Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 08, August-2022

A Content based Mail Detection Technique

Irani Hazarika
Dept of Computer Science
Gauhati University
Guwahati, India

Purabi Choudhury
Dept of Computer Science
Gauhati University
Guwahati, India

***Abstract*— Email spam is one of the major problems of the today's internet. There is various ways to detect the spam emails. In this paper a content based mail detection techniques has been proposed. For this purpose, the content of each mail is considered as a text document. To measure the performance, the proposed method is applied on a real life dataset. The performance of the proposed method is compared with standard classification algorithm K-Nearest Neighbor (KNN). It is seen that proposed method gives better accuracy than the KNN algorithm.**

Keywords—Classification, Spam mail, Vector space model, Content based mail detection

I. INTRODUCTION

The term spam is generally used to denote an unsolicited commercial e-mail. Email spam is one of the major problems of the today's internet. There are various methods that have been proposed to automatic classify messages as spam or legitimate, such as rule-based approaches, white and blacklists, collaborative spam filtering, challenge-response systems etc. Some popular spam mail detection methods include approaches like rule based algorithm [1], Support Vector Machines [2], Naive Bayes classifiers [3], Bayesian classifier [7] etc to automatically detect and remove these spam's. A Bayesian classifier is statistical classifier works on independence computation of probability [7]. In paper [4] a non-content based mail filtering approach is proposed. Here, an intelligent hybrid spam filtering method is given, which only analyzes email headers for spam detection. The content based mail detection technique [5] [6] is an important and popular one. Content based method checks for text in the body of email for classification. It performs text classification task by employing some preprocessing on the text. In this paper, we have proposed a content based mail classification technique by using vector space model. To classify an input mail as ham or spam here we consider content of each mail as a text document.

The remainder of this paper is organized as follows. In section II, some pre-processing steps are given. In section III, the proposed method is described. Experimental results are showed in Section IV. Finally, Section V offers conclusions and outlines for future works.

II. STEPS OF PREPROCESSING

As we mentioned earlier the aim of the proposed method is to classify a test mail as spam or ham by analyzing its content and content of sets of available training mails. In the proposed method we considered two types of training mails sets. One training mails set contains only ham mails and other contains only spam mails.

Thus, we have two training datasets that contains two sets of training mails and a test dataset that contain an input test mail. Before applying the proposed method, first we have to apply some pre-processing tasks on the datasets. The pre-processing steps are described below:

- *Converting HTML files to TEXT files*

The mails in the datasets are present in the form HTML. Thus these mails are first converted to text files using the software named "Okdo HTML to TXT converter".

- *Stop word removal*

After converting the mails to text file, the non textual content and stop words are removed from the text files by using a standard stop wordlist. After removing the stop words, each text file in the datasets contain only a set of keywords.

- *Creating master wordlist*

After removing the stop words from the files, we have to create master wordlist for each of the training mail set and test mail. The master word list contains every distinct keyword present in the dataset.

III. PROPOSED METHOD

In this proposed method the mails present in the training datasets and test dataset are represented as vector space models by using normalized term frequency of the mails. After that arithmetic mail vectors are computed from the vector space models. Using these arithmetic mail vectors the test mail is classified as spam or ham. The procedures are described below-

A. Vector space model representation of mails

After applying the pre-processing steps on the mails present in the datasets, each mail is represented as a fuzzy set by using normalized term frequency of the keywords present in its master wordlist.

Let W be the set of all distinct keywords appearing in the master wordlist of the dataset. Let $|W| = n$. The keywords are arranged in a random order and thus get a sequence of the form

$$W = \{w_1, w_2, w_3,...., w_n\}$$

Now keeping this ordering in mind, we can represent each mail *d as*

$$d=\{o_1, o_2, o_3,..... o_n\}$$

where o_i indicates the term frequency of the word w_i in the mail d i.e. the number of occurrences of word w_i in d. If a word w_i is not present in the mail, then o_i =0 for that mail.

We can normalized the term frequency value of the word w_i in mail d to [0, 1] as follows-

$$ntf_i = o_i/n$$

Thus, the fuzzy set representation of mail *d* is as follows-

$$d= \{ ntf_1, ntf_2, ntf_3, \ldots , ntf_n \}$$

Again, the vector space model is an algebraic model for representing text documents as vectors of identifiers. Here, the vector space model for the mails present in a dataset has created as follows-

The vector space model V_{mxn} is represented based on the normalized term frequency vector for *m* numbers of mails present in the dataset and *n* numbers of distinct words in its master wordlist, where entry V_{ij} denotes normalized term frequency (ntf_{ij}) of j^{th} keyword in i^{th} mail.

