An Intelligent SPAM filter – GetEmail5

Tarek Hassan, Peter Cole
School of Information Technology
Murdoch University,
Murdoch, WA, Australia 6150
{thassan | pcole}@murdoch.edu.au

Chun Che Fung
CECIS, School of Information Technology
Murdoch University,
Murdoch, WA, Australia 6150
l.fung@murdoch.edu.au

***Abstract*—As the increasing reliance on electronic mail (email) continues, unsolicited bulk email (SPAM) also continues to grow because it is a very cheap way for advertising. These unwanted emails are now causing a serious problem in clogging the internet traffic and filling up the email inboxes thereby leaving no space for legitimate emails to pass through. In addition, dealing with SPAM messages is costly to the users as it requires time and effort to examine them individually. In this paper, we propose an intelligent and trainable SPAM filter called GetEmail5. We have also evaluated the proposed filter against two commercial Filters, EmailProtect and SpamEater.**

Keywords—Unsolicited Bulk E-mail (UBE), Unsolicited Commercial E-mail (UCE), SPAM filter, Bayesian filter, black list, white list.

I. INTRODUCTION

As of 2004, 934 million people worldwide had access to the Internet, according to the Computer Industry Almanac. Between 1996 and 2004, the number of people worldwide who gained access to the Internet increased 16 times. Experts estimate another 35 percent increase by the end of 2005 [1]. The trend is expected to grow continually in the coming years.

One of the reasons for the exponential growth is the electronic mail (email) which has provided a cheap and near instantaneous mode of communication world-wide. Email has now become one of the most important applications of the Internet, and it poses as an indispensable communication tool rivals the traditional mail service. Email has also become an essential communication facility for business. A study has estimated that 5.48 trillion email messages were sent in 2002, 2.15 trillion of these messages were personal in nature, and 3.33 trillion were business related.

While Email has provided an unparallel means of global communication, unwanted email messages, "SPAM", also pose a serious problem resulting from ever increasing email traffic. The same study shows that 1.5 trillion of the messages were SPAM [2]. This figure is also increasing at an alarming rate.

There are two common definitions used to refer to SPAM; Unsolicited Bulk E-mail (UBE), or Unsolicited Commercial E-mail (UCE). The key aspect is "unsolicited", meaning that the recipients are never intended to receive the message in the first place. SPAM causes two serious problems, particularly for businesses:

1. Cost of dealing with SPAM.
2. By filling up the email inbox and thereby leaving no space for legitimate emails to pass through.

With respect to the above, the user should never have to deal with the SPAM messages. It therefore calls for some means of intelligent methods to block or filter off the unwanted messages or SPAM. The most common method to block SPAM and let the legitimate email messages to pass through are referred as *SPAM filter*. While there are many forms of such filters available, they rely on some forms of static rules to determine the legitimate of the messages. Such approaches either are ineffective or will take too long during the execution. The aim of this paper is to propose an intelligent SPAM Filter called GetEmail5. The proposed system has the ability to learn and adapt from the user's choices and establish a "black list" and "white list" of the messages and SPAM. The paper also presents an evaluation of the GetEmail5 against two popular commercial SPAM filters. It has been found that the proposed system is more effective and accurate in blocking the unwanted messages.

II. PROBLEMS CAUSED BY SPAM

Sending UCE is the cheapest form of direct online marketing. It can reach a huge number of people with minimal cost. Unlike traditional mail systems which incur posting and material costs, UCE only incurs Internet connection cost which is negligible as compared to the normal means of distribution such as postage or mailbox delivery as there are no printing or delivery costs [3].

According to a commissioned study on UCE and data protections, it was estimated that Junk email (SPAM) costs Internet users around 16 billion AUD a year worldwide [4]. Spam is considered to be a serious problem, because of the amount of resources involved in dealing with SPAM. The followings are examples of wastage incurred.

