Intelligent Model for Classification of SPAM and HAM

Amandeep Singh Rajput, Vijay Athavale, Sumit Mittal

Abstract: In our study, we propose a collaborative approach by using cluster computing with the help of parallel machines for fast isolation of SPAM and HAM. A cluster approach can increase the computing power many folds with existing hardware and resources thus by increasing the speed of processing without incurring any extra cost. In this study, we only use header based filtering method, thus by keeping the privacy of the user intact. The standard test set for HAM and SPAM from Spam Assassin [1][2] is used. Two types of parallel environments are used in this research. First is where multiple Anti Spam methods are used in the parallel environment against the test corpora and false positive and false negative accuracy recorded. The second parallel environment is where standard test corpora are divided into parts and fed into parallel machine environment with single anti spam method used at all machines and the time saving is recorded against standalone machine being used. Weka Data Mining Software is used to apply the anti-spam methods (available at http://www.cs.waikato.ac.nz/~ml/weka/) [3].

Keywords: ***SPAM, HAM, Classification***

I. INTRODUCTION

Electronic mail is still a very cost effective communication method but is also targeted by hackers to use it as a method to spread the virus, phishing, malicious code, unnecessary advertising etc. Email is very handy and convenient to use but is also misused by many. Although other communication media like Facebook, Twitter, and chat programs have come there is a steady growth in the use of email and also the number of spam have increased. The usage of e-mail service has grown at a exponential rate. There was an increase of 5% since 2010 which on an average amounts to 538.1 million messages sent daily during 2015 [4]. The volume of spam was 89-92 % during 2008 (Message Anti Abuse Working Group) [5]. The growth of Spam has been phenomenal 7% over last 3 years [6].

Spam emails are becoming a major concern to email users and are adversely affecting the email users. HAM is a genuine mail that recipient is intended to receive and SPAM is a spurious mail that is sent from unreliable sources in bulk to thousands of users with mala fide intentions. Spam is sent intentionally to users where the receiver is not supposed to receive it. Spam operators try to extract sensitive information of a user by luring them with attractive offers.

As the Email user base is very huge, therefore the damage in terms of time wasted in opening these emails, sensitive information leaked and efforts in segregating these mails from HAM mail is also very huge. There is no consensus regarding the financial cost incurred due to spam.

But during a survey USA alone incurred loss of $40 billion due to spam in 2003 [1]

As there are large numbers of methods to separate HAM and SPAM, it is difficult to narrow down to a single method that can generate very less false positive rate. Though there are many methods and algorithms to counter spam problem, no single method or algorithm is perfect. Many HAM mails are marked as SPAM during the isolation process. This situation of wrongly isolating HAM as SPAM is termed as False.

Positive. In another situation, many SPAM mails are marked as HAM. This situation of wrongly isolating SPAM as HAM is termed as False Negative. Machine Learning Self adaptive methods like SVM, HMM and also Non Machine Learning methods like KNN are quite effective. In this research, two separate scenarios are explored. First is a collaborative approach where multiple methods are used. Both methods machine and non machine learning are used in a parallel environment to increase the efficiency and accuracy of separating Ham and Spam mails. The accuracy is increased by reducing the False Positive and False Negative Rate and the efficiency is increased by using parallel environment. The second scenario is to use single spam segregation method multiple times in a parallel environment to increase the speed of SPAM and HAM isolation.

This paper is organized in six different sections. The first section focuses on the related work in the areas of machine and non machine learning methods used for email classification as HAM or SPAM. The second section takes an insight of the collaborative approach for spam controlling and the related work in this area. The third section talks about the spam avoidance techniques i.e. either to detect spam on the basis of the origin or detect it once it is received at the mail server.

The fourth section is about feature extraction from an Email Header. This section highlights the features like to, from, cc, bcc, subject that are to be extracted. It also shows the adaptive approach for header feature extraction based on the length of the email subject header. This section also takes a look at generating new rules and collaborative approach for feature extraction. The section five presents information about Email Corpus Analysis for classification. Various corpus are analyzed for suitable selection for this research. Four corpus are used together for a wider scope. The last section the sixth section analysis the collaborative approach for classifying emails as HAM and SPAM.

