IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Special Issue-10, May 2019 E-ISSN: 2347-2693

Spam Detection Approach Using Modified Pre-processing With NLP

Neelam Choudhary[1*], Nitesh Dubey[2]

[1,2] Department of Computer Science Engineering, GNCSGI, Jabalpur, MP, India

[*]*Corresponding Author: neelamch21@gmail.com, Tel.: +91-62602-27716*

DOI: https://doi.org/10.26438/ijcse/v7si10.158161 | Available online at: www.ijcseonline.org

Abstract— However, the growth in emails has also led to an unprecedented increase in the number of illegitimate mail, or spam 49.7% of emails sent is spam - because current spam detection methods lack an accurate spam classifier. We are excited by the decline in the volume of email spam but it also raises the question as to whether the email spam business is dying and will continue to decline. Besides the volume change, we also consider the quality of email spam and the impact, which may constitute a new trend of email spam business. For instance, spammers may post email spam in a more complicated way using spoofed email addresses and changing email relay servers. That kind of email spam may slip away under the inspection of spam filters. Thus, it motivated us to investigate the evolution of email spam using advanced techniques such as topic modelling and network analysis. We try to find out the real trend of email spam business through email content, meta information such as headers, and sender-to-receiver network over a long period of time.

Keywords—Spam detection, email, NLP, spam classification

I. INTRODUCTION

Spam is problematic not only because it often is the carrier of malware, but also because spam emails hoard network bandwidth, storage space, and computational power [1], [2], [3]. Additionally, the commercial world has significant interests in spam detection because spam causes loss of work productivity and financial loss [1], [2]. It is estimated that American firms and consumers lose 20 billion annually, even while sustained by the private firms' investment in anti-spam software. On the other hand, spam advertising earns 200 million per year [4]. Although extensive work has been done on spam filter improvement over the years, many of the spam filters today have limited success because of the dynamic nature of spam [1]. Spammers are constantly developing new techniques to bypass filters, some of which include word obfuscation and statistical poisoning. Although these two text classification issues are recognized, research today has largely neglected to provide a successful method to improve spam detection by counteracting word obstruction and Bayesian poisoning, and many common spam filters are unable to detect them.

Many existing methods of spam detection are ineffective, exemplified by the increase in spam mail. The two categories of email-filtering techniques are knowledge engineering and machine learning [2]. In knowledge engineering, a set of rules to determine legitimacy is established or rule-based spam filtration. However, rule-based spam detection has slight disadvantages because this method is exclusively based on spammers' previous methods, while spammers' methods are continuously expanding [2]. On the other hand, machine learning methods are customized based on the user and is able to adapt to the changing spamming methods, yet is slower. Another major issue with knowledge engineering spam detection is that, although some of the rules are often characteristics of spam emails, they do not necessarily imply that the message is spam. Since emails are text in the form of strings, they must be converted into objects such as vectors of numbers, or feature vectors, so that there is some measure of similarity between the objects [2] in this conversion process, there may be loss of information. Feature selection is a prominent yet neglected issue in modern spam filtering because spam and ham emails with the same feature vector will be incorrectly classified, resulting in a high false positive rate, or ham emails being misclassified as spam [2]. Thus, most effective spam detection methods utilize some form of machine learning [3]. Non-machine learning methods of spam detectors include using the IP numbers of senders, calculating the correlation of text to a preset list of words used to find spam, and many etc. The differentiating characteristic between machine learning techniques and non-machine learning methods is that after being trained using a dataset; the machine is able to make more accurate predictions on its own instead of constantly requiring human programming. Spam detection methods employing machine learning include the Naive Bayes, Vector Space Models, clustering, neural networks, and rule based classification.

Spam will be terribly expensive to e-mail recipients; it reduces their productivity by wasting their time and inflicting annoyance to alter an outsized quantity of spam. Consistent with Ferris analysis, if an worker got 5 e-mails per day and consumes thirty seconds on every, then he/she can waste fifteen hours a year on them [5]. Multiplying this to the hourly rate of every use in an exceedingly company offer the value of spam to the current company. Additionally, spam consumes the network information measure and space for storing and might cut down email servers. Spam software system also can be utilized to distribute harmful content like viruses, Trojan horses, worms and different malicious codes. It will be a way for phishing attacks also [6]. As a result, spam has become a locality of growing concern attracting the concentration of the many security researchers and practitioners. Additionally to rules and legislations, varied anti-spam technical solutions are projected and deployed to combat this downside. Front-end filtering was the foremost common and easier path to reject or quarantine spam messages as early as potential at the receiving server [7].
The job of email spam filtering is nothing however manually removing harmful, unwanted, and offensive email messages before they're delivered to a user -is a vital, massive scale application space for machine-learning strategies. These techniques have associate degree perfect assumption that a learning-based filter are given absolutely correct label feedback for each message that the filter encounters. In observe, label feedback is provided periodically by users, and is much from absolutely correct. Few previous spam filters will meet the necessities of being easy, attack-resilient, and customized.