- Employee time spent on checking, interacting and removing SPAM emails.
- Network bandwidth (increasing traffic over the network)
- Network administrator's time required to spend dealing with SPAM (scanning, cleaning) and/or associated problems on viruses and malicious applications.
- Legal costs of pursuing spammers.

1-4244-0023-6/06/$20.00 ©2006 IEEE CIS 2006

Unfortunately, even up to now there is no clear distinction between the SPAM emails and those that might be considered a legitimate marketing strategy [5]. Hence, the burden of dealing the SPAM rests on the users or recipients. SPAM is truly a growing major problem and it is important to have an effective means to stem the growth of SPAM traffic which has already account 40-80% of the Internet traffic [6].

III. EXISTING SPAM FILTERS

A SPAM filter is a set of instructions for determining the status of the received email. SPAM filters are used to prevent SPAM email passing though to the recipient. The challenge is how to design an effective SPAM filter that allows desired email mail to pass through while blocking the unwanted SPAM emails. The potential unwanted problem is that often a SPAM filter may identify a legitimate email as a SPAM, and block it (false positive), or identify SPAM email as legitimate email, and allow it to pass through (false negative). Of these two cases, implications on the false positive can be very serious as important legitimate emails may not reach the receiver.

A means to quantify the effectiveness of a SPAM filter can be based on the percentage of SPAM emails being blocked, whilst allowing legitimate emails to pass through to the recipients. Listed below are three commonly used methods in SPAM filtering.

A. Black List Filter

Black list is effectively a list of emails that is not allowed to pass through. This can be based on the assumption that the email could contain a common word or phrase in the header, an IP address, or domain name. The use of a black list SPAM filter in isolation can result in false positive error. Assuming the word "*results*" is a keyword in the list, the following example will block both emails.

If the email header is (your exam results), another email has (use our product for quick results), what is going to happen is the filter will block both emails. (False positive) [7].

B. White List Filter

In this case, all the emails are treated as SPAM except the ones in the white list database (in other words, emails which will be accepted). The database is built using a confirmation process by the recipient. The problem with this time consuming technique is that it causes unnecessary burden to the users [8].

C. Bayesian Filter (Content Focus)

This approach is an extension of text classification technology, which searches the textual content of an email and uses algorithms to identify SPAM email. The algorithms are able to classify the occurrence of certain words and phrases in terms of how and where they appear in the email, not by their existence alone. The challenge with content filtering is that SPAM emails sometimes contain images, which are difficult to interpret their contents [9].

IV. PROPOSED INTELLIGENT APPROACH

No perfect SPAM filter has been found so far, the following proposed method [10] is aimed to develop an intelligent trainable SPAM filter that can block SPAM emails and let legitimate emails pass through using a combination of techniques including the use of the above approaches [11].

The proposed filter (GetEmail5) has been written with JAVA, and it is compatible with IMAP protocols only which make it easy to deal with flags The architecture and algorithm used in our approach is shown in Figure (1) below. Explanations for the components of the filter are also given the subsequent session.