Revised Manuscript Received on December 22, 2018.

Amandeep Singh Rajput, GGI, Khanna, Punjab
Vijay Athavale, ABESEC, Ghaziabad
Sumit Mittal, MMICT & BMMMDU, Ambala

This section explores the use of complete corpus with different machine and non machine algorithms in a collaborative manner for classification of emails.

II. PROPOSED METHODOLOGY

Section I-Machine and Non Machine learning methods

There are two major categories Machine Learning and Non Machine Learning used for filtering Spam. Non Machine learning methods use White-Listing, Black-Listing and keyword search approach to filter out spam [7]. Non Machine Learning approach is easy to implement and experiment, therefore there are high chances of spammers bypassing through Non Machine Learning approaches. But strong keyword searching and constant updating of whitelist and blacklist can still have a higher rate of success. Machine Learning Approach highly matches with text characterization, thus attracts the interest of researchers. Researchers have applied many Machine Learning approaches like support vector machines, memory-based learning, Ripper rule-based learning, boosting decision trees, rough sets, neural networks, Bayesian classifiers, and fuzzy logic [7]. Most email classification approaches follow single algorithm text classification method. Rough set theory or rough set theory based methods are most popular among the researchers for email classification. Emails are not written as they are supposed to be written when they are used for the malicious task. In these cases, emails are hard to classify, but rough set theory handles email which is incomplete, inconsistent, imprecise pretty well [8]. Some researchers used data mining approach like support vector machine (SVM), neural network, naive Bayesian or rough set theory. But the problem was that these methods were used with a fixed set of rules [8]. Researchers like Chouchoulas, Zhao and Zhang, Zhao and Zhu only experimented with keyword frequency for spam rule generation.

Section II - Collaborative approach

Many solutions are given for spam classification under standalone servers. Some of the solutions provided on standalone servers are Counter Attack, Opt-Outlist, Spam filter. Standalone servers have limited computing power and speed, thus it is difficult for standalone servers to filter out a new type of spam mails [8]. Any new type of spam mail requires new rules to be generated and used. The collaborative approach has the computing power and the necessary speed to generate and use new anti spam rules as and when required.

Section III - Spam avoidance

SPAM can be dealt in two ways, i.e. either check and block spam at originating place or while receiving the mail, check and classify it as HAM or SPAM. Servers that allow another server to use them as an intermediate channel to forward messages are targeted by spammers. These unattended servers with low security are termed as Botnets [9][10]. Due to use of Botnets by spammers and constantly changing originating place, it is difficult to check spam at originating place itself. There are some servers that are black listed for spreading SPAM or being used as a channel to spread SPAM. The IP addresses of these servers are black listed and distributed to mail servers. Mails coming from servers with blacklisted IP addresses are termed as SPAM straightway without applying any kind of analysis. Since the source is not trusted, the mails are directly termed as spam. Many open proxy servers are present on the internet. These servers are also targeted by the spammers to spread SPAM. In case of open proxy server used by spammers it becomes difficult to identify the actual source of email. The second method of checking email and classifying it as HAM and SPAM is once the email is received at the mail server. This method is an effective method and is used in this research.

Section IV - Feature Extraction of Email Header.

