Spam Filtering at the Client E-mail Level

Dr. Imad AL-Hussaini*, Dr. Mumtaz AL-Mukhtar & Mohammed M. Mazin*****

Received on:8/5/2006
Accepted on:3/1/2008

Abstract

Spam has now become a significant security issue and a massive drain on financial resources. In this paper, a method for filtering the spam at the client level is presented. The proposed filter combines more than one filtering mechanism that would make the filter more efficient, faster and low false positive. The main mechanism implemented is the Bayesian filter combined with a blacklist and whitelist. The header of the incoming e-mail will be tested against the whitelist to determine whether the e-mail is legitimate or not. Also it will be tested against the blacklist to determine whether the e-mail is spam or not. In case of no matched results the e-mail will be checked by the Bayesian filter. The results of this check will be then used to update the whitelist and blacklist.

Keywords: Spam, Spam filter, Bayesian filter, Whitelist, Blacklist.

مرشح رسائل الدعاية على مستوى مستخدم البريد الاكتروني

الخلاصة

اصبح بريد الاعلانات غير المرغوبة و بريد الدعاية (spam) احدى المشاكل الاساسية التي يتعرض لها البريد الاكتروني مما يسبب هدر الموارد المالية المخصصة لهذا الغرض. يقدم هذا البحث مرشحا لرسائل الدعاية مخصصا للمستخدم (client). المرشح المقترح يتضمن عددا من المراحل التي تعمل معا لتجعل المرشح اكثر فعالية، اسرع و اقل عرضة للخطأ. مرشح (Bayesian) هو المرشح الرئيسي الذي تم استخدامه كمصنف للبريد اعتمادا على محتوى البريد. مرشحا القائمة البيضاء (whitelist) و القائمة السوداء (blacklist) يعتمدان الترشيح على اساس المصدر المرسل مما يسمح بتصنيف البريد المستلم على انه شرعي (legitimate) او غير ذلك (spam). النتائج المستحصلة تستخدم لتحديث هذه القوائم.

1 Introduction

The spam is a commercial mail but there is no standardized definition for spam [1]. Spam mail, also called unsolicited bulk e-mail or junk mail, is Internet mail that is sent to a group of recipients who have not requested it [2]. Spam constitutes a major problem for both e-mail users and Internet Service Providers (ISP) [1]. These unsolicited mails have already caused many problems such as filling mailboxes, engulfing important personal mail, wasting network bandwidth, consuming users' time and energy to sort through it, and crashing mail-servers.

The task of spam filtering is to rule out unsolicited e-mails automatically from a user's mail stream [2]. The average number of spam messages received is continually increasing exponentially. Figure (1) shows recent statistics on the number of spam messages received by one e-mail user. It explains how the spam is becoming a real problem in the last years [1, 3, 4].

There are several effective statistical filtering spams available. Some common approaches are artificial neural

* Iraqi Commission for Computers and Informatics
**Information Engineering Faculty, AL-Nahrain University
*** Informatics Institute of Postgraduate Studies

https://doi.org/10.30684/etj.27.8.13
2412-0758/University of Technology-Iraq, Baghdad, Iraq

networks and Intrusion Detection Systems (IDS) [5]. The Bayesian filters have now become the standard for spam filtering [6]. Bulk mailers use several different techniques to send their spam. Often Bulk mailers misuse the Simple Mail Transfer Protocol (SMTP) or use badly configured Mail Transfer Agent (MTA) (so-called open-relays) to hide their tracks. Filtering is a highly popular technique, with many ways to deal with spam [7].

There are at least three fundamentally different ways to counter spammer. First, bulk mailers can be prevented to send spam by blocking or limiting access to mail servers. A second method is to make spamming less profitable, for example by incurring a cost on every e-mail message sent. A third method aims to detect and remove all spam once it is sent by applying different types of filtering techniques that use the special characteristics of spam to recognize it [8].

In order to address the growing problem, each organization must analyze the tools available to determine how best to counter spam in its environment. Tools, such as the corporate e-mail system, e-mail filtering gateways, contracted anti-spam services, and end-user training, provide an important arsenal for any organization [9, 10].

This work focuses on the filtering spam using statistical approach that depends on the e-mail contents and header analysis to differentiate between the spam and legitimate e-mail.

2 Related works

2-1 SpamAssassin

SpamAssassin is an example of a rule-based scoring system. To identify spam, SpamAssassin uses a wide range of heuristic tests on e-mail headers and body text. Because spammers and their spam-making applications are not static, rule-based scoring systems are facing some of the same challenges that word filters face. Rules must be updated regularly in order for rule-based scoring systems to remain effective. For example, if a rule-based scoring system has a rule that assigns points to the word “Viagra”. Spammers can easily circumvent this rule by purposely misspelling “Viagra” as many different ways as required to successfully deliver the spam. Rule-based scoring systems, however, if used properly, can be very effective, eliminating over 90 percent of incoming spam [11].

2-2 Bogofilter

This filter is characterized by doing smarter lexical analysis. In particular, hostnames and IP addresses are retained as recognition features rather than broken up. Various kinds of MTA craft such as dates and message-IDs are discarded so as not to bloat the word lists.

Speed was an important consideration because thousands of e-mails would be processed through two different filters. If the computer used to run the filters by more than one user, this test could lock up the machine for hours. Bogofilter appeared to be the least obtrusive Bayesian filter available. It takes an e-mail (the standard input to the program) and it returns 0 or 1 depending on whether or not it thought it was spam. With different command line switches, it can be told to register a word as spam or ham (legitimate). It also has the ability to undo a previous addition to the database if it was erroneous [12].

3 Spam Filter Techniques

Filtering is a highly popular technique. It involves selecting and removing spam from the legitimate e-mail. Some of the filtering techniques can be discussed briefly as follows:

3-1 Rule-Based Filter

Perhaps the most straight-forward method of filtering spam is a rule-based algorithm. Rules are defined to classify e-mails as spam or legitimate

based on different characteristics. An example rule could be that all e-mails with magenta-colored text are spam. Another example would be that all e-mails that contain the text "order confirmation" are legitimate. A good rule-based filter would note which rules match, and make a decision based on all of the rules combined [12].

3-2 Whitelist & Blacklist

A whitelist is a list containing a collection of contacts which we will accept e-mail messages from. If an e-mail arrives but does not come from one of the contacts in the whitelist then it is rejected (placed in spam folder). While this technique is effective for some users it is clear to have faults. Any e-mail sent by a stranger will simply be incorrectly classified as a spam in other words it's a false positive. In all but a few scenarios it's inconceivable to know a priori all contacts that will send us an e-mail [7]. Because of this, the present approach does not reject the mail, but it sends it to a blacklist to be checked.

A blacklist is a list of traits that spam e-mails have, and if the e-mail tested contains any of those traits, it is marked as spam. It is possible to organize a blacklist based on "From:" fields, originating IP addresses, the subject or body of the message, or any other part of the message that makes sense. Blacklists can be used on both large and small scales. A large-scale blacklist would usually be provided by a third party. The user typically does not contribute to a large list like this. On a smaller scale, the user could simply tell his e-mail client not to allow e-mails from certain addresses. A small-scale blacklist works fine if the user gets spam from one particular address. On a larger scale, where the user does not have any control over the blacklist, there must be a mechanism in place for dealing with accidental blacklisting of other users. [12].

3-3 Bayesian Filter

Bayesian filtering is a statistical approach that involves teaching a system that a particular input gives a particular result [1]. For Spam filtering, this teaching is repeated, many times over, with many spam and legitimate mails. Once this is finished, a Bayesian system can be presented with a new e-mail and will give a probability of the result being spam. For best results, teaching should be a constant process. The Bayesian engine provides a single probability figure for each e-mail processed. This probability ranges from (0% likelihood that an e-mail is a spam) up to (99% likelihood) [11]. The Bayesian filter's first big advantage is already evident. There is no human intervention required to generate the feature recognizers. A simple white space delimiter detector can break the incoming text into words, and each word is considered a feature in the database [6].