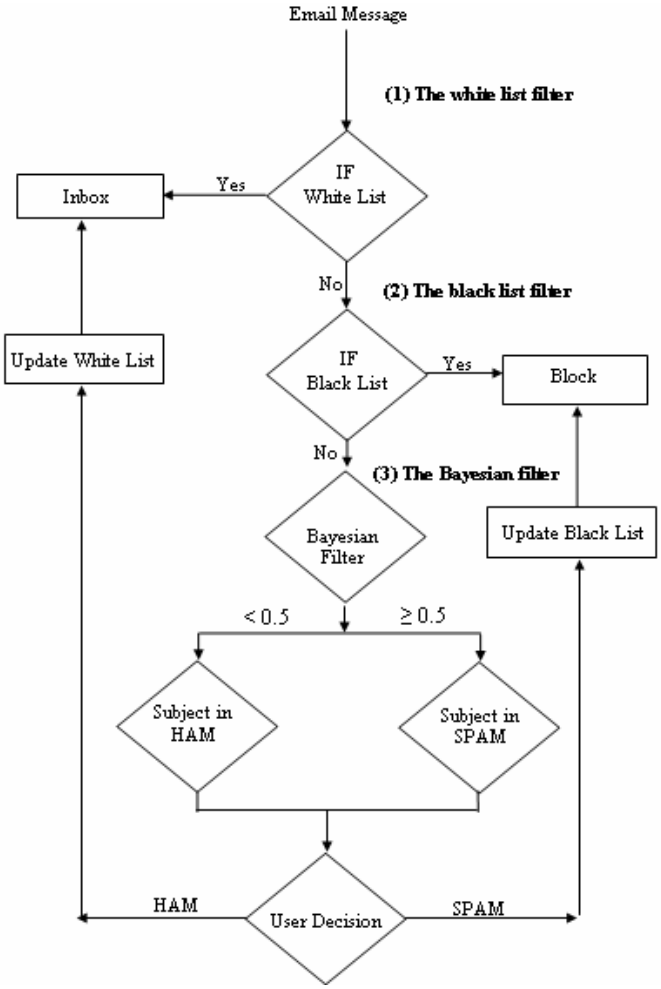

Figure 1. The proposed approach

Design of the filter is a combination of the existing techniques. The key contribution is the hybrid approach is the adaptation of an intelligent learning approach in building the white and black list known as "Ham" or "SPAM' messages.

1) The white list filter:

The filter checks the incoming email against the white list. If the email address is found in the white list, then the filter will allow the message to pass through to the INBOX.

2) The black list filter:

If the incoming email address is not found in the white list, then the filter will check the incoming email address against the black list. If a match is found the message is determine as a SPAM, and the filter will block it.

3) The Bayesian filter:

If the filter has not recognized the incoming message as a white list or a black list, the Bayesian filter will be applied on <SUBJECT> field and the content <BODY> of the message.

- The filter scans through the message, and creates a probability of every word it knows about. This probability value assigned to each word is commonly referred to as *spamicity*, and ranges from 0.0 to 1.0.
- If the spamicity value is greater than or equal to 0.5 then the message containing the word is likely to be SPAM and the program will prompt the user with the message: *"the incoming message is subject to SPAM do you want to add to BLACK LIST? "Y/N"*. If the user enters "*Y*", the message will be blocked, and the email address will be added to the black list array. If the user entered *"N"*, then the message will pass through to the INBOX, and the email address will be added to the white list array.
- If the spamicity value is less than 0.5, it means that a message containing the word is likely to be HAM, and program will prompt the user with the message: *"the incoming message is subject to HAM do you want to add to WHITE LIST? "Y/N"*. If the user entered *"Y"*, then the message will pass through to the INBOX, and the email address will be added to the white list array. If the user entered *"N"*, the message will be blocked, and the email address will be added to the black list array.

Implementation:

- A file is established to store the SPAM word list, which contains suspicious words. The filter uses the list to compare with the incoming messages, and this file is updated regularly.
- There is another file of the HAM word list, which stores unsuspicious words, which the filter uses to compare with the incoming messages, and this file is updated regularly by the user.

V. RESULTS

To carry out the experiment, an email account has been created, and **Prospect Mailer©** software has been used to send bulks of SPAM emails [12]. A set of SPAM emails was obtained from the SPAM archive website [13].

Initial test was preformed with a group of 615 emails (SPAM & Good emails), and the messages were divided as shown in Table (1).

TABLE I. SETS OF EMAILS AND SPAM DETECTED

Total No of Emails	No. of SPAM Detected	Subject to SPAM (User decision)	Subject to HAM (User decision)	White List
19 (5 good emails)	1	8	7	3
12 (All SPAM)	1	6	5	0
26 (All SPAM)	1	13	12	0
19 (All SPAM)	4	8	7	0
20 (3 good emails)	15	2	1	2
29 (3 good emails)	27	1	0	1
56 (4 good emails)	50	2	1	3
36 (3 good emails)	33	1	0	2
38 (All SPAM)	37	1	0	0
103 (3 good emails)	98	2	1	2
65 (2 good emails)	59	3	2	1
107 (5 good emails)	100	2	1	4
85 (All SPAM)	82	2	1	0
Total = 615	Total = 508	Total = 51	Total = 38	Total = 18

The first set was 19 emails (5 good emails, and 14 SPAM), and it is clear that the first set the filter detected only one SPAM, because it has not been fully trained. As more training was carried out, the system is capable to detect more SPAM messages.