Email classification is done on the basis of header features thus by keeping the content privacy of the sender intact [11]. In many countries email content scanning is an illegal process. Therefore we confine ourselves to header extraction process only. It is shown that the accuracy of classification of emails as HAM or SPAM by using only header features is as good as using the content of the email. In case of non-content base spam classification, header feature based spam filtering is most effective and accepted [12]. Scanning the emails by content not only breaches the privacy of the sender but also slows down the process of classification of emails. Header Features like From, To, CC, BCC, Subject, parts of the subject, words used in the subject, number of characters used in a subject are used as input to email classification method. This research tries to improve the accuracy of the classification by extracting more number of header features. The algorithm used here adapts itself to extract features of long headers, short headers, and smartly written headers. Spammers are cleaver to manipulate words, writing words cleverly to avoid detection. Words like Money written as M0ney, mo.ney, m o n e y, mooney etc. are some of the tricks used by spammers. Header Features Extraction is an important aspect of this research. Header features like From, To, CC, BCC, attachment, subject are common features that are used to scan email. It is easy to classify mails coming from a known spam source, but the challenge is to classify mails that look genuine but are not genuine. It is also suspicious if there are multiple numbers of receivers of the mail. Carbon Copy (CC) and Blind Carbon Copy (BCC) if used, then very likely it can be classified as SPAM. This is because spammers target is to spread SPAM as much as possible and BCC can hold many recipient addresses [13]. Genuine mails have all the parameters like recipient address in ‘To’ field, a decent subject of a line or two. Genuine mails normally do not use BCC field of the email. On the other hand, spammers use BCC to reach out to more number of email users and also there is no restriction on the number of emails address that can be used in the BCC field [12]. The research aims at an adaptive and collaborative approach to develop rules based on corpus analysis to make SPAM and HAM apart. The feature extraction from the subject line of an email is most crucial. The following rules are followed to extract features from the subject of an email to mark them as SPAM.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

a. Algorithm

- ○ Unigram approach:- Single word matching words with suspicious words like the lottery, free, gift, Adult, sex etc.
- ○ N-Gram approach:- Multiple words or combination of words matching with suspicious words like free credit card, buy for free, give bank details, Congrats First winner etc.
- ○ No subject.
- ○ Long subject line, more than 250 words.
- ○ Count frequency of each word, suspicious if the frequency of a suspicious word is high.
- ○ Gaps used in words like money written as M O N E Y.
- ○ Deliberately misspelled words like Medicine written as Med1cine.
- ○ Identifying cleverly written subject lines "Get Insurance amount without Insurance".
- ○ For receiver address, BCC field used instead of To field.
- ○ Rules are also generated while training the algorithm from the already labeled data.
- ○ If no rule matches, then algorithm adapts itself and uses KNN (K-Nearest-Neighbor) method to find out how near it is to a SPAM classification rule. If it is nearer to a rule and within acceptable distance D, then it can be classified as SPAM. In case the rule does not fall near to any SPAM rule then a new rule is created by taking header feature and classifying it as HAM.

The figure-3 shows a collaborative approach that is followed in header feature extraction. Since header feature extraction is a time consuming process, we propose here a collaborative approach where email header is fed to the parallel machine for faster header feature extraction. Header features from each mail are extracted and stored separately and in the next stage HAM, SPAM isolation is done.

b. Block Diagram of work

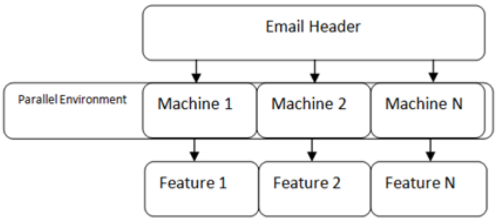


Fig. 1 Block Diagram for feature extraction

Section V – Email Corpus Analysis for classification

Corpus selection is one of the major factors for classification accuracy. Corpus can be generated by collecting emails from reliable email users or collected from standard corpus. Some of the reliable corpus available are SpamAssassin, Ling-Spam, PU1 to PU3, Spam-Base.

Table. 1 Reliability vs Acceptability

Corpus	% age of usage in experiments	Reliability and Acceptability
Own Generated	10	Low
Spam-Assassin	50	High
Ling-Spam	20	Medium
CSDMC-Spam	20	Medium

These corpus sources have been effectively used by many researchers. Out of these Spam-Assassin corpus has been widely used and accepted by researchers for HAM and SPAM classification. More than 9000 messages are present in Spam Assassin. Even percentage of SPAM and HAM is suitable for experimental results.

The research proposes corpus from various places instead from one place.

