Various Classifiers with Optimal Feature Selection for Email Spam Filtering

[1]Vrinda Sharma, [2]Monika Poriye, [3]Vinod Kumar
[1]M.Tech Student, [2,,3]Assistant Professor:
vrindasharma206@gmail.com, monikaporiye@gmail.com, dcsavinod@gmail.com

Abstract: Electronic mails (Emails) are the most commonly used utility of internet for communication. But now-a-days in the era of digital world, it contains some malicious code to steal some confidential information of the user or to infect the user's system through malicious mails. Spam mails are unwanted, unsolicited or undesired electronic mails which are communicating through social networks. They reduces productivity, reliability, spoils storage space and spread viruses, worms, malware or ransomware. So, there is a need of spam detection to reduce their consequences. In this paper, term finder method ($\boldsymbol{tf - idf}$) threshold and optimal Feature selection method through information gain is proposed. These two methods are applied on four different classifiers i.e. Support Vector Machine, Naïve Bayes, K-Nearest Neighbors and Random Forest and tested one by one on different well known datasets. The performance is evaluated in terms of accuracy, time and F-Measure.

Keywords: Information Gain; Optimal Feature Selection; Random Forest; Support Vector Machine (SVM); Spam Classification.

I. INTRODUCTION

Email is the instant messaging method of exchanging information among different users. Email is very popular tool for personal or commercial use. But junk mails (i.e. spam mail) are becoming nuisance in the communication because many copies of same message is flooded over the network which may choke the network as bandwidth provided to a network is very limited. Spam mails may be sent to the same user or to the multiple user at the same time. Emails are the primary vector of delivering the malicious code i.e. virus, worms or malware etc.[1]

Sending Unsolicited Bulk Emails (UBE) or Unsolicited Commercial Emails (UCE) are banned by all Internet Service Providers (ISPs) worldwide. Thus spammers send these messages through Zombie Networks i.e. virus infected systems so that they couldn't be responsible for such illegal activities. The SpamHaus's anti-spam block list is used by more than one billion internet users to avoid spam messages. [2]According to the Cyberoam Report 2014, 54 billion spam messages are sent every day. [3]

There are many techniques present which is used to filter out spam or ham like blacklisting, whitelisting, signature-based techniques, mail header checking, content based analysis, machine learning techniques (i.e. classification or clustering) and many more.

The primary goal of Machine Learning implementation is to develop a general purpose algorithm that solves a practical and focused problems. In Machine Learning, many Classification, Clustering, Regression and Dimensionally Reduction Techniques are defined. In this study, four different classification techniques are applied with some work directed to improve the performance of the classifiers. Here, term finder technique threshold and optimal feature selection method are proposed. Those methods are applied on multiple dataset and their performance is measured on the basis of accuracy and time. The main concern of this research is to get more accuracy with less execution time.

The rest paper is segmented into following sections: Section II: summary of the related work belongs to this research. Section III: description of the proposed work and machine learning classifiers. Section IV: description of the datasets used. Section V: Results and Analysis Section VI: Conclusion SectionVII: Future Scope and Section VIII: References.

I. Related Work

Many supervised or unsupervised machine learning techniques exists for spam filtering and some work is done to provide effective results in terms of accuracy, F-measure or time. Some of the supervised machine learning techniques related to this research work and their results are summarized below:

Spam Detection Filter using KNN Algorithm and Resampling, 2010.Feature extraction is performed using top n words approach.Resulting output is resampled and again given as an input to the classifier. Further final output is measured on the basis of F-measure and true positive rate. The performance is compared on two parameters i.e. before resampling and after resampling. K-Nearest Neighbors classifier is used to filter out the spam mails.[4]

Content Based Spam Detection in Email using Bayesian Classifier, 2015. Here, major concern is on preprocessing i.e. html tag removal, stop-word removal, tokenization, word frequency and then Bayesian Classifier is used for spam filtering. This classifier shows the accuracy of more than 96.46%.[5]

A Combining Classifiers Approach for detecting Email Spam, 2016. This study focuses on “Boosted Bayesian”, “Boosted Naïve Bayes and Support Vector Machine”. Finally, combined classifiers gives the accuracy of 98.8%.[6]

A Survey and Evaluation of Supervised Learning Techniques for Spam E-Mail Filtering, 2015. Clustering (J48, ID3 etc.) and Classification techniques (Naïve Bayes etc.) for spam filtering are discussed here. Finally, A comparative study of each technique is given in terms of accuracy and time graphs.[7]

II. EXPERIMENTAL DESIGN

Framework of the Proposed Model:

In the proposed model, a no. of steps performed to classify the given input dataset into two categories i.e. spam or ham. Generally, an email contains two parts i.e. header (information of sender, receiver, subject and so on) and body (main text data of an email). This model focuses on body part only.

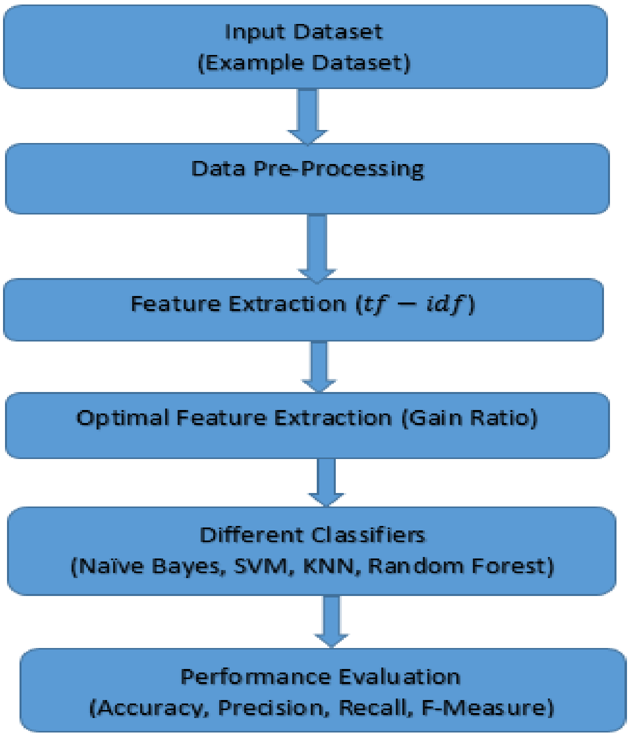
Input Dataset
(Example Dataset)
Data Pre-Processing
Feature Extraction ($tf - idf$)
Optimal Feature Extraction (Gain Ratio)
Different Classifiers
(Naïve Bayes, SVM, KNN, Random Forest)
Performance Evaluation
(Accuracy, Precision, Recall, F-Measure)

Figure 1. Proposed System Architecture

A. **Input Dataset:** The email received is always in the tree like structure of html form. It contains lots of html tags and body text. First work is to get the actual text data from the html form. This step is performed by using a “total mail convertor” [8] tool which converts all .eml (email) file to .txt file. The resulting text file contains only body text without any html tag.

B. **Data Pre-Processing:** transformation,Cleaning, tokenization and stemming is performed here. Transformation means to convert the complete dataset into desired form i.e. text file and extract main body text from html tags format. Cleaning is done by removing all the repeated words, stop words[9] and then tokenization is performed to split remaining data into tokens. Here, Porter Algorithm is used as stemming algorithm which is used to reduce the terms into their stems.

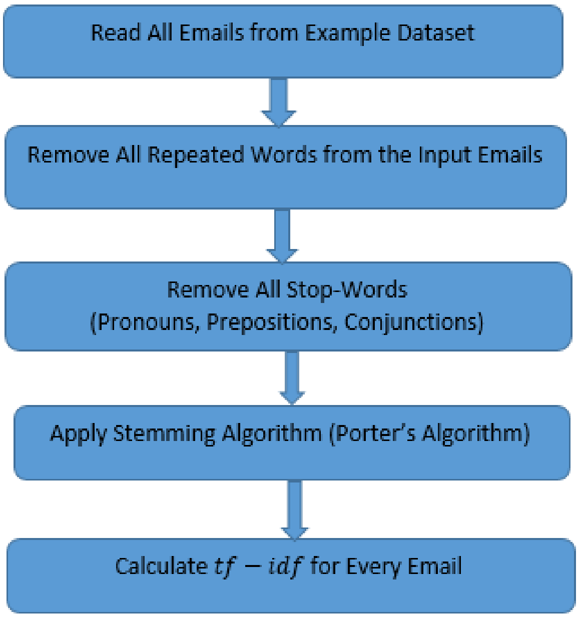
Read All Emails from Example Dataset
Remove All Repeated Words from the Input Emails
Remove All Stop-Words
(Pronouns, Prepositions, Conjunctions)
Apply Stemming Algorithm (Porter's Algorithm)
Calculate $tf - idf$ for Every Email

