Volume 8, No. 5, May – June 2017
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Spam Mail Detection Using Hybrid Secure Hash Based Naive Classifier

Sana Ajaz
Department CSE
Jamia Hamdard
New Delhi, India

Md. Tabrez Nafis
Department of CSE
Jamia Hamdard
New Delhi, India

Vishal Sharma
Department of CSE
Bharti Vidyapeeth College of Engineering
New Delhi, India

Abstract: E-mail is the most prevalent approaches for communication because of its obtain ability, quick message altercation and low distribution cost. Spam mail seems as a serious issue influencing this application today's internet. Spam may contain suspicious URL's, or may ask for financial information as money exchange information or credit card details. Classification is a way to get rid of those spam messages. Naïve byes classification based spam filtering technique is a popular method. In this work a detection of spam mail is proposed by using Naïve byes classification method by combining secure hash algorithm (SHA-512) as security purpose. Experimental results present a significant improvement in accuracy with higher F-measure compare to traditional algorithms.

Keywords: spam mail detection, SHA-512, Naive byes classification etc.

I. INTRODUCTION

Various researches are proposed for spam filtering by classifying them into labels of spam and business messages. Also SVM based classifications are also used. K-nearest neighbor classification is simple, straightforward and easy to implement and has high F-measure compare to Bayesian and SVM classification. But accuracy of traditional SVM and KNN is lower than Naïve byes classification. Emails have become one of the most frequently used methods for cyber-attacks. The supreme disturbing email-based attack is Targeted Malicious Email [1] [2]. In TME, attackers send malicious emails to certain people targeted in an organization, such as executives of large companies, high-ranking government personnel, military officials and even famous researchers, in order for the attackers to obtain valuable confidential information and latest research of the targeted people. In TME, an email often has an attachment with malicious codes that can be installed automatically upon opening without the victims realizing it. In some cases, the victims' computer will become the back door for the attackers who in turn have the authority to enter the network of the targeted persons and thus steal confidential information. One more characteristic email-based cyber-attack is the malicious spam email attack, which goals to spread many emails with Uniform Resource Locator (URL) links leading to malicious websites. Previously, malicious codes were sent through the attachment of such spam emails. Though, numerous positive filters have been industrialized to notice malicious attachments. Thus, attackers are now turning to malicious spam campaigns that attack using the links attached in the emails. According to the Symantec annual report in 2014 [3], about 87 percent of scanned spam messages contained at least one URL hyperlink. Moreover, recent findings by Symantec [4] show a sharp rise of emails containing malicious links, from 7% in October 2014 to 41% in the following months. Apart from that, currently, attackers also use more relevant email contents [1] that are specific to their victims' line of work, besides addressing the name of the recipient in the email body to convince the victim that the email received is a normal email. For instance, a fake email notification regarding a conference or journal targeted towards a recipient with academic status, notifications regarding false documents such as telecommunication service bills, fax and voicemail in which the victims are given a link to get more information [4].

II. RELATED WORK

For the estimation of maximum possibility of nearest neighbor classifier they estimate the enactment of the main possibility support vector machine, nearest neighbor classifier, which is a development, concluded the SVM nearest neighbor classifier on the assignment of spam filtering. In instruction to categorize a instances, it first chooses k samples to train an SVM model which is used to variety decisions. As such, the SVM nearest neighbor procedure suggests no rule for selecting the parameter completed an attempt to evaluation k by internal training and testing on a training data, but this method brought indeterminate consequences.[5] Through privacy responsive cooperative spam clarifying a large privacy conscious cooperative spam filtering method ALPACAS was calculated in instruction to identify spam effectively .But they were based on the content in e-mail. The spammers tried to defeat this algorithm by inserting a random paragraph like structures into the e-mail messages which does not detect spam mails effectively in e-mail.[6] Mail ranking method which considered e-mail address of senders by ranking priority if it is detected as a spam by its content with its two mail rank variants basic mail rank and personalized mail rank. Trust and reputation algorithms have become increasingly popular to rank .Building upon the e-mail network graph; a power iteration algorithm is used to compute the score of e-mail address. This process does not have much scalability and faster executable and is more complicated.[7] Markov random chain process uses the incoming mails with its contents are only identified and uses its weighting scheme for spam filtering .As there is a chance of inserting a random paragraph into it while only contents are taken. This has a drawback of storage utilization is high and not efficient to use [8]. In false positive safe neural network an approach of online cumulative training is proposed .If a system would learn each time something new it arrives, the phrases are