- Own generated corpus contains 1000 emails from previous six months. Emails collected in such a way are manually classified first as HAM and SPAM.
- Spam Assassin is a standard corpus used by many researchers. Even number of HAM and SPAM mails are present in this corpus. In this research 9000 emails are used for classification.
- Ling-Spam is also a standard corpus that is available at http://csmining.org [14].
- CSDMC-Spam Base is another email corpus that is available at http://csmining.org.

Section VI – Collaborative approach for email classification

Weka data mining tool is to be used to apply machine learning and non machine learning methods on the corpus. A collaborative approach is followed by creating a parallel environment by combining hardware machine in a cluster format. The first collaborative approach, as shown in figure-1, uses multiple methods. Each machine takes care of a method with complete dataset. Multiple rules improve the spam filtering accuracy [8].

In this research, both supervised as well as non supervised machine learning methods are used. The supervised machine learning methods will be used to classify the emails at the first stage. For classifying the emails that are not classified in the first stage, non- supervised machine learning methods like clustering will be used.

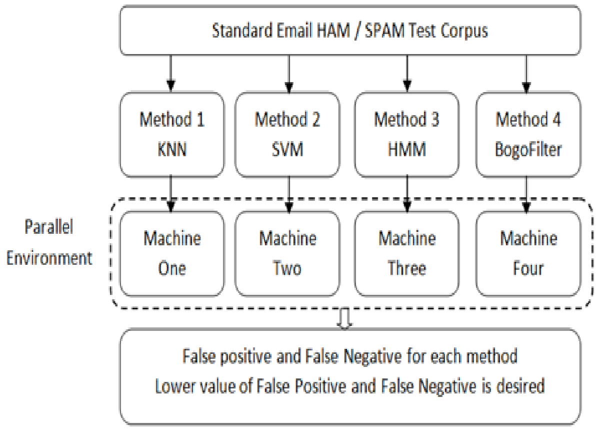

Standard Email HAM / SPAM Test Corpus
Method 1
KNN
Method 2
SVM
Method 3
HMM
Method 4
BogoFilter
Parallel
Environment
Machine
One
Machine
Two
Machine
Three
Machine
Four
False positive and False Negative for each method
Lower value of False Positive and False Negative is desired

Fig. 2 Flow chart for separating HAM & SPAM

This research also apply an innovative idea to tag each email with likely percentage of email being SPAM, instead of out rightly classifying it as SPAM or HAM. If a SPAM tag percentage is zero that means the email is a genuine and is a HAM.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

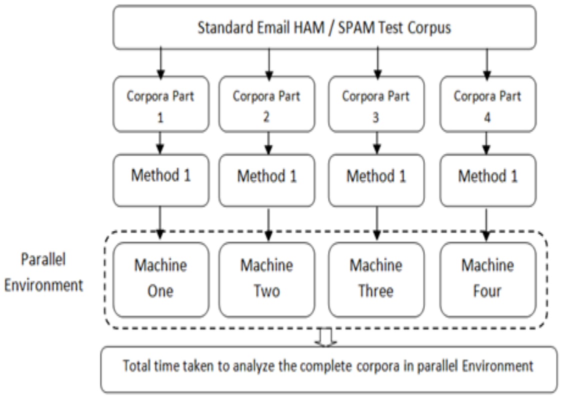


Fig. 3 Single SPAM segregation

In case SPAM tag percentage is 100 that means the email is SPAM. Every extracted feature which matches the rule adds to the percentage. The adaptive algorithm gives percentage 25%, 50%, 75% to emails scanned for classification. The percentage at the higher end indicates that, though the email is not marked as SPAM, very likely it is a SPAM. The percentage at the lower end indicates that, though the email scanned is not marked as HAM but very likely it is a HAM. A mail tagged as 50% needs more analysis before and the decision can be made. Figure-2 takes the approach of dividing the corpora into N equal sets and feeding the corpora to the parallel environment. Each machine in the parallel environment is provided with an email training data set and an email classification data set. The classification method that can be chosen depends on the accuracy required. In this research Decision Tree, Rule Based, Rough Set Theory, Naïve Based classifiers are used.

