# HYST - HYbrid machine learning based e-mail Spam filtering Technique

Ramesh A, LohitKapoor
Sreenidhi Institute of Science and Technology, Hyderabad, TS

**Abstract:** E-mail is the most commonly known mode of communication across the globe; however, it comes with a problem of spamming. Email-spam is used for publicity of any product/company or any kind of malware which is undesirable in client's mailbox. To prevent such e-mails, spam filters are commonly used by a number of email service providers. These spam filters constitutes different techniques to filter out malicious emails. In this paper, proposed algorithm's outcome is based on predicted probabilities by multiple classifiers. The predicted results are used to weight the outcome in order to get more effective results. Results describe the effectiveness of the proposed algorithm.

*Keywords: Machine learning, Neural Network, Naïve Bayes, Weighted Majority Algorithm, Regression, Cloud computing*

## Introduction

Electronic mail is one of the effective tool for exchanging of information in the present era. Spam which acts as a tool for malicious attacks and elaborating unwanted content such as bug, worm and various other malwares. Email is considered as spam if there is a mismatch in user's personal data and the context of the mail is irrelevant. Moreover the use of spam mails, network bandwidth and employee productivity will be affected [1].The sending of spam messages cost is relatively low, thus the mails could be sent to huge number of recipients. There would be a gradual increment in the count of malicious mails until there is a counterpart to these kinds of emails.

Various methods such as spam filtering gateways, machine learning based techniques, artificial neural network back propagation must be identified in order to counter the growing problem of spam mails such that it would benefit for any individual or for industry. It is exciting to see whether the identified techniques is showing any impact on the spam emails and how effectively it can stop the spam messages before entering into the recipient's inbox[10].

Many experiments are being conducted on spam mails to generate algorithms which are capable of identifying spam mails. Email filtering is generally categorized on the content, which involves images, attachments, Ip address or their header that gives the data about the recipient. As the amount of spam data goes on increasing, [2] has proposed and set up the problem to stop malicious attack. There are many individuals around the globe who may respond to such type of attack would risk their financial or personal info and in order to counterpart to this, author has described few techniques. Compared to other methods, machine learning based techniques would automatically examine the body of the email and figure out the most robust model that would fix the problem of spamming. Many machine learning based methods are being used for electronic mail non-ham filtering such as: SVM and Artificial Immune System [9], Anti-spam Email Filtering [3], Comparison of Naïve Bayes and Memory based Approach [4], Naïve Bayesian and rule learning [5], Neural Networks and Bayesian Classifiers [6], Bayesian Filtering Junk Email [7], and fuzzy similarity [8].

Research had been conducted in existing methods for email spam detection, but the accuracy was quite less; hence performance needs to be improved in electronic mail spam detection. In this paper the proposed HYST is considering the outcomes probabilities obtained from different classifiers and calculating the most ideal probability of electronic mail content as ham or spam. This paper has been classified into different portions where Section II represents the related work followed by System Model (Section III) which constitutes the proposed algorithms. Further experimental setup is discussed in Section IV.

## II. Related Work

Zhuang et al.'s (2008) [11]mainly focused on detecting botnets. With the help of common keywords authors have attempted to detect fingerprints which are identical in creating spam or similar mail.

In [12],author's overview of different spam filtering techniques has been discussed and also few anti-spam protection approaches are discussed. Among those methods few methods have mainly focused on content of emails and others have also considered parameters such as length, attachments, URL, to, from, Ip, etc. Feature extraction methods are also used for image based and content-based filtering.

Webb et al.'s (2006) [13] paper, similar to email spam, researches have been made to detect web spam, with the help of email spam detection techniques. By observing URLs found in email spam messages they try to identify whether the web page is spam or not. The other method is they try to extract different features from different web pages using common keywords.

Authors in [14], studies have been made to classify emails by combining labeled and unlabeled data. While approaching different methods and algorithms in order to classify emails based on predictions vector space model VSM has produced the better output when related to other methods in terms of performance. Text classification is generally used to categorize electronic mails based on its content, based on content emails are grouped in to different folders.

Authors in [15] [16] and [17], used Enron email database to categorize emails that is based on the graph entropy model. This model tries to select interesting nodes and edges in the graph. Edges are nothing but the messages between sender email and receiver mails.

In [18], authors have gone through connections between mails and contacts or threads of mails which are conversation between two or more people happened several times. This paper has done experiments by considering Enron email database.