In this proposed method, we will create three vector space models, one for training ham mail dataset, one for training spam mail dataset and one for the test data.

B. Calculation of arithmetic mean vector

From each vector space model a new vector is created which is called as arithmetic mean vector.

Suppose, A = $\{atf_1, atf_2, \ldots\ldots, atf_n\}$ is the arithmetic mean vector for the vector space model V_{mxn}. Then is atf_i calculated as follows-

$$atf_j = \sum_{i=1}^{m} ntf_{ij} / m$$

C. Fuzzy similarity measure

A fuzzy similarity function for pair of mails is defined which can be calculated from the fuzzy logic [8]. Let ntf_{m1} and ntf_{m2} be two normalized term frequency vector for mail *m1* and *m2*, then fuzzy similarity between *m1* and *m2* can be calculated using-

$$sim\ (m1, m2) =. | ntf_{m1} \cap ntf_{m2}| / | ntf_{m1} \cup nf_{m2}|$$

D. Proposed algorithm

The proposed spam mail detection algorithm is as follows-

Step 1:

A) For training dataset containing *n* numbers of ham mails

i. Create a master ham word list L_H which contains all distinct words (W_{ham}) from ham mails.
ii. Create a vector space model $V_{n\ x\ wham}$ for *n* numbers of ham mails.
iii. Calculate the arithmetic mean vector H_{train} for the vector $V_{n\ x\ wham}$

B) For dataset containing *m* numbers of ham mails

i. Create a master spam word list L_S which contains all distinct words (W_{spam}) from ham mails.
ii. Create a vector space model $V_{m\ x\ wspam}$ for *m* numbers of input spam mails.
iii. Calculate the arithmetic mean vector S_{train} for the vector $V_{m\ x\ wspam}$.

Step 2:

For the test mail

i. Create a word list L_N with all distinct words from the test mail.
ii. Find out wordlists X_H and X_S with E_H, and E_S numbers of extra distinct word respectively from L_N. The words in X_H and X_S are in L_N, but not in L_H and L_S respectively.
iii. Append E_H and E_S number of zeroes to the arithmetic mean vectors *H* and *S* respectively.
iv. Calculate the vector space models $V_{1\ x\ wham1}$ and $V_{1\ x\ wspam1}$ for the test mail. To create $V_{1\ x\ wham1}$, the normalized term frequency vector for the test mail is calculated by appending the wordlist X_H to L_H *(wham1:* total number of words in X_H and L_H*)* and to crate $V_{1\ x\ wspam1}$, the normalized term frequency vector for the test mail is calculated by appending the wordlist X_S to L_S *(wspam1:* total number of words in X_S and L_S*)*
v. Calculate the arithmetic mean vector H_{test} for $V_{1\ x\ wham1}$ and S_{test} for $V_{1\ x\ wspam1}$

Step 3:

i. Calculate fuzzy similarity $fsim_{ham}$ between the arithmetic mean vectors H_{train} and $V_{1\ x\ wham1}$.
ii. Calculate fuzzy similarity $fsim_{spam}$ between the arithmetic mean vectors S_{train} and $V_{1\ x\ wspam1}$

Step 4: Assign the mail to the group ham mails if value of $fsim_{ham}$ more than $fsim_{spa}$, otherwise assign it to the group spam mails.

IV. EXPERIMENTAL RESULTS AND DISCUSSIONS

For the experimental purpose the proposed method has been applied on the real life mail dataset Eron. The Eron email dataset was collected and prepared by the CALO project. The mails in the data set are classified into two categories ham and spam. The ham category contains a large set of ham mails and spam category contains a large set of spam mails.

The performance of the proposed method is compared with standard KNN algorithm by using classification accuracy. Both the methods are implemented in C^{++}.

For the convenience the KNN algorithm is described below-

Consider a training dataset that contains both ham and spam mails. Find the distinct word lists for both training and test mail sets separately. Now, add the words which are present in test mail list but not present in the training mails list into the distinct word list of the training mails set. Suppose this new distinct word list is L_{new}.