Many approaches are not getting in consideration the closeness relationships and (dis)interests of people. Previous spam filtering approaches are often principally divided in 2 categories: content-based and identity-based. However, each class strategies having limitations and thus suffered from variety issues that invalidate such strategies underneath real time settings. Some strategies has been correct, however not user friendly. Some strategies have been user friendly however not customized, and susceptible to numerous alternative attacks. To deal with these limitations recently improved technique conferred named as SOAP (Social network assisted customized and effective spam) [8], that is showing much economical results as compared to previous strategies. The limitation of SOAP is that it uses basic Bayesian spam filters. In this spam filtering method is complicated and suffered from limitations. This becomes new analysis drawback during this domain. The main objectives of this research are:

a. To develop a high speed NLP-based anti-spam engine to filter out spam. This engine will improve email system by reduce the risk of email user expose to spam email.

b. To detect spam email that used the minor string change strategy to evade Naive based spam filter. This spam filtering engine is able to detect spams which are hard to detect by Naive based spam filtering system when spammer obfuscate the spam keyword.

II. RELATED WORK

Comparison of related work is shown below:

Table 1. Related Work

Paper	Method	Limitations
[9]	This paper proposes the linear approach of K-means and Naive Bayes algorithms. It computes the final result using naive bayes algorithm and checks the text is spam or not. Accuracy is calculated based on training dataset.	Using the same data for training and testing. It focuses on time rather than accuracy. It does not deal with unstructured data. Pre-processing of data set is not done.
[10]	They proposed an SVM-NB system to achieve effective and efficient spam email filtering. SVM-NB aims at removing the assumption of independence among features extracted from training set, when the NB algorithm is applied. The solution leverages SVM technique to divide training samples into different categories and identifies dependent training samples. Removing those samples results in a more independent training set with few overlapping features.	Currently, the SVM-NB algorithm is only applicable to text based spams detection. It is weak for unstructured data. Number of Iterations is high. Pre-processing not have a great impact to training and classification results.
[11]	This paper proposes a novel classification method based on feature space segmentation. Naive Bayes (NB) model is a statistical filtering process which uses previously gathered knowledge. Instead of using a single classifier, we propose the use of local and global classifier, based on Bayesian hierarchal framework. We have presented a hybrid Bayesian Classifier with local classifier for each user, and a global classifier which governs the parameters of tokens in the local classifier. This leads to a type of information sharing among the users while maintaining the individuality.	Naive Bayes is a simple and efficient technique of spam filtration. Creating a Naive Bayes network helps in exploiting the commonness among different tasks, thus learning and modifying accordingly. Training a spam filter through tokenization is slow process as large scale data is generated in the training process. So selection of right data structure is essential for increasing the performance of the model

III. METHODOLOGY

Most of the current email servers have the capability to block the spam mails by making use of some filters. However, some spam emails still sneak through the spam filters. This is mainly because the spammers manipulate the spam filters by appending certain words to the mail which rarely occur in spam mails, or substituting certain characters of the spam

words, or by adding the synonyms of the words which occur in the spams, to fool the spam filters.

Our main idea is to design a baseline system that will train the machine and classify the emails as ham or spam based on naïve bayes training. The emails identified as spam would be kept aside as correct spams, whereas, the ham emails would be sent to the NLP engine that we will design to classify them further. We propose to check some lexical and semantic features to help the bayes engine classify them correctly.

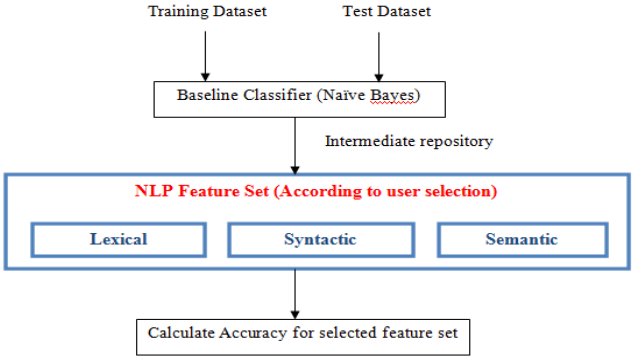

Figure 1: Proposed System

We attempted to improve the naïve technique by applying lexical and semantic features by looking at the content of the text like emails. Figure below will represent complete process for training and testing of classifier.