The Table -2 shows deciding parameter for each email. The HAM emails with zero as SPAM percentage are correctly classified emails as HAM. The HAM emails with 100 as SPAM percentage are false positive where in a HAM genuine email is classified as SPAM. In either case, HAM or SPAM if the percentage is 50% then it is a Not Decided case. The cases where SPAM percentage for emails is 25, 50 or 75%, more analysis is required. In these cases, each email needs to be reanalyzed with new rules. New rules can be generated with the help of KNN method where in emails in these cases are clustered with correctly classified emails and reclassification is done.

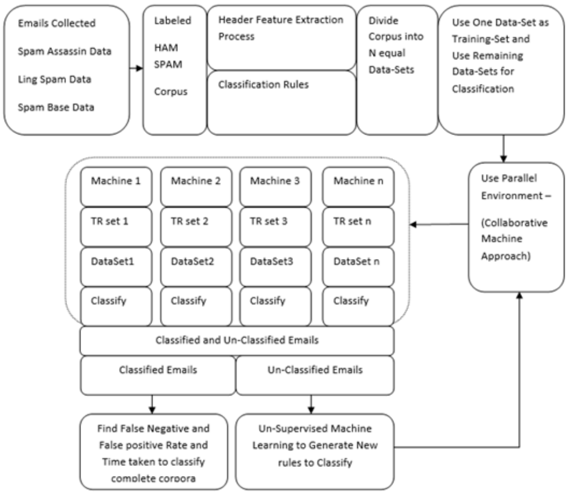
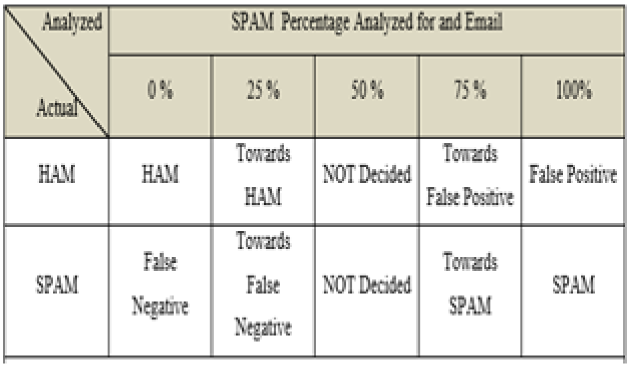

Emails Collected
Spam Assassin Data
Ling Spam Data
Spam Base Data
Labeled
HAM
SPAM
Corpus
Header Feature Extraction Process
Classification Rules
Divide Corpus into N equal Data-Sets
Use One Data-Set as Training-Set and Use Remaining Data-Sets for Classification
Machine 1
Machine 2
Machine 3
Machine n
TR set 1
TR set 2
TR set 3
TR set n
DataSet1
DataSet2
DataSet3
DataSet n
Classify
Classify
Classify
Classify
Use Parallel Environment – (Collaborative Machine Approach)
Classified and Un-Classified Emails
Classified Emails
Un-Classified Emails
Find False Negative and False positive Rate and Time taken to classify comolete coroora
Un-Supervised Machine Learning to Generate New rules to Classify

Fig. 4 Process of classification

III. RESULT ANALYSIS

The complete classification process is explained in figure 3. The email corpus selected is already labeled as HAM or SPAM. The corpus is used in the parallel machine for faster header feature extraction. Classification rules are already fed into the database. The classification starts by dividing the corpus into N equal sets and feeding each training set and to be classified set to the parallel environment shown in figure 4. The variable N is adjustable to fine tune the accuracy.