3-4 Stopping Spam from the E-mail Server

Spammers have to get their Internet connection through some Internet service provider (ISP) and cutting spam off on the sending side would be the job of the ISP. Whether the spammer uses the ISP's e-mail server or their own, it should not be too hard to detect when a user sends out thousands of e-mails. Thereafter terminating these accesses would probably be sufficient to block spammers.

The problem does not lie in detecting the spam. However, the problem is that some ISPs are willing to let spammers use their service to send out thousands of e-mails [12]. Convincing all ISPs to aggressively monitor for and terminate spammers is not an easy task, and is not in the scope of this work.

4 Filtering Spam Implementation at the Client E-mail Level

Most Mail User Agents (MUAs) have some sort of features for

categorizing e-mails based on a set of rules determined by the user. These rules can be constructed to examine an e-mail message body for keywords or phrases given by the end-user. A common use of such rules is to categorize a newly arrived e-mail into a specific folder. For example, some users have a folder for work e-mails. A rule could be setup to transfer a new arrived e-mail that contains the word "job" into the work folder [7].

4-1 A Proposed Filtering Spam Strategy

When the client e-mail (MUA) establishes a connection with POP3* Server, the MUA would request from the server the list of messages to be downloaded from the server. The header of the arrived message will be analyzed to extract information about the source of this message. Then the extracted information will be checked against a whitelist that contains a list of acceptable addresses. This list is stored in a table in the client machine. If there is a matching address, the message will be passed to the Inbox Folder; else the extracted information will be examined against a blacklist to ensure that the message is not coming from a spammer. If there is no matching, the unlabeled message will be examined in the statistical Bayesian filter to determine whether the message is a spam or not. The result of the filter will be labeling the unlabeled message as a spam or not spam. The tables of blacklist and whitelist will be updated accordingly. The non spam message will be added to the Inbox Folder with a view challenge to the owner to add it to the whitelist when calling the message to be read. Figure (2) explains the flow of the message through the proposed filter.

The proposed filter uses probabilistic reasoning to decide whether or not a message is spam. This filter bases its choices on the Baye's rule, which is useful for calculating the probability of one event when one knows another event is true. In our case, the rule used to determine the probability that an e-mail is spam given that it contains certain words. What makes Bayesian filters different from other filters is that they learn. To decide the probability that an e-mail is spam based on the words that it contains, the filter needs to know about the e-mails that a user receives.

For the implementation of the Bayesian filter it is required to learn with a set of labeled messages. There are two stages carried out by the Bayesian filter:

i. Training Level

This level is called training or learning level. This level is focused on gathering the information, concerning both spam and legitimate e-mails. At this stage the filter extracts the tokens (words) of the labeled e-mail by an operation called tokenization that will be discussed later, and store them in tables. Two tables will be used, one for tokens of spam mails and other for tokens of legitimate mails. When an e-mail is declared as a spam, the spam table is updated by incrementing the frequency counts for each word contained in that e-mail. Legitimate e-mail counts are incremented similarly. The count number of spam and non spam e-mails is also recorded for use in the test level. We can get a list of spam mails from some dependable location in the web to learn filter with it. In addition, when the unlabeled message is labeled by the filter, it will be considered as input to learn with at the test level. This process is illustrated by the following algorithm:

Given an e-mail message X, labeled with C_j . . . Where j= {spam, legitimate}

1- Break X to tokens { x_i … x_n }, each token represents a word.
2- For each token x_i :

* POP3: Post office Protocol version. This protocol is used to retrieve mails from the mail server box to the mail client server.

If x_i exists in the table of type C_j, then freq [x_i] = freq [x_i] + 1.
Else freq [x_i] = 1.

3- Increment the e-mail count of type C_j : count [C_j] = count [C_j] +1.

ii. Testing Level

In the test level, the collected information about spam and non spam will be used as vectors to find the probability that the incoming e-mail is spam or not. This process is implemented by the following steps:

1- Compute the probability for each x_i, where the training information from the training level is represented as:

$$\Pr [x_i | C_j] = \text{freq} [x_i] / \text{total} [x_e]$$

where the freq (x_i) represents the frequency of a particular word in the incoming message and the total (x_e) represents the total frequencies for all words in the training information for all labeled C_j messages.