Cejudo et al's [19] have introduced GNUs mail for email file classification which is an open source, it includes a tool for launching certain experiments. This tool helps in analyse emails from clients and helps to analyse data mining through WIKA data mining tool. Most important decision is to select features such as subject of email, from address and to address, content etc in order to classify emails. Natural language processing helps to process and analyse those text data.

## III. System Model

In the proposed System Model as described in below Figure 1, all the incoming mails are entering into Email Servers.

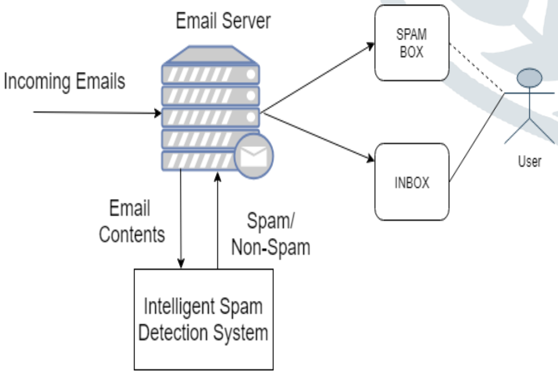

Figure 1: System Model

An electronic mail server is a system with mail transfer agent (MTA) that performs functions. Mail has been transferred among email servers that runs across aparticularprogram, which is developed across standardized protocols for managing mails and their varied content. It generally accepts mail from another mail transfer agent, a mail submission agent (MSA) along with information of transmission was evaluated by simple mail transfer protocol. When the MTA gets anE-mail and the user of thatE-mail is not hosted locally, the E-mail is transfered to other mail transfer agent. When ever it is done the mail transfer agent adds a "received" trace header on the top header of the message. Itdemonstrationsthat total mail transfer agents that have managed the mail before it reaches the users inbox. These emails are further directed towards Intelligent Spam Detecttion system.ISD is a software that is used to identifymalicious email and to stop those incoming mails fromentering into receipients inbox. More sophisticated machine learning based methods, such as Naïve Bayes filters, Random Forest or other neural network filters, try to detect spam through suspicious word patterns or frequencies as shown in figure 2.

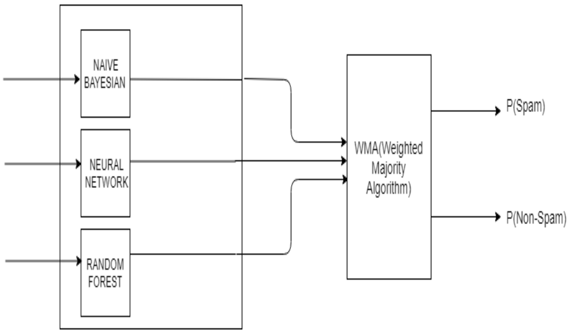

Figure 2: Intelligent Spam Detection system algorithms

In this module, we are training three classifiers i.e., Naïve Bayesian, Neural Network and Random forest for spam classification. Weighted majority algorithm is being applied to these three classifiers and based on score and threshold value we are identifying the best classifier among the three. Then by selecting the one classifier we are identifying the message is spam or ham. Algorithm 1 below explains the steps used in WMA

```
Algorithm 1: Weighted  Majority Algorithm (WMA)
Step 1:     Classifier_prob_cal (Classifier_name)
Step 2:         for ∀ (c=get_classifier(Classifier_name,
                    w1,w2,w3,w4,……$w_n$,n ) do
Step  3:            W=0; i=0;
Step 4:                 for ∀$n_i$∈ attributes
Step 5:                         P = prob($n_i$)
Step 6:                         W = $w_n$ * P
Step 7:                         arr[i] = W
Step 8:                           i++ ;
Step 9:                 end for
Step 10:                return ∑ arr[i]
Step 11:        end for
```

In the initial step we are considering one of the three classifiers. In a given email of every attribute the probabilities of each one is being listed in the dataset. Based on these probabilities we apply weights with these ones initially we assign equal weights to these classifiers. After initialization of weights we are multiplying the weights with the probabilities of every attribute. Obtained result will be stored in an array. This process will be repeated until the end of the loop.