Based on this list L_{new} find the vector space models for both the training mails set and test mail using normalized term frequency of the mails. Now, both these vector space models are inputted to the KNN algorithms for classify the test mail.

The steps of KNN algorithm are-

i. Calculate the fuzzy similarity value c_i between the test mail and each training mails d_i (where i=1, 2, 3..., n) using respective vector space model.
ii. Each pair [(c_i, d_i) is stored in a structure A as follows-

$$A= [(c_1, d_1), (c_2 , d_2), ... , (c_n , d_n)]$$

iii. Reverse the similarity values present in A as-

$$c_i = 1 - c_i$$

iv. Arrange the pairs (c_i, d_i) present in A in ascending order according of the value c_i.
v. Set $D_{max} = c_n$
vi. Calculate the probability of mail d_i as-

$$P_i=1-c_i/D_{max} \text{, for } i=1,2,3,......,n$$

vii. Select first K mails from A.
viii. Set B_j=0, where value of j is 1 up to number of classes.
ix. $B_j= B_j + P_i$, where P_i is the probability of mail d_i from class j and $d_i \epsilon$ first K mails from A
x. Select j with maximum value of B_j
xi. Assign the test mail to class j

- *Accuracy Measure*

The accuracy measures the percentage of prediction that is correct. The accuracy (*Acc)* of the proposed method has been calculated using the following measure-

.

$$Acc = (TP+TN) / (TP+FP+FN+TN)$$

Where-

TP= Number of true positive
FP= Number of false positive
FN= Number of false negative
TN= Number of true negative

True Positive (*TP*): This term states the number of spam mails correctly classified as spam.

False Negative (*FN*): This term states the number of ham mails that is classified as spam.

False Positive (*FP*): This term states the number of ham mails that is classified as ham.

True Negative (*TN*): This term states the number of spam mails that is classified as ham.

- *Results and discussion*

To measure the performance of the proposed method we consider two sets of training datasets (one for spam mails and one for ham mails) each containing 300 emails from ERON mail dataset. Thus, in KNN algorithm the training mails set contains 600 mails as it contains both ham and spam mails. After that the accuracy of the proposed method and KNN algorithm has been measured for different numbers of test emails. The results are shown in Table 1. From Table 1, it is seen that the proposed method gives better results than KNN algorithm in all cases.

Table1: Accuracy of the KNN algorithm and proposed method on different numbers of test mails

No of test mails	Accuracy of KNN	Accuracy of Proposed Method
30	0.4	0.4
50	0.78	0.78
80	0.6	0.6
100	0.68	0.78

V. CONCLUSION AND FUTURE WORKS

This paper presents a content based mail classification method. The evaluation demonstrates that the proposed method gives better results than the KNN algorithm in classification of spam mails. In future this method will be extended, so that it can detect attachment based mails and also ccomparison of the result of the algorithm on some real data set will be done.

REFERENCES

[1] G.Santhi, S.Maria Wenisch, Dr. P. Sengutuvan, "Fuzzy Rule based Novel Approach to Spam Filtering", International Journal of Computer Applications, Vol. 71(14), pp. 0975 – 8887, May 2013

[2] H. Drucker, V. Vapnik, D. Wu, "Support vector machines for spam categorization", IEEE Transactions on Neural Networks, Vol. 10(5) pp. 1048–1054, 1999.

[3] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G.Sakkis, C.D. Spyropoulos, P. Stamatopoulos, "Learning to filter spam e-mail: A comparison of a naive Bayesian and a memory based approach", Proceedings of the Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases , (PKDD 2000), pp. 1–13, 2000.

[4] Yong Hu, Ce Guo, E.W.T. Ngai, Mei Liu, Shifeng Chen "A scalable intelligent non-content-based spam-filtering framework" Expert Systems with Applications, Vol. 37, pp. 8557-8565, 2010

[5] M. Basavaraju, Dr. R. Prabhakar, "A Novel Method of Spam Mail Detection using Text Based Clustering Approach", International Journal of Computer Applications, Vol. 5(4), pp. 0975 – 8887, August 2010

[6] Dr. Sonia, "Spam Filter: VSM based Intelligent Fuzzy Decision Maker", IJCST, vol. 1(1), September 2010.

[7] Sunil B. Rathod, Tareek M. Pattewar, "Content based spam detection in email using Bayesian classifier", International Conference on Communications and Signal Processing (ICCSP), pp. 1257-1261, 2015.