rephrased after sometime if the features are not too good the system would correctly recognize as a spam .This can be done on client side which makes a lot of work for user. The neural network defines whether the patterns are important high false positive rates are achieved [9].In the paper [10] authors introduced a novel task to computational linguistics and machine learning: determining whether a news-wire article is ―true or satirical. The authors found that the combination of SVMs with BNS feature scaling achieves high precision M T Nafis et al[11] studied that as against the popular notion that the users with maximum social connection might not be the actual influencers. The Page Rank algorithm does not take into account the enthusiasm of users actually contributing in the information propagation by rewetting the posts shared by the content generator. More the number of rewets by multiple users in the follower graph is better than the Influencing capability of the user. Mangal Singh, M Tabrez Nafis [12] demonstrated sentiment classification and scaling with similarity evaluation among reviews. Review data is pre-processed and cleaned for with similarity evaluation among reviews. Review data is pre-processed and cleaned for are used to transform reviews to intermediate form.

III. STATEMENT OF THE PROBLEM

This research finally wants to a spam filter, that is: a decision function f, that would express us whether an assumed e-mail message m is spam (S) or genuine mail (L). If we denote the set of all e-mail messages by M, we may state that we search for a function f: M→ {S, L}. We shall look for this function by training one of the machine learning algorithms on a set of pre-classified messages {(m1, c1),(m2, c2), . . . ,(mn, cn)}, mi ∈ M, ci ∈ {S, L}. This is closely a wide-ranging declaration of the typical ML problem. There are, however, two special aspects in our case: we have to extract features from text strings and we have some very strict requirements for the precision of our classifier [15].

Extracting features

The objects we are trying to classify are text messages, i.e. strings. Strings are, unfortunately, not very convenient objects to handle. Best of the ML (machine learning)procedures can only categorize arithmetical objects (real numbers or vectors) or otherwise require some measure of similarity between the objects (a distance metric or scalar product). In the first case we have to convert all messages to vectors of numbers (feature vectors) and then classify these vectors. For instance, it is very usual to income the vector of numbers of occurrences of confident words in a message as the feature vector. When we extract features we usually lose information and it is clear that the way we define our feature-extractor is crucial for the performance of the filter. If the structures are selected so that there might exist a spam message and a genuine mail with the similar feature vector, then no substance how good our machine learning algorithm is, it will make mistakes. On the other hand, a wise choice of features will make classification much easier (for example, if we could choose to use the "ultimate feature" of being spam or not, classification would become trivial). It is worth noting, that the features we extract need not all be taken only from the message text and we may actually add information in the feature extraction process. For example, analyzing the availability of the internet hosts mentioned in the Return-Path and Received message headers may provide some useful information. But once again, it is much more important what features we choose for classification than what classification algorithm we use [15]. Oddly enough, the question of how to choose "really good" features seems to have had less attention, and I couldn't find many papers on this topic. Furthermost of the time the simple vector of word frequencies or somewhat similar is used. In this article we shall not emphasis on feature extraction either. In the subsequent we will signify feature vectors with letter x and we use m for messages.

IV. METHOD USED

A. *Naive Bayes*

The naïve Bayes model usages an NB classifier as its classification model for spam mail responsibilities. Assuming the features are independent given the class, the probability of a certain class given all of the features p(Cj|f1,f2, …fn) can be found by computing where both p(Cj)and p(fi|Cj) can be projected from training data (Cj refers to class j, fi denotes to feature i), the class with the upper most possibility will be confidential as the predicted class. For imperfect data, the probability design and classification manufacture are calculated over experiential data (the subscript o in the subsequent equation designates experiential values), which is an actual way to handle missing values when there are enough observed data to make reliable classifications [13].

Class=$argmax\ p(class_j)\prod_0 p(X_0 = x_0 \backslash class_j)$ (1)

The Laplace Estimator can be used to smooth the probability calculation and avoid a conditional probability of 0.

$P(X_i = x_i|Y = y)=\frac{\#(X_i=x_i,Y=y)+1}{\#(Y=y+|X_i|)}$ (2)

Where |Xi| is the size of the set {Xi}. For an example of binary class, P(Xi=0|Y=1)=0/2 will be (0+1)/(2+2)=1/4, P(Xi=1|Y=1)=2/2 will be (2+1)/(2+2)=3/4 using the Laplace Estimator [15].