2- Compute the probability Pr [X| C_j]:

$$\Pr [X| C_j] = \Pr [x_1 | C_j] \Pr [x_2 | C_j] \ldots . \Pr [x_n | C_j]$$

$$= \prod_{i=1}^{k} \Pr[x_i|C_j]$$

3- Calculate Pr [C_j] which represents a probability of a message being a spam or non spam i.e., the frequency of spam or legitimate e-mails:

$$\Pr [C_S] = (\text{count} [C_S]) / (\text{count} [C_S] + \text{count} [C_n])$$

$$\Pr [C_n] = (\text{count} [C_n]) / (\text{count} [C_S] + \text{count} [C_n])$$

where (count [C_S]) is the count of spam e-mails, (count [C_n]) is the count of non spam e-mails.

4- For an unlabeled message, X, evaluate the quantities Pr [C_n |X] and Pr [C_S |X], where C_n denotes the class of legitimate e-mail messages and *Cs* is the class of spam e-mail messages:

$$\Pr [C_j |X] = \Pr [X| C_j] \Pr [C_j] / \Pr[X]$$

Pr[X] represents the estimated data about the incoming message, but it will be ignored because it has no effect on the first and second steps.

5- Label the message X as legitimate, if:

$$\Pr [C_n |X] > \Pr [C_S |X]$$

Else it is labeled as spam.

4-2 Features Extraction

A feature might be a single character, a word, an HTML token, a MIME* attachment etc. Spammers and users usually send simple language text information but they may use other types of information like URLs, links or e-mail addresses. These features may give some information to the filter. All frequently used words like (end, or, for, etc) will be removed. These words will be found in a list. In most tokenize machines the words are stemmed to the morphological words for example 'getting' is converted to 'get'.

The main advantages of applying word stemming and frequently used word removal is to take the actual value of the words for accuracy and possible improvement on classifiers' prediction accuracy by alleviating the data sparseness problem.

4-3 Tokenization

The purpose of tokenization is to transform the message in the mail corpus into a uniform format that can be understood by the learning and testing algorithms. Tokenizing consists of

* MIME: Multipurpose Internet Message Protocol. This protocol is used to attach with an e-mail messages, a multimedia file like video, audio, or image.

separating a message into a list of tokens (words). Tokenization may be implemented on different parts of a message as the header, body or the attachment. The results of the Bayesian filter are affected by the quality of the tokenization method. However, the real problems may come from the spammers shuffle.

4-4 Spammers Obfuscation

Spammers are working like the hackers and crackers as they are trying to succeed in spreading the spam messages to be read by the e-mail users. The spam filters are real main problems to the spammers. The spammers are trying to find new gaps in the filters to pass through. Therefore our proposed filter combines several techniques to make these gaps very difficult to be located by spammers. We will elaborate on some of these gaps and how they are manipulated by our filter.

As was explained in the introduction, the spammers misuse the SMTP to send the spam. Some of the spammers send the spam with a faked header. The blacklist may be unable to detect the real information from the header. Because of the dependence of the blacklist on the header analysis, more information in the message will be checked as the subject, from the address. However, spammers could repeat the same subject or most words in the subject to match it with the incoming message.

Spammers are using the obfuscation to evade many types of filters. Fortunately the presence of obfuscation is often a strong indicator of spam. One of this obfuscation is the random character in the html mail; an example is:

Fr<!--hr5kkj90-->om la<!—
ki61h7h6g9>st- - - - - - etc

The spammer uses the html comment to split the words to appear as unread words. To circumvent these attempts, we split the comments from the messages by adding more tolerations of spammers shuffle in our tokenize method. Such tokenization will remove the comments from the words.