Figure 2. Data Pre-Processing

C. **Feature Extraction:**$tf - idf$,[10] short for term frequency-inverse document frequency is performed to filter out important words from the documents. It is most popular term-weighting scheme i.e. the ratio of the no. of times a term exists in a document is named as term frequency and the inverse ration of no. of times a term exists in all documents to the total no. of documents in the input dataset is named as inverse document frequency. The formula for$tf - idf$ is given as

$$Aij = tfij.\log\left(\frac{N}{dfi}\right)$$

- A is the vector space matrix where each elements defines a term document weight. N is the total no. of documents in the input dataset. $dfi$is the no. of documents in which ith term appears.
- On the basis of Top-K-Word approach, two term finder methods are proposed, i.e. each contains threshold of 3 and 5 respectively. The vector space matrix is prepared separately for both the term finder methods and given as an input to the Optimal Feature Selection process.

D. **Optimal Feature Extraction (OFS):**OFS is calculated on the basis of Information Gain. The Entropy (I) is used to calculate the homogeneity of an attribute, it characterizes the (im)purity of an arbitrary collection of dataset. Information Gain (G) is the expected reduction in entropy caused by partitioning the dataset according to a given attribute.

$$G(S,A) = I(S) - \sum\nolimits_{v \in V\ values(A)} \frac{|Sv|}{|S|} I(Sv)$$

E. Classifiers: Thereare many types of machine learning techniques used to classify the problems. Some of them are discussed below:

- **Support Vector Machine**:- Support Vector Machine(SVM)[11] is a supervised learning model with associated learning algorithm that analyze data used for classification, regression and other tasks like outlier detection. It is also helpful in text, hypertext categorization, image segmentation etc. This kind of binary classifier encourages researchers to apply this method to email spam filtering i.e. categorization of text document into spam or ham. SVM is used to embed the text document into vector space and apply linear classification method to create separation between two classes i.e. spam or ham in the vector space.

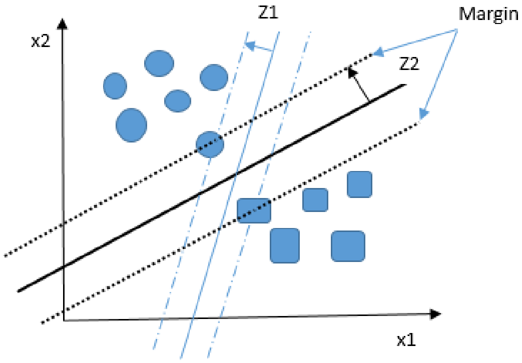

Figure 3. Support Vector Machine

An example is shown above, in which a given object can belongs to either class 1 i.e. spam (circle in the given fig) or class2 i.e. ham (rectangle in the given fig). On the basis of linear classification method, two hyper-planes are created which might classify the data. The hyper-plane whose separation or margin between the two classes is the largest;that will be chosen as best hyper-plane and the reason behind the choice of largest margin is that the misclassification errors can be minimized. So in the above example, Z2>Z1 i.e. second hyper-plane is chosen as the best as its margin between the two classes is the largest.

- **Naïve Bayes:** Naïve Bayes[12] is a more sophisticated classification method. It formulates the classification problem as a probability rule and solve it by using Bayes rule (a probabilistic approach). This approach works on prior probability, posterior probability and evidences.

Prior Probability

Posterior Probability

$$P(C = c1|[f1 = v1, f2 = v2 \ldots\ldots, fn = vn]) = \frac{[P(C = ci) * P([f1 = v1, f2 = v2, \ldots..fn = vn]|C = c1)]}{P(f1 = v1, f2 = v2, \ldots.fn = fn)}$$

Evidences

Figure 4. Probability model for Naïve Bayes

Here, C defines the classes i.e. spam or ham (in spam filtering). Prior Probability based on the prior knowledge that the particular mail is spam or ham. Posterior Probability, also known as Likelihood, defines the class of particular mail after applying the Bayes rule and Evidence is just a probability of word appears in a mail given in training data set that doesn't depends on class C . Finally, the higher probability tells the class of the input mail i.e. either ham or spam. Each input data is totally independent means presence of one feature doesn't assure anything about the presence of another feature. This is the major drawback of this classifier So it may not be applicable on real world problem.