Table. 2 Decision table for Classification



Analyzed / Actual	SPAM Percentage Analyzed for and Email				
	0 %	25 %	50 %	75 %	100%
HAM	HAM	Towards HAM	NOT Decided	Towards False Positive	False Positive
SPAM	False Negative	Towards False Negative	NOT Decided	Towards SPAM	SPAM

IV. CONCLUSION

The sets so formed are further used as training set and the classification set. The process is to use the first set as training set and the remaining N-1 sets as the sets to be classified. In the next iteration the second set is used as the training set and the remaining sets are sets to be classified. The process is repeated until all the sets are used as training sets. The emails are classified based on the spam percentage each mail gains. The zero spam percentage for a mails indicates that the mail is genuine i.e. it is a HAM and if the spam percentage found is hundred percent then it indicates that mail is SPAM. The accuracy of the classifier is found by comparing the result with already labeled mail.

REFERENCES

1. The Apache Software Foundation, "http://spamassassin.apache.org/downloads.cgi," 2015. [Online]. Available: http://www-us.apache.org/dist//spamassassin/source/Mail-SpamAssassin-3.4.1.tar.gz. [Accessed 20 August 2017].
2. GitHub, Inc, "SpamAssassin," 21 April 2016. [Online]. Available: https://github.com/dmitrynogin/SpamAssassin.git. [Accessed 20 August 2017].
3. U. o. Waikato, "Weka 3: Data Mining Software in Java," Machine Learning Group at the University of Waikato, 2017. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/. [Accessed 20 August 2017].
4. D. R.-O. F. F.-R. a. J. R. M. N. Pérez-Díaz, "Boosting Accuracy of Classical Machine Learning Antispam Classifiers in Real Scenarios by Applying Rough Set Theory," Scientific Programming, Vols. 2016, 10 pages, 2016, no. Article ID 5945192, pp. 1-10, 2016.
5. Clotilde Lopes a, Paulo Cortez a,⇑, Pedro Sousa b, Miguel Rocha b, Miguel Rio c, "Symbiotic filtering for spam email detection," Expert Systems with Applications, vol. 38, no. 8, pp. 9365-9372, 2011.
6. Francisco Salcedo-Campos, JesúsDíaz-Verdejo, Pedro García-Teodoro, "Segmental parameterisation and statistical modelling of e-mail headers for spam detection," Information Sciences, vol. 195, pp. 45-61, 2012.

7. El-Sayed M. El-Alfy, Radwan E. Abdel-Aal, "Using GMDH-based networks for improved spam detection and email feature analysis," Applied Soft Computing, vol. 11, no. 1, p. 477–488, 2011.
8. Gu-Hsin Lai a,*, Chia-Mei Chen a, Chi-Sung Laih b, Tsuhan Chen c, "A collaborative anti-spam system," Expert Systems with Applications, vol. 36, no. 3, pp. 6645-6653, 2009.
9. YinglianXie, Fang Yu, Kannan Achan, Rina Panigrahy, Geoff Hulten, Ivan Osipkov, "Spamming Botnets: Signatures and Characteristics," ACM 978-1-60558-175-0/08/08, Seattle, Washington, USA., 2008.
10. Jessa dela Torre, Sabrina Lei Sioting, "Spam and All Things Salty: Spambot v2013," 03 12 2013. [Online]. Available: https://www.botconf.eu/wp-content/uploads/2013/12/03-JessadelaTorre-SpamBot-paper.pdf. [Accessed 20 August 2017].
11. ZhenhaiDuan a, KartikGopalan b, Xin Yuan a, "An empirical study of behavioral characteristics of spammers: Findings," Computer Communications, vol. 34, p. 1764–1776, 2011.
12. C. G. a. E. N. b. M. L. c. S. C. Yong Hua, "A scalable intelligent non-content-based spam-filtering framework," Expert Systems with Applications, vol. 37, no. 12, pp. 8557-8565, 2010.
13. C.-C. Wang, "Sender and Receiver Addresses as Cues for Anti-Spam," Journal of Research and Practice in Information Technology,vol. 36, no. 1, Feb 2004.
14. "Ling-Spam datasets," CSMINING GROUP, 17 July 2000. [Online]. Available: Csmining.org. (2017). Ling-Spam datasets - Chttp://csmining.org/index.php/ling-spam-datasets.html. [Accessed 30 July 2017].

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*