Other type of spammer obfuscation on the mail is adding characters between the letters of the words that have high rated spam. For example: "Viagra" is represented as "V-i-a-g-r-a". This problem is solved as the following: the words that are not found in the tables will be isolated in the storage. Therefore the tokenization will remove all characters from every word, and pass them to the Bayesian testing. This toleration has been added to the tokenize method.

5- Results Evaluation

To examine the efficiency of the proposed filter, a set of inputs have been generated comprising spam and non-spam mails in a chronological order. The spam detected percentage and false positive percentage are calculated for each mail count. To check accuracy of the filter, one thousand of mails have been tested. Five hundred for each type of mail (spam and non-spam) has been considered. The results are illustrated Figure (3).

The false positive percent in the proposed filter decreases with the number of count mails. The spam detected percent increases with the increasing in the count of incoming messages. When the mails count reaches at least 280 mails, the two percentages almost stabilize showing less variation with the increasing count.

6- Conclusion

The proposed filter combining several techniques to filter the spam enhances the Bayesian filter in aspects of efficiency, speed and accuracy. Using this filter the users will be able to control what they need to read from the e-mail boxes. Also it reduces the time that may be lost by a human capability filter. Delphi language has been used to

develop a software package for the spam filtering at the client level. Delphi contains some efficient components that allow the control and retrieval of mails from the server mail box. The routines that comprise the proposed filter are developed as a dynamic link library. This allows the use of the spam filter within different environments.

The information collected for the Bayesian filter is stored in tables as SQL database which add power for storing and retrieving this information. To enhance the work and to add more accuracy, it would be desirable to extract more features from the message like colors, html attachment pictures and links. Each such feature will be expressed by a weight that depends on the user profile in training levels.

7-References

[1] L. Pelletier, J. Almhana, V. Choulakian, "Adaptive Filtering of SPAM",University of Moncton, E1A 3E9 , 2004.
https://www.umoncton.ca/greti/papers/Adaptive Filtering of Spam.pdf

[2] Le Zhang, Jingbo Zhu, Tianshun Yao , "An Evaluation of Statistical Spam Filtering Techniques", Natural Language Processing Laboratory Institute of Computer Software & Theory Northeastern University, China , 2004.
https://homepages.inf.ed.ac.uk/s0450736/paper/2004-spameval.pdf

[3] Anurag Garg, Roberto Battiti, Roberto Cascella, " Exchanging Filters to Combat Spam in Community", Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06), 2006.

[4] Mikko Siponen, Carl Stucke, "Effective Anti-Spam Strategies in companies: An International Study", Proceedings of the 39th Hawaii International Conference on System Sciences, 2007.

[5] Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, George Paliouras and Constantine D. Spyropoulos, "An Evaluation of Naive Bayesian Anti-Spam Filtering", Software and Knowledge Engineering Laboratory National Centre for Scientific Research "Demokritos", 2000
https://www.ics.forth.gr/~potamias/mlnia/paper_2.pdf

[6] William S. Yerazunis, "The Spam-Filtering Accuracy Plateau at 99.9% Accuracy and How to Get Past It", Presented at the 2004 MIT Spam Conference, MIT, Cambridge, Massachusetts, January 2004.
https://www.merl.com/reports/docs/TR2004-091.pdf

[7] Michael Davy, September, "Feature Extraction for Spam Classification", Masters thesis in Computer Science, Department of Computer Science, University of Dublin, Trinity College, September 2004.
https://www.cs.tcd.ie/publications/ tech-reports/reports.05/TCD-CS-2005-09.pdf

[8] Flavio D. Garcia, Jaap-Henk Hoepman, Jeroen van Nieuwenhuizen, "Spam Filter Analysis", University of Nijmegen, Netherlands, 2004.
https://www.cs.ru.nl/~flaviog/publications/spam-filter.pdf

[9] Md. Rafiqul Islam, Morshed Chowdhury, Wanlei Zhou, "An Innovative Spam Filtering Model Based on Support Vector Machine", Proceedings of the 2005 International Conference on Computational Intelligence for Modelling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and International Commerce (CIMCA-IAWTIC'05), 2005.