- **K-Nearest Neighbor Classifier:** Here**,** KNN[13] stores all the available cases and the classification of new cases based on a similarity measure (eg distance function i.e. Euclidian, Manhattan, Minkowaski etc.) are proposed here. In spam filtering, a mail is classified by majority vote of its neighbors, with the mail being assigned to class (i.e. ham or spam) most common amongst its K-Nearest Neighbors measured by distance function. A large K value is more precise as it reduces the overall noise but there is no guarantee. The optimal K for most datasets has been between 3-10; it produces much better results.

- **Random Forest Classifier:** Random forest is based on bagging technique and similar to boosting technique. It develops lots of decision tree based on random selection of data and random selection of variables. Every random tree gives a class type as a result and the final result is produced by getting the maximum count of the class type i.e. spam or ham.

F. **Evaluation Measure:**[14]To determine the performance of a classifier, following measures are used:

- **Recall** specifies the number of accurately classified spam against spam.

- **Precision** determines the ratio of the numbers of emails correctly classified spam to the number of emails noticed as spam.

- **Accuracy** signifies the ratio of the number of correctly classified spam and legitimate mails to the total e-mails employed for testing.

$$Accuracy = \frac{totalno.ofclassifiedemails}{totalemailfiles}$$

- **F-Measure** signifies the harmonic sum of precision and recall.

- $F(value) = \frac{2*Precision*Recall}{\text{Precision}+\text{Recall}}$

- **Time** signifies the total time taken to provide a result.

G. System Specification: The code is implemented in java (NetBeans IDE 8.1) using JAVAML (java machine learning) libraries on window10.

III. Datasets Used

Multiple standard datasets i.e. DBWorld e-mails[15], LingSpam[16]and Enron6[17] (sixth version of Enron dataset series) are used to evaluate the performance of four different classifiers. The total number of emails are given in the form of two classes i.e. spam (unsolicited mails) or ham (legitimate emails). After applying preprocessing, the total mails are given further for the testing and training purpose.

Table I. Description of Datasets Used

Sr. No.	**Dataset**	**Spam**	**Ham**	**Total Emails**
1.	DBWorld e-mail	27	34	61
2.	LingSpam	481	523	1004
3.	Enron6	1500	1500	3000

IV. Results and Analysis

The proposed method is applied on multiple datasets by using the Optimal Feature Selection (OFS) on various classifiers. Optimal Features are selected on the basis of information gain. By applying this method, results are improved in terms of accuracy and time.

After applying term-weighting scheme i.e. $tf - idf$, the number of terms for DBWorld, LingSpam, Enron6 are 1126, 5529 and 7590 respectively. On the basis of given terms for each dataset; a separate Vector Space Matrix (VSM)[18] is created. Further, Optimal Feature Selection is applied to remove less informative features that reduces these terms to 660, 1057 and 6747 respectively. So that timing of execution can be improved. Here also; New VSM is created on the basis of these reduced terms. Information Gain threshold 0.1 is applied to DBWorld, LingSpam corpus and for Enron6 threshold value is 0.01. Both VSM and NVSM is given as an input to all four classifiers and separate reading is noted. Each classifier is run five times for the single reading given in the below table.

On analyzing the results provided in table 2 and 3. It is clearly observed that the accuracy of Naïve Bayes increases in all of three corpus and accuracy of KNN classifier improved in LingSpam and Enron6 corpus. Also, F-value of Random Forest is improved in three provided corpus.

Table 4 shows the total time taken to classify ham or spam from the complete corpus. It is clearly visible that the time taken before applying the OFS is more than the time taken after applying the OFS.

So, the motive of this research is fulfilled by improving the accuracy and time by reducing the less informative features from the corpus.

Table II. Result Before Applying OFS

Dataset/classifiers	**DBWorld emails**		**LingSpam**		**Enron6**	
	Acc	**F-value**	**Acc**	**F-value**	**Acc**	**F-value**
NB	0.8196	0.5819	0.7071	0.6702	0.8333	0.8333
KNN	0.8196	0.5264	0.7758	0.7621	0.876	0.8744
RF	0.9852	0.976	0.9978	0.8854	0.9993	0.8973
SVM	1	1	1	1	1	1