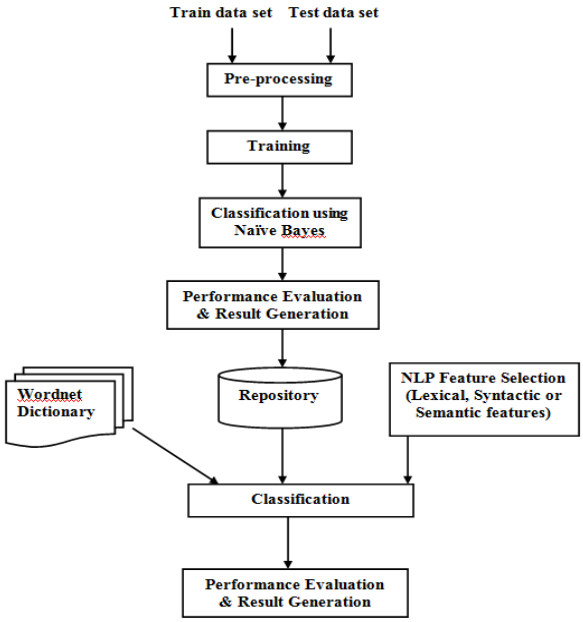

Figure 2: Training & Testing Process

At the very beginning, we tokenize the text into words and initialize their ham and spam word count based on their placement, whether in ham or spam. In the next step, we train our system by using some lexical features.

Stop words removal

Stop words are words which do not add into much of a meaning to the topic and yet appear most frequently in the documents like articles or prepositions. We maintain a stop words list. During training phase, we skip the word if it is a stop word, and do not consider it in the Bayes probability calculation.

Swear words handling

Mostly, the spam emails may contain swear words, which make them categorized as spam emails. But this may not always be the case. So, while training, we make a note of the swear words occurring in the ham emails, and increase their ham count. This will increase the weight of that swear word being in ham.

Common spam phrases handling

We maintain a list of common spam phrases. We, scan the emails line by line, and check whether it contains the commonly occurring spam phrases. The occurrence of the spam phrase in the mail, increases the probability of the email being a spam. So, we accordingly increase the spam probability in the Bayes calculation.

Further, we tried to use the syntactic constructs of the emails.

N-Gram POS tagger

We tagged all the emails' content with their part of speech. Then, we considered a window of 5 tags and scanned the emails by moving the window one tag at a time. These combination of tags and their frequency of occurrence was recorded. And, once all the emails were scanned for this tag window, the n-gram probability (with n = 5), was used to test the emails.

We observed that spammers use synonyms or hypernyms of spam words in text like emails.

IV. RESULTS AND DISCUSSION

We have implemented the system using java with some predefined tools like Stanford core nlp, Wordnet Dictionary. Following is the GUI for proposed Spam filter. Window below compute will show the results when features on left side are selected and compute button is hit. Firstly we have to give path of train and test dataset and then go for applying methods.

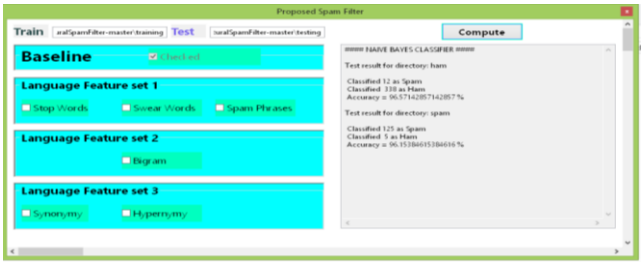

Figure 3: Main GUI

Baseline method (Naive Base) and proposed methods are tested on two sets of directories: ham and spam. Firstly they are trained and then tested. For evaluation, parameter used is accuracy of the method. For spam, accuracy increases with using combinations of features and in ham accuracy decreases. Chart below represents accuracy comparison of results.