B. *SHA-512*

SHA-512 is a variant of SHA-256 which operates on eight 64-bit words. The message to be hashed is first

(1) Padded with its length in such a way that the result is a multiple of 1024 bits long, and then

(2) Parsed into 1024-bit message blocks $M^{(1)}$,$M^{(2)}$,……$M^{(N)}$.

The blocks of message are treated one at a time: Opening with a secure original hash value $H^{(0)}$, consecutively calculate

$H^{(i)} = H^{(i-1)} + C_{M^{(i)}}(H^{(i-1)})$ (3)

Where C is the SHA-512 compression function and + means word-wise mod 2^{64} addition. $H^{(N)}$ is the hash of M [14].

The SHA-512 compression function operates on a 1024-bit message block and a 512-bit intermediate hash value. It is fundamentally a 512-bit block cipher procedure which encrypts the middle hash value using the message block as key. Therefore there are two main components to describe: (1) the SHA-512 compression function, and (2) the SHA-512 message schedule. We will use the following notation:

$\oplus$	bitwise XOR
$\wedge$	bitwise AND
$\vee$	bitwise OR
$\neg$	bitwise complement
$+$	mod 2^{64} addition
R^n	right shift by n bits
S^n	right rotation by n bits

Table 2: Notation

For SHA-512, all of these operators act on 64-bit words.

The initial hash value $H^{(0)}$ is the following sequence of 64-bit words (which are obtained by taking the fractional parts of the square roots of the first eight primes):

$$H_1^{(0)} = \texttt{6a09e667f3bcc908}$$
$$H_2^{(0)} = \texttt{bb67ae8584caa73b}$$
$$H_3^{(0)} = \texttt{3c6ef372fe94f82b}$$
$$H_4^{(0)} = \texttt{a54ff53a5f1d36f1}$$
$$H_5^{(0)} = \texttt{510e527fade682d1}$$
$$H_6^{(0)} = \texttt{9b05688c2b3e6c1f}$$
$$H_7^{(0)} = \texttt{1f83d9abfb41bd6b}$$
$$H_8^{(0)} = \texttt{5be0cd19137e2179}$$

FEATURES

The algorithm is used to compute a message digest for a message or data file that is provided as input.

The Text/message or data file should be measured to be a bit sequence.

The length of the message is the number of bits in the message (the empty message has length 0).

If the no. of bits in a text is several of 8, for density we can signify the message in hex.

The determination of message padding is to mark the entire length of a padded message a numerous of 512.

The determination of message padding is to mark the total length of a padded message a multiple of 512.

As a summary, a "1" followed by m "0"s followed by a 64-bit integer are appended to the end of the message to produce a padded message of length 512 * n.

The 64-bit number is l, the length of the unique message.

The padded message is then processed by the SHA-1 as n 512-bit blocks [14].

V. RESULT AND DISCUSSION

Pseudo code

Input: Mail Token

Output: Accuracy of Classifier for Spam /non spam mail

1 Begin

2 {

3 SET no. of Token Mail = k

4 FOR no. of Token Mail = 1 to k

5. {

6. Set correct Classification Count Mail to 0MailToken

Results: for corrected classification using decimal matrix

6.4000 0.2300 0.3700 7.9000 0.0500 60.0000
150.0000 0.9949 2.8600 0.4900 9.3000
5.9000 0.3400 0.2500 2.0000 0.0420 12.0000
110.0000 0.9903 3.0200 0.5400 11.4000
5.0000 0.3300 0.2300 11.8000 0.0300 23.0000
158.0000 0.9932 3.4100 0.6400 11.8000
5.4000 0.2900 0.3800 1.2000 0.0290 31.0000
132.0000 0.9890 3.2800 0.3600 12.4000
8.0000 0.3300 0.3500 10.0000 0.0350 22.0000
108.0000 0.9946 3.1200 0.3600 11.6000

7.Calls Set Training Set and Test Set with

7 For each Test Instance Ti in Test Set

8 {

9. Call Train Classifier with Training Set

Results for training data of mail trial in decimal set

7.0000 0.2700 0.3600 20.7000 0.0450 45.0000
170.0000 1.0010 3.0000 0.4500 8.8000
6.3000 0.3000 0.3400 1.6000 0.0490 14.0000
132.0000 0.9940 3.3000 0.4900 9.5000
7.2000 0.2300 0.3200 8.5000 0.0580 47.0000
186.0000 0.9956 3.1900 0.4000 9.9000
8.1000 0.2800 0.4000 6.9000 0.0500 30.0000
97.0000 0.9951 3.2600 0.4400 10.1000
6.2000 0.3200 0.1600 7.0000 0.0450 30.0000
136.0000 0.9949 3.1800 0.4700 9.6000

10. CALL Classify Test Data with Ti

Results: Test data

6.3000 0.3000 0.3400 1.6000 0.0490 14.0000
132.0000 0.9940 3.3000 0.4900 9.5000
8.1000 0.2800 0.4000 6.9000 0.0500 30.0000
97.0000 0.9951 3.2600 0.4400 10.1000
6.2000 0.3200 0.1600 7.0000 0.0450 30.0000
136.0000 0.9949 3.1800 0.4700 9.6000
8.1000 0.2700 0.4100 1.4500 0.0330 11.0000
63.0000 0.9908 2.9900 0.5600 12.0000
7.9000 0.1800 0.3700 1.2000 0.0400 16.0000
75.0000 0.9920 3.1800 0.6300 10.8000

11. IF classification = Ti. Class THEN

12. INCREMENT correct Classification Count Mail using hash = INITIAL_VALUE;

Results: for correct in find in classification

Spam probability= probs (create token, i);

%probability these words occur in a spam email

Not spam probability=not spam probs (create token, i);

%probability these words occur in a non-spam email

13 ENDIF

14.}

15. CALL Calculate Accuracy with correct Classification Count Mail and Test Set. Count RETURNING Accuracy

16.For i = 1, length (strKey) do {

17.Hash = M * hash + strKey [i]

Results: Confusion matrix for boundaries 0 showing accuracy of spam detection

1 2 1 0 0 0 0
1 8 12 9 2 0 0
2 12 177 76 24 0 0
2 4 114 192 125 1 1
0 0 27 49 97 3 0
0 1 3 6 23 3 0
0 0 0 0 1 0 0

18.}
19. CALL Train Classifier with Training Set
20 CALL Classify Test Data with Ti
21 IF classification = Ti. Class THEN
22 INCREMENT correct Classification Count Mail
Results: probability in spam =0.1600
Probability not in spam = 0.1600
23 ENDIF
24. }
25 CALL Calculate Accuracy with correct Classification Count Mail and Test Set. Count RETURNING Accuracy
You can show our results / figure /graph
26. Return hash % TABLE_SIZE;
27 End

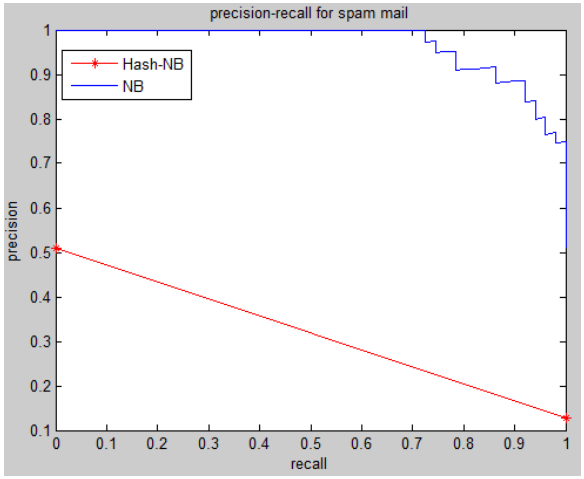

Figure 1: precision and recall using Naïve Bayes and Hash-Naïve Bayes

Precision and recall are the basic measures used in evaluating search strategies.

RECALL is the ratio of the number of relevant records retrieved to the total number of relevant records in the database. It is usually expressed as a percentage.

PRECISION is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved. It is usually expressed as a percentage.

In the graph above, the two lines may represent the performance of different search systems. While the exact slope of the curve may vary between systems, the general inverse relationship between recall and precision remains.

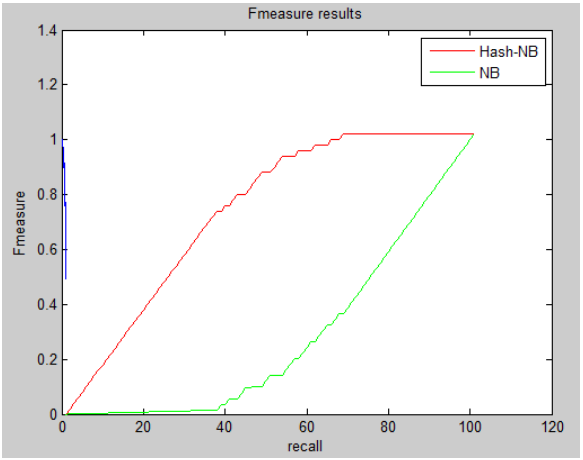

Figure 2: F-measure using EMR-KNN and Proposed algorithm

The F-measure can be viewed as a compromise between recall and precision. It is high only when both recall and precision are high. It is equivalent to recall when $\alpha = 0$ and precision when $\alpha = 1$. The F-measure assumes values in the interval [0, 1]. It is 0 when no relevant data have been retrieved, and is 1 if all retrieved data are relevant and all relevant data have been retrieved.