Table III. Result After Applying OFS

Dataset/classifiers	**DBWorld emails**		**LingSpam**		**Enron6**	
	Acc	**F-value**	**Acc**	**F-value**	**Acc**	**F-value**
NB	**0.8524**	**0.6579**	**0.9133**	**0.9118**	**0.8553**	**0.8552**
KNN	0.8196	0.5264	**0.9352**	**0.9346**	**0.884**	**0.883**
RF	0.9852	**0.9773**	0.999	**0.9703**	0.9997	**0.9332**
SVM	1	1	1	1	1	1

Table IV. Time (sec) Before and After Applying OFS

Time /classifiers	**DBWorld emails**		**LingSpam**		**Enron6**	
	Before OFS	**After OFS**	**Before OFS**	**After OFS**	**Before OFS**	**After OFS**
NB	0.6889	0.4534	26.201	4.767	82.059	71.039
KNN	0.5069	0.3817	37.8244	8.6784	377.4486	338.106
RF	1.2242	0.6439	8.422	2.595	16.419	15.313
SVM	0.7585	0.558	24.3148	3.6692	230.543	55.538

The three graphical representation of LingSpam corpus results are shown below in terms of accuracy, F-Measure and Time Complexity.

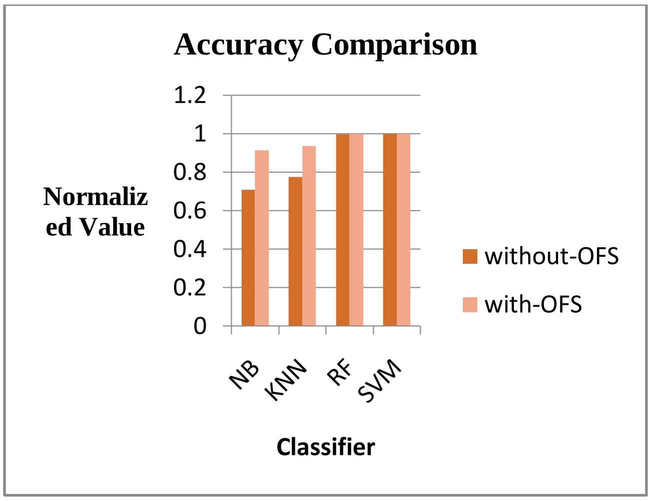

Figure 1. Accuracy Of LingSpam Corpus

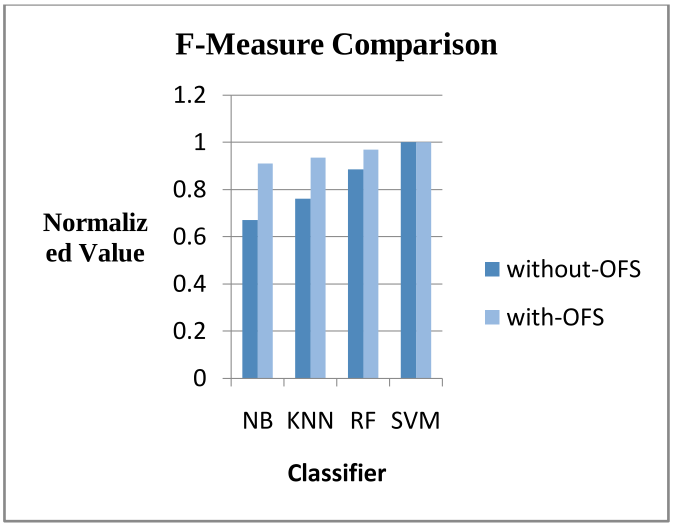

Figure 2. F-Measure Of LingSpam Corpus

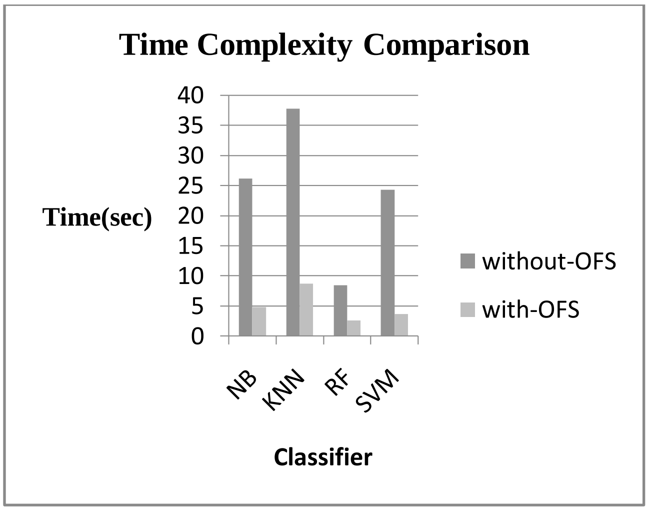

Figure 3. Time Complexity Comparison Of LingSpam Corpus

V. Conclusion

As the frequency of Spam mails are increasing day by day so there is a need of robust and effective algorithms which can increase the performance and secure the system from maliciousattack via emails. Machine Learning classifiers are more vulnerable to such kind of attacks. There is some work is directed to improve the effectiveness of these classifiers. The purpose of this research is to improve the accuracy and timing of execution which is successfully achieved. SVM and Random Forest shows the excellent output in terms of accuracy but SVM takes more time for the task of spam filtering as compare to Random Forest. Naïve Bayes and KNN also improved their performance in both the factors i.e. accuracy and time.

VI. Future Scope

In this research paper, Information Gain is used to extract the more informative features. Instead of that Chi-Square method can be used for the same task. The task can be increased up to two layers of filtering for more accurate results. Outer layer will contains the features like Email size, number of hyperlinks or type of attachments etc. and inner layers contains the more informative feature values. The corpus data can also be maximized by including the header part of the email.

VII. References

[1] J. Lyne, "Security Threat Trends," Sophos, Feb 2015.

[2] "The Spamhaus Project- definition of spam," Spamhaus Project Ltd., 2017. [Online]. Available: https://www.spamhaus.org/consumer/definition/.

[3] "Cyberoam Press Release-Cyeroam," Cyberoam releases Security , 23 march Year book for 2013. [Online]. Available: https://www.cyberoam.com/pressrelease_securityyearbook2013.html. [Accessed 2014].