Figure 2 shows the number of SPAM emails detected. It can be observed that as GetEmail5 received more updates, the filter provides more accurate results.

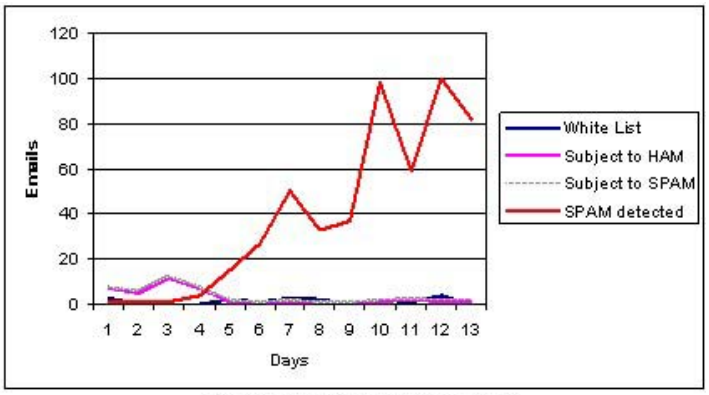

Figure 2. Number of Spam detected by GetEmail5

VI. EVALUATION

To evaluate the performance of the proposed filter, a comparison has been performed against other two commercial filters: **EmailProtect©**, and **SpamEater©.** The reasons why those two filters have been chosen, because of their ranks on 2005 SPAM filter review. Comparisons between different SPAM filters have been made, and the top ten SPAM filters have been chosen because of the feature set, ease of use, ease of setup/installation, stability, and customization. **EmailProtect©**, and **SpamEater©** were ranked number 1, and number 2 of the top ten SPAM filters [14].

Another two email accounts have been created for **EmailProtect©**, and **SpamEater©.** Both of them use Graphical User Interface (GUI), but GetEmail5 uses a command prompt.

A group of 100 emails (80 emails are SPAM, and 20 emails are good) were used to test the three filters. The results are shown in Table 2.

TABLE II. EVALUATION 1

	GetEmail5	EmailProtect	SpamEater
No. Emails	100 (80 Spam + 20 Good)	100 (80 Spam + 20 Good)	100 (80 Spam + 20 Good)
Filtering Time	2.22 min	2.30 min	4.05 min
Maintenance Time	2.50 min	4.00 min	7.10 min
No. Spam Detected	71	67	44
False Negative	0	11	0
False Positive	0	0	0
White List (GOOD)	19	20	18
Comments	**Prompt** 10 emails (user) 3 emails (HAM) 7 emails (SPAM)	**Manual** user (11 SPAM)	**Manual** User (36 SPAM) + (2 good) = 38

From Table 2, it is obvious that **GetEmail5** (command prompt) filtering time is less than filtering time of the other two filters (GUI). The maintenance time for **SpamEater** is long, because of the user has to train the filter manually (i.e. the user has to deal with 38 SPAM emails individually).

GetEmail5 prompts the user to enter (Yes/No) Y/N for the suspicious email, and that is why the maintenance time is less. Figure 3 shows the comparison of SPAM detection of the three filters, and it can be observed that the **GetEmail5** filter detects more SPAM than the other two filters.

Another group of 100 emails were used to confirm the results that obtained from the previous test, and the results are shown in Table 3. The comparison of the SPAM detection between the three filters is shown in Figure 4.