Limitation

Classification techniques that filter spam at the receiving client are easier to deploy and monitor; however they cannot prevent spam from misusing network bandwidth and storage resources and thus are considered to be least effective techniques. Filtering methods that filter spam on the getting attendant can stay spam even earlier it is deposited locally. Further, they can make better decisions, by aggregating information across multiple spam recipients. Enterprise solutions can detect messages that are delivered to multiple users and that are likely spam, or can even connect to centralized repositories of spam information.

VI. CONCLUSION

In this paper we proposed more sophisticated and robust e-mail abstraction scheme based on Bayesian with new scheme. We can efficiently capture the near duplicate of the spams. Our scheme achieved efficient similarity matching and educed data storage. So we conclude that proposed method is apt for spam discovery.

VII. REFERENCES

[1] Vuong, T.P. and Gan, D. (2012) A Targeted Malicious Email (TME) Attack Tool. 6*th International Conference on* Cybercrime, Forensics, Education and Training (CFET), Christ Church Canterbury.

[2] Nagarjuna, B.V.R.R. and Sujatha, V. (2013) An Innovative Approach for Detecting Targeted Malicious E-Mail. *International* Journal of Application or Innovation in Engineering & Management (IJAIEM), **2**, 422-428.

[3] Symantec Corporation (2014) Internet Security Threat Report 2014, Vol. 19, 1-98. http://www.symantec.com/content/en/us/enterprise/other_resources/b-istr_main_report_v19_21291018.en-us.pdf

[4] Hurcombe, J. (2014) Malicious Links: Spammers Change Malware Delivery Tactics. http://www.symantec.com/connect/blogs/malicious-links-spammers-change-malware-delivery-tactics

[5] E. Blanzieri and A. Bryl, "Evaluation of the Highest ProbabilitySVM Nearest Neighbor Classifier with Variable Relative ErrorCost," Proc. Fourth Conf. Email and Anti-Spam (CEAS), 2007

[6] Kang Li, Zhenyu Zhong and Lakshmish Ramaswamy,privacy aware collaboration spam filtering"

[7] P.-A. Chirita, J. Diederich, and W. Nejdl, "Mailrank: UsingRanking for Spam Detection," Proc. 14th ACM Int'l Conf.Information and Knowledge Management (CIKM), pp. 373-380, 2005.

[8] S. Chhabra, W.S. Yerazunis, and C. Siefkes, "Spam Filtering Usinga Markov Random Field Model with Variable WeightingSchemas," Proc. Fourth IEEE Int'l Conf. Data Mining (ICDM),pp. 347-350, 2004.

[9] A.C. Cosoi, "A False Positive Safe Neural Network; The Followersof the Anatrim Waves," Proc. MIT Spam Conf., 2008.

[10] Syed TahaOwais, Md Tabrez Nafis, Seema Khanna, "An Improved Method for Detection of Satire from User-Generated Content" in (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (3) , 2015, 2084-2088,ISSN:0975-9646.

[11] M Tabrez Nafis, Alok Pathak, "To Find Influential's in Twitter based on Information Propagation" in International Journal of Computer Applications (ISSN:0975 – 8887) Volume 118 – No. 13,2015.

[12] Mangal Singh, M Tabrez Nafis, Neelmani, "Sentiment Analysis and Similarity Evaluation for Heterogeneous-Domain Product Reviews" in International Journal of Computer Applications (ISSN:0975 – 8887) Volume 144 – No.2,2016.

[13] Doktoringenieur "Data Mining with Graphical Models"Magdeburg, den 04. Juli 2000.

[14] Descriptions of SHA-256, SHA-384, and SHA-512 (http://www.iwar.org.uk/comsec/resources/cipher/sha256-384-512.pdf)

[15] Konstantin Tretyakov "Machine Learning Techniques in Spam Filtering" May 2004, pp. 60-79.