[4] L. Firte, C. Lemnaru and R. Potolea, "Spam Detection Filter using KNN Algorithm and Resampling," in Intelligent Computer Communication and Processing (ICCP), 2010. R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[5] S. B. Rathod and T. M. Pattewar, "Content Based Spam Detection in Email using Bayesian Classifier," in Communications and Signal Processing (ICCSP), 2015.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[6] S. K. Trivedi and S. Dey, "A Combining Classifiers Approach for detecting Email Spam," in *Advanced Information Networking and Applications Workshops (WAINA)*, 2016.D. Kornack and P. Rakic, "Cell Proliferation without Neurogenesis in Adult Primate Neocortex," Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/science.1065467. **(Article in a journal)**

[7] T. Vyas, P. Prajapati and S. Gadhwal , "A Survey and Evaluation of Supervised Learning Techniques for Spam E-Mail Filtering," in *Electrical, Computer and Communication Technologies (ICECCT)*, 2015.

[8] "Batch EML Converter That Works," April 2017. [Online]. Available: https://www.coolutils.com/Batch-EML-Converter.

[9] "Stopwords," 2017. [Online]. Available: http://www.ranks.nl/stopwords.

[10] R. Patel and P. Thakkar, "Opinion Spam Detection Using Feature Selection," in *Computational Intelligence and Communication Networks (CICN), 2014 International Conference on*, Bhopal, India, 2014.

[11] L. Niu and Y. Shi, "Using Projection Gradient Method to Train Linear Support Vector Machines," in *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on*, Toronto, ON, Canada, 2010.

[12] N. Jatana and K. Sharma, "Bayesian spam classification: Time efficient radix encoded fragmented database approach," in *Computing for Sustainable Global Development (INDIACom), 2014 International Conference on*, New Delhi, India, 2014.

[13] A. Harisinghaney, A. Dixit and S. Gupta, "Text and image based spam email classification using KNN, Naïve Bayes and Reverse DBSCAN algorithm," in *Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on*, Faridabad, India, 2014.

[14] S. K. Trivedi, "A study of machine learning classifiers for spam detection," in *Computational and Business Intelligence (ISCBI), 2016 4th International Symposium on*, Olten, Switzerland, 2016.

[15] "UCI Machine Learning Repository DB World e-mail dataset," 2011. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/DBWorld+e-mails.

[16] "Ling-Spam Datasets - Csmining Group," 2000. [Online]. Available: http://csmining.org/index.php/ling-spam-datasets.html

[17] "Enron Spam Datasets - Csmining Groups," 19 june 2006. [Online]. Available: http://csmining.org/index.php/enron-spam-datasets.html.

"Basis Linear Algebra," [Online]. Available: https://en.wikipedia.org/wiki/Basis_(linear_algebra).