[10] Yang Li, Binxing Fang, Li Guo, Shen Wang, "Research of a Novel Anti-Spam Technique Based on Users' Feedback and Improved Naïve Bayesian Approach", Institute of

Computing Technology, Chinese Academy of Sciences, IEEE, 2006.

[11] Alistair McDonald, "SpamAssassin", Packt Publishing, September 2004 https://packtpub.com/files/1124_09.PDF

[12] Joseph Chiarella, Jason O'Brien "An Analysis of Spam Filters", a Major Qualifying Project Report, Project Number: CS-CEW-0203, April 2003, https://www.cs.wpi.edu/~claypool/mqp/spam/mqp.pdf

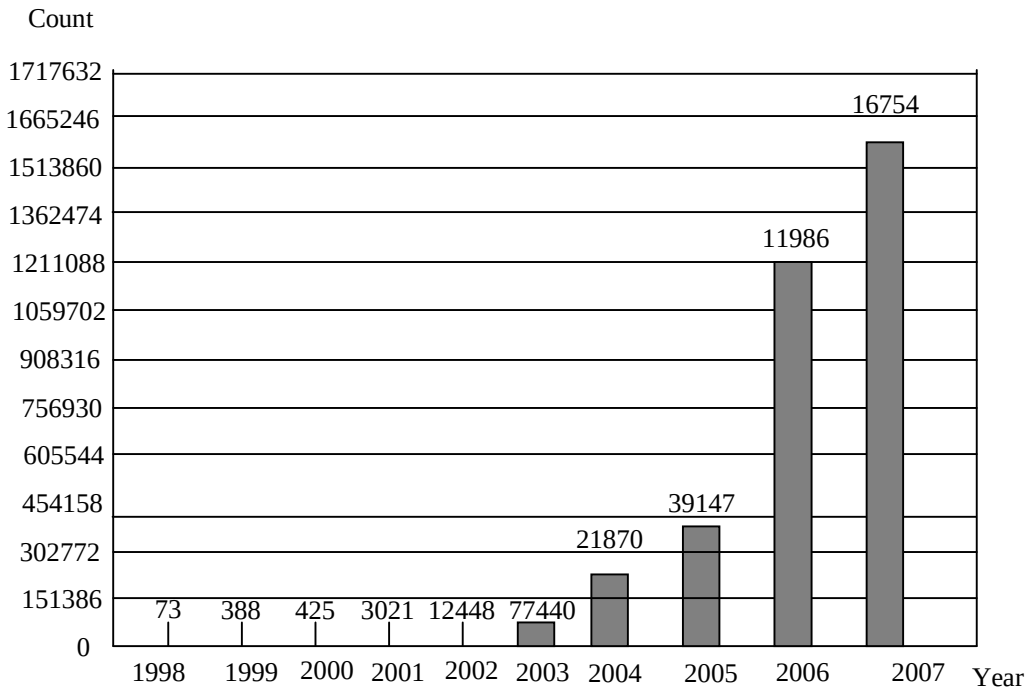

Figure (1) Annual Spam Evolution

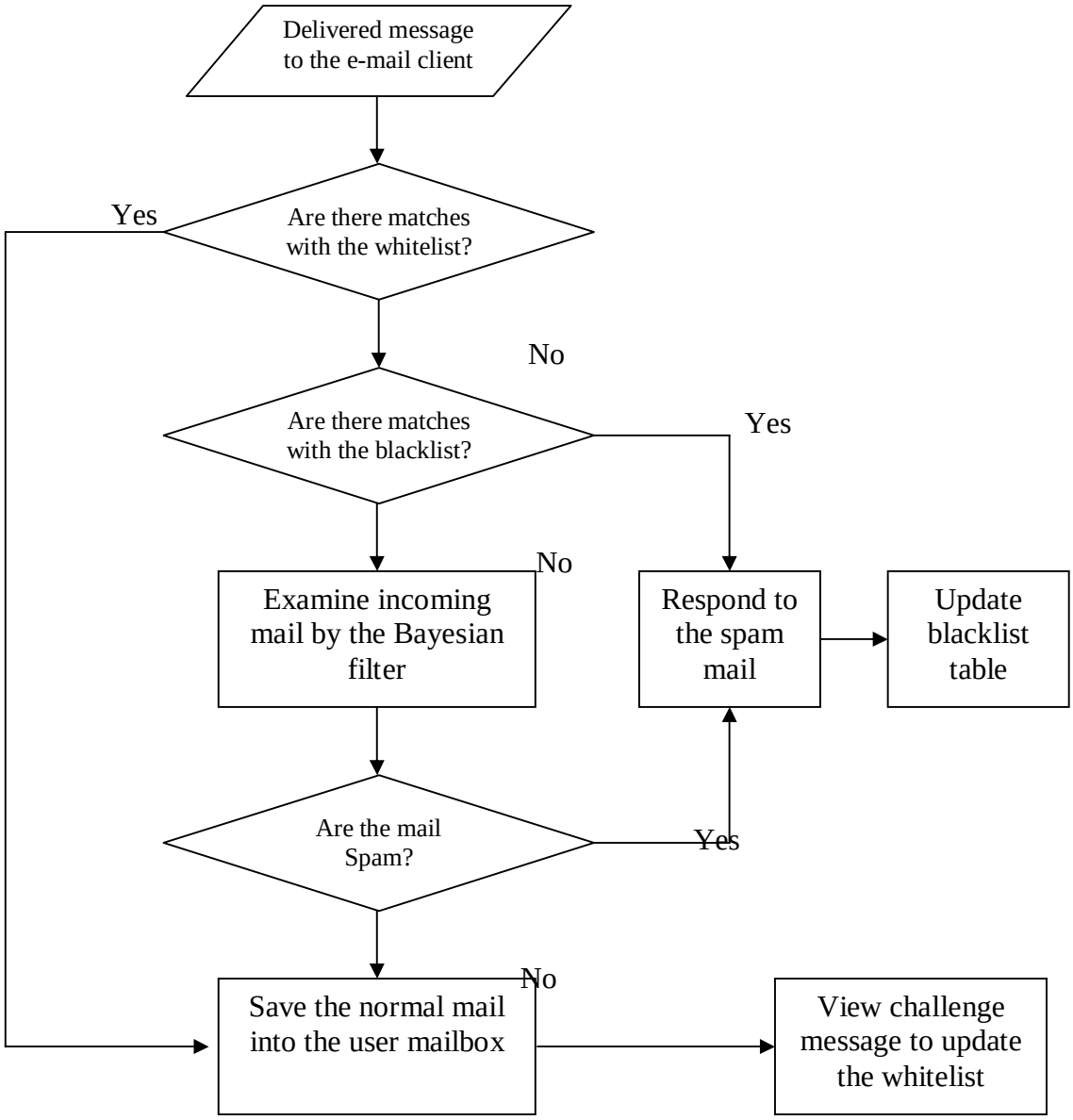

Figure (2) Flow of the Message through the Proposed Filter

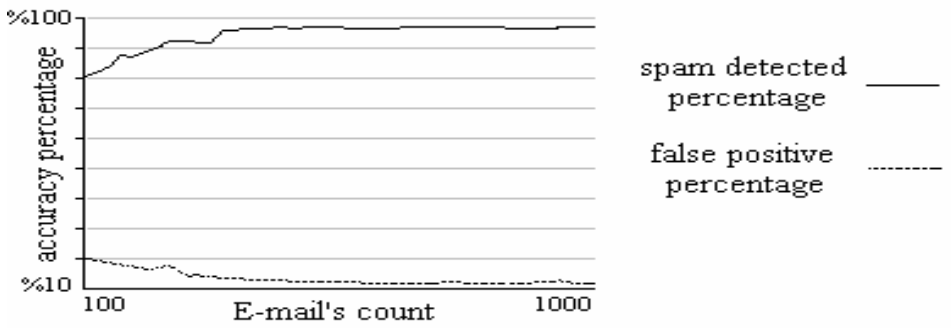

Fig (3) Spam Detected Percentage and False Positive Percentage, Versus E-mails Count Chart