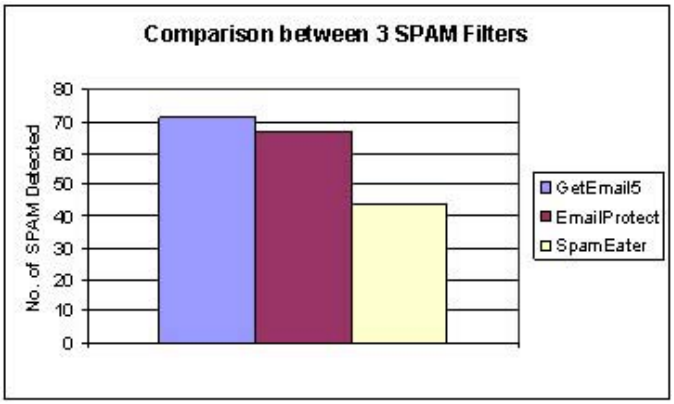

Figure 3. Number of SPAM detected by the three filters (I)

It can be observed that in the case of **GetEmail5**, the more user interaction the long maintenance time it will take (the user has to deal with 25 emails wither are they Good or SPAM). This can be improved by incorporating a dynamic variation on the threshold value for the Bayesian filter.

TABLE III. EVAUATION 2

	GetEmail5	EmailProtect	SpamEater
No. Emails	100 (80 Spam + 20 Good)	100 (80 Spam + 20 Good)	100 (80 Spam + 20 Good)
Filtering Time	4.50 min	3.33 min	4.39 min
Maintenance Time	4.55 min	4.09 min	8.50 min
No. Spam Detected	57	52	40
False Negative	2	28	0
False Positive	0	0	3
White List (GOOD)	16	20	13
Comments	**Prompt** 25 emails (user) 14 emails (HAM) 11 emails (SPAM)	**Manual** user (28 SPAM)	**Manual** User (40 SPAM) + (4 good) + (3 FP) = 47

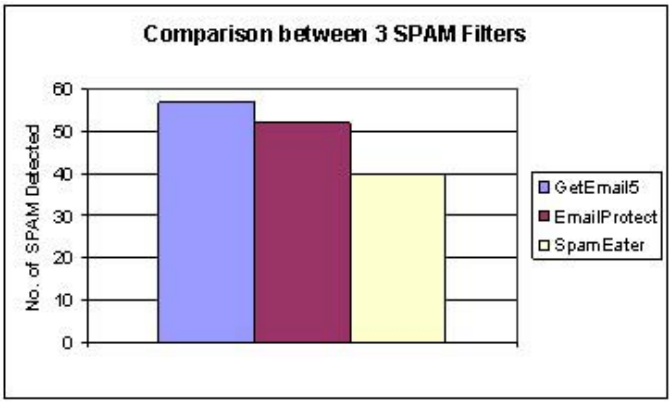

Figure 4. Number of SPAM detected by the three filters (II)

VII. CONCLUSION

This paper provided the background problem caused by SPAM emails, and it also described the methodology and the algorithm of a proposed SPAM filter (GetEmail5). The proposed system comprises a hybrid of the popular White List, Black List and Bayesian Filters approaches. The system is intelligent in the sense that it learns from the user's feedbacks and it is able to determine whether an incoming email message is a SPAM. Initial testing of GetEmial5 filter against two popular SPAM filters has demonstrated the superiority of the proposed approach. In particular, the system has improved its performance with increasing number of training. From the results provided in this paper, a promising approach to combat the problem of SPAM has been shown. We are aiming to improve the system by reducing the training requirements and to provide a dynamic adaptive threshold value used by the Bayesian Filter. We are also exploring the use of other computational intelligent techniques such as neural networks and fuzzy logic to augment the Bayesian Filter. In addition, we are going to add the ability to handle POP3 protocol, and automatic updating capability to the system.

REFERENCES

[1] Global Reach, "Global Internet Statistics" 30 Sept. 2004, (Accessed from: http://www.glreach.com/globstats)

[2] "Worldwide Email Usