 1, No. 4 (2006) 335–455©2008.

[2] M. Mangalindan, "For bulk E-mailer, pestering millions offers path to profit," Wall Street Journal, November 13, 2002.

[3] Fabrizio Sebastiani. "Machine learning in auto-mated text categorization- ACM Computing Surveys", 34(1):1-47, 2002.

[4] Qian Xu, Evan Wei Xiang and Qiang Yang, "SMS Spam Detection Using Non-Content Features" publication in IEEE Intelligent Systems, Nov.-Dec. 2012 (vol. 27 no. 6)pp. 44-51.

[5] http://en.wikipedia.org/wiki/Machine_learning

[6] Anton Borg, Niklas Lavesson, "E-mail Classification using Social Network Information" Seventh International Conference on Availability, Reliability and Security,IEEE,2012

[7] M. McCord and M. Chuah, "Spam Detection on Twitter Using Traditional Classifiers" pp. 175–186, 2011.© Springer-Verlag Berlin Heidelberg 2011

[8] Zhang, Y., Li, H., Niranjan, M., and Rockett, P. (2008). Applying cost-sensitive multi objective genetic programming to feature extraction for spam e-mail filtering. In Proc. of the 11th European conference on Genetic programming, pages 325–336. Springer-Verlag.

[9] Dudley, J., Barone, L., and While, L. (2008). Multi objective spam filtering using an evolutionary algorithm, pages 123–130. IEEE.

[10] Lourdes Araujo and Juan Martinez-Romo, "Web Spam Detection: New Classification Features Based on Qualified Link Analysis and Language Models" IEEE Transactions On Information Forensics And Security, Vol. 5, No. 3, September 2010

[11] Sang Min Lee, Dong Seong Kim, Ji Ho Kim, Jong Sou Park, "Spam Detection Using Feature Selection and Parameters Optimization", pp. 883-888, 2010,IEEE.

[12] Peng Wan, Uehara, "Multiple Filters of Spam Using Sobel Operators and OCR" IEEE, pp.164-169,July,2012

[13] K. S. Xu, M. Kliger, Y. Chen, P. J.Woolf, and A. O. Hero, "Revealing social networks of spammers through spectral clustering", in Proceedings of the IEEE International Conference on Communications, (ICC '09), Dresden, Germany, April 2013.

[14] Mohammad Razmara, Babak Asadi, Masoud Narouei, Mansour Ahmadi, "A Novel Approach toward Spam Detection Based on Iterative Patterns", 2012, IEEE.

[15] Ram B. Basnet, Andrew H. Sung, "Classifying Phishing Email Using Confidence-Weighted Linear Classifiers", pp. 108-112, 2010 IEEE.

[16] Juan Martinez-Romo, Lourdes Araujo, "Detecting malicious tweets in trending topics using a statistical analysis of language", Expert Systems with Applications 40 (2013) 2992–3000, Elsevier.

[17] Joshua Goodman, Gordon V.Cormack, and David Heckerman, "Spam and the Ongoing Battle for the Inbox", Communications of the ACM, February 2007/Vol.50,No.2.

[18] Sarwat Nizamani, Nasrullah Memon, Uffe Kock Wiil, Panagiotis Karampelas, "Modelling Suspicious Email Detection using Enhanced Feature Selection", April 2012.

[19] Qian Xu, Evan Wei Xiang and Qiang Yang, "SMS Spam Detection Using Non-Content Features" publication in IEEE Intelligent Systems, Nov.-Dec. 2012 (vol. 27 no. 6)pp. 44-51.

[20] M. IKONOMAKIS, S. KOTSIANTIS, V. TAMPAKAS, "Text Classification Using Machine Learning Techniques" WSEAS TRANSACTIONS on COMPUTERS, Issue 8, Volume 4, August 2005, pp. 966-974

[21] http://scikit-learn.org/stable/modules/naive_bayes.html

[22] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. Information Processing and Management, 24(5):513{523, 1988.

[23] Y. Yang and J. Pedersen. "A comparative study on feature selection in text categorization." In International Conference on Machine Learning (ICML), 1997

[24] R Kishore Kumar, G Poonkuzhali, and P Sudhakar. Comparative study on email spam classifier using data mining techniques. In Proceedings of the International MultiConference of Engineers and Computer Scientists, volume 1, 2012.

[25] Sudipto Guha, Adam Meyerson, Nina Mishra, Rajeev Motwani and Liadan O'Callaghan, "Clustering Data Streams", IEEE Transactions on Knowledge & Data Engg., 2003

[26] Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques ", Second Edition.

[27] http://www.iicm.tugraz.at/cguetl/courses/isr/opt/classification/Vector_space_Model.html

[28] Sahil Puri, Dishant Gosain, Mehak Ahuja, Ishita Kathuria, Nishtha Jatana, "Comparison And Analysis Of Spam Detection Algorithms", IJAIEM, Volume 2, Issue 4, April 2013

[29] http://scikit-learn.org/stable/modules/sgd.html

[30] http://www.pymvpa.org/generated/mvpa2.clfs.knn.kNN.html

[31] http://en.wikipedia.org/wiki/Precision_and_recall