The results generated by the Algorithm 1 is used to identify the data is malicious or not. In the stage we have obtained the final weights of every email for the three classifiers. Based on the result of each classifier, the classifier which is having the highest weight should be selected.

```
Algorithm 2: Classification
Step 1:  for ∀ Classifier(Classifier_name) do

Step 2:     Score  = $\frac{\sum \text{Classifier\_prob\_cal(Classifier\_name)}}{Total\_Classifiers}$
Step 3:     if (Score < threshold) do
Step 4:        return “non malicious”
Step 5 :    end if
Step 6:        if(Score > threshold) do
Step 7:          return “malicious”
Step 8:        end if
Step 9: end for
```

Those results are being stored in Score. In the next step we are comparing Score with that of threshold value. If score is lesser than that of threshold value it would return the email as non-malicious. If Score is higher than that of threshold value it would return the email as malicious one. Finally we are obtaining the best classifier among the three and number of spam and ham emails in the obtained dataset. The malicious mail is feed into Spam Box. This is a module which is present in email which consists of spam messages that has been sent from various oranizations.It includes Advertisements,links, etc.

## IV. Experimetal Setup and Results

In order to execute proposed model the E-mail spam dataset is obtained from UCI Machine Learning repository [20]. Spam Email database consists of 4601 Instances and Number of Attributes is 58 in which 57 are continuous and 1 nominal class label. The detailed description of Email spam dataset has been provided in the UCI repository. Based on the obtained dataset we are performing operations i.e. implementation of algorithm with the spam base dataset. The complete description of attributes is been described in below table.

Table 1: Description of attributes in the dataset

| Attributes names | Attributes type | Attribute description |
|---|---|---|
| A to AV | word_frequency_WORD | Matching of a particular word in the given mail in percentage |
| AW to BB | character_frequency_CHAR | Matching of a particular word in the given mail in percentage |
| BC | capital_run_length_average | Average length of capital letters with uninterrupted sequences. |
| BD | capital_run_length_longest | Continuous sequence of capital letters with longest length |
| BE | capital_run_length_total | Over all count of capital letters in the given mail |
| BF | Class attributes | if the e-mail is spam (1) or ham (0) |

Based on the discussed attributes we executed proposed algorithm using Python. We have calculated our results based on following parameters:

a. True positives: TP is the correct positive predictions of ham mails.
b. True negatives: TN is the correct negative predictions which are of spam mails.
c. False positives (FP): correct negative predictions i.e., spam mail as spam.
d. False negatives (FN): Incorrect negative predictions i.e., spam mail as ham

# Results:

Below Table 1 shows the confusion matrix for a classifier model on the set of test data for which the correct values are known:

Table 1: Confusion Matrix

| | | Predicted Type | |
|---|---|---|---|
| | | Spam | Ham |
| Actual Type | Spam | 10 | 12 |
| | Ham | 5 | 8 |

Along with confusion matrix following parameters are being described to know the performance of the algorithms:

**Accuracy:**

It is calculated based on total no of all correct predictions i.e., the sum of True Positives & True Negatives divided by the total ones.

**Misclassification Rate:**

It is calculated based on sum of the False Positives & False Negatives by total.

**True Positive Rate:**

True Positive rate is also called as recall. Itis evaluated as the number of right positive predictions by total number of positives ones.

**False Positive Rate:**

It is described as the number of incorrect positive predictions by total number of negative.

**True Negative Rate:**

It is computed as the number of right negative predictions divided by the total number of negatives.

**Precision:**

Precision is computed as the number of right positive predictions by the total number of positive predictions

In the Figure 3, it is evident that HYST is outperforming other algorithms by the average of 1.4% in TN, 1% in FP, 1% in TP and 1 % in FN.

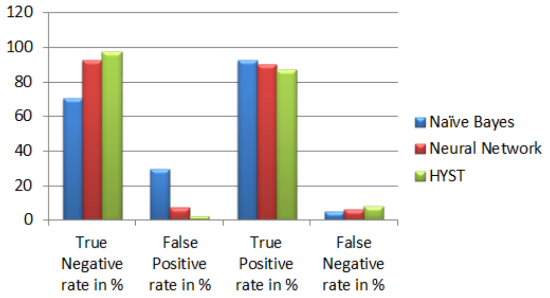

Figure 3: TN, TP FP and FN comparison

In order to evaluate the proposed algorithm on other parameters as well, Table 2 below shows the outcomes. It is evident from the results that HYST is performing exceptionally well in all fronts.