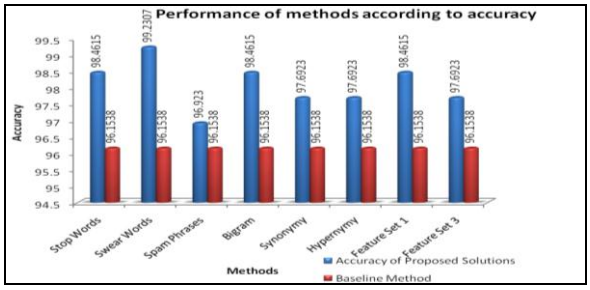

Figure 4: Chart of Resultant Accuracy for SPAM Directory

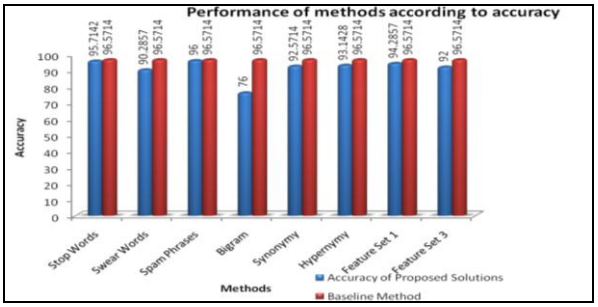

Figure 5: Chart of Resultant Accuracy for HAM Directory

V. CONCLUSION AND FUTURE SCOPE

In this thesis work we are explain about the e-mail spam classification to identify the spam and ham mails. For this purpose we are using Naïve Bayesian Classifier. In this thesis we are creating an email spam classification system for classify the spam and non-spam mails. In this dissertation, a novel spam filtering engine which implements Naive Bayes Algorithm and NLP Algorithm in combination with various features has been proposed. As spammers had improve their spamming technique by obfuscate the spam email keywords to evade spam filter, normal rule-based spam filtering system will be very hard to detect this kind of spam. But this problem can be solve by this proposed engine due to it have the ability to detect obfuscated spam. This proposed engine is implemented in Netbeans and using Micro Blaze soft-core processor, thus it comes with the advantages of better speed performance. This spam filtering engine is able to provide better filtering speed compare with software-based spam filtering system.

However, users can always adjust the threshold to suit their needs and email environment. As a conclusion, this spam filtering engine is a recommended solution to improve email system for every email user by minimizes spam risks.

Graphical User Interface (GUI) tool for user control Currently this engine only display the filtering result in Hyper Terminal and there is no has any GUI tools for user to control this system, so this will give some trouble and inconvenient to users. In order to allow users to handle the filtering task more conveniently, it is suggested to create a GUI tool. The GUI tool should have a few modules such as:

- Filtering Module: Allow user to start, pause or stop spam filtering process.
- Reporting Module: Allow user to print the report of filtering result and alert user when spam detected. It also should have log file of the filtering process.
- Pattern Database Update Module: Provide an easy way for user to update the pattern database regularly.

REFERENCES

[1] A. Bhowmick and S. Hazarika, "*Machine learning for e-mail spam filtering: review, techniques and trends*," https://arxiv.org/abs/1606.0104, 2016, accessed: 2017.

[2] A. Aski and N. Sourti, "*Proposed efficient algorithm to filter spam using machine*," in Pacific Science Review A: Natural Science and Engineering, vol. 18, 2016, pp. 145–149.

[3] J. Rao and D. Reilly, "*The economics of spam*," in Journal of Economic Perspectives, vol. 26, no. 3, 2012.

[4] H. Tschabitscher, "*How many emails are sent every day?*" https://www.lifewire.com/how-many-emails-are-sent-every-day-117121, 2017, accessed: 2017.

[5] J.S. Kong, P.O. Boykin, B.A. Rezaei, N. Sarshar, and V.P. Roy chowdhury, "*Let Your Cyber Alter Ego Share Information and Manage Spam*," Univ. of California, Los Angeles, CA, technical report,2005.

[6] F. Zhou, L. Zhuang, B.Y. Zhao, L. Huang, A.D. Joseph, and J.D. Kubiatowicz, "*Approximate Object Location, and Spam Filtering on Peer-to-Peer Systems*," Proc. Middleware, pp. 1–20, 2003.

[7] SPAMNET, http://www.cloudmark.com, accessed in Mar. 2014.

[8] Haiying Shen, Senior Member, IEEE, and Ze Li, Student Member, IEEE, "*Leveraging Social Networks for Effective Spam Filtering*", IEEE TRANSACTIONS ON COMPUTERS, VOL. 63, NO. 11, NOVEMBER 2014.

[9] Dr Devendra K. Tayal, Amita Jain, Kanak Meena," *Development of Anti-spam techniques using modified K-means & Naive Bayes Algorithms*" IEEE-2016.

[10]Weimiao Feng, Jianguo Sun, Qing Yang, "*A Support Vector Machine based Naive Bayes Algorithm for Spam Filtering*", IEEE-2016.

[11] Rohit Kumar Solanki, Karun Verma, Ravinder Kumar*," Spam Filtering Using Hybrid Local-Global Naive Bayes Classifier*" IEEE-2015.

.