Table 2: Performance Comparisons

| | Naïve Bayes (%) | Neural Network (%) | HYST (%) |
|---|---|---|---|
| Accuracy | 79 | 91 | 93 |

| | | | |
|---|---|---|---|
| Misclassification Rate | 20 | 46 | 6 |
| True Positive Rate | 92 | 90 | 87 |
| False Positive Rate | 29 | 7 | 2 |
| True Negative Rate | 70 | 92 | 97 |
| Precision | 67 | 89 | 96 |

## V. Conclusion and Future Work

In order to solve the problems in existing E-mail spam filtering technique, the proposed work has identified a new technique that has utilized HYST algorithm to derive the emails as spam or not in the most efficient way. Precision rate has been gradually increased by the proposed algorithm. HYST performed very well with an improvement of 5%.Fututre research will be concerned with attribute selection or else feature selection for improvement in the accuracy rate because electronic mail dataset consists of huge number of attributes which are irrelevant.

# References:

[1] T. Subramaniam, H. A. Jalab, and A. Y. Taqa, "Overview of textual anti-spam filtering techniques," Int. J. Phys. Sci, vol. 5, pp. 1869-1882, 2010

[2] E.-S. M. El-Alfy, “Learning Methods for Spam Filtering,” International Journal of Computer Research, vol. 16, no. 4, 2008.

[3]Karl-Michael Schneider: “A Comparison of Event Models for Naïve Bayes Anti-Spam E-Mail Filtering.” In Proceedings of the 10thConference of the European Chapter of the Association for Computational Linguistics, Budapest, Hungary, 307-314, April, 2003.

[4] I. Androutsopoulos et al.: “Learning to Filter Spam E-mail: AComparison of a Naïve Bayesian and a Memory-based Approach.” In Proceedings of the Workshop on Machine Learning and Textual Information Access, Pages 1-13, 2000..

[5] J. Provost, “Naïve-Bayes vs. rule-learning in classification of email,” The University of Texas at Austin, Department of Computer Sciences, Technical Report AI-TR-99-284, 1999.

[6] Y. Yang, S. Elfayoumy, “Anti-spam filtering using neural networks and Bayesian classifiers,” in Proceedings of the 2007 IEEE International Symposium on Computational Intelligence in Robotics and Automation, Jacksonville, FL, USA, June 2007.

[7] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, “A Bayesian approach to filtering junk e-mail,” in Proceedings of AAAI’98 Workshop on Learning for Text Categorization, Madison, WI, July 1998.

[8] E.-S. M. El-Alfy and F. S. Al-Qunaieer, “A fuzzy similarity approach for automated spam filtering,” in Proceedings of IEEE International Conference on Computer Systems and Applications (AICCSA’08), Qatar, April 2008.

[9] K. Jain, “A Hybrid Approach for spam filtering using Support Vector Machine and Artificial immune System pp. 5–9, 2014.

[10] Le Zhang, Jingbo Zhu, Tianshun Yao: “An Evaluation of Statistical Spam Filtering Techniques.” ACM Transactions on Asian Language Information Processing, Vol. 3, No. 4, Pages 243-269, December, 2004.

[11] Zhuang, L., Dunagan, J., Simon, D.R., Wang,H.J., Tygar, J.D., Characterizing Botnets from Email spam Records,LEET’08 Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent threats Article No.2.2008

[12] Enrico Blanzieri, Anton Bryl, A survey of learning-based techniques of email spam filtering, Technical Report # DIT-06-056. 2008

[13] CloseSteve Webb, James Caverlee, CaltonPu. Introducing the Webb Spam Corpus: using Email spam to identify web spam automatically, CEAS.2006.

[14] Sculley, D., Gabriel M. Wachman, 2007. Relaxed online VSMs for spam filtering, SIGIR 2007 Proceedings

[15] The Enron corpus: a new dataset for email classification researchECML (2004), pp. 217-226

[16] JiteshShetty, JafarAdibi, 2005. Discovering Important Nodes through Graph Entropy the Case of Enron Email Database, KDD'2005, Chicago, Illinois.

[17] Shinjae Yoo, Yiming Yang, Frank Lin, I1-Chul Moon, 2009.Mining Social Networks for Personalized Email Prioritization, KDD'09, June 28-July 1,Paris, France

[18] The Enron corpus: a new dataset for email classification research ECML (2004), pp.217-226

[19] Using GNU smail to compare data stream mining methods for online email classification J.Mach. Learn. Res. Proc. Track, 17 (2011),pp. 12-18

[20] UCI Machine Learning Repository:

https://archive.ics.uci.edu/ml/datasets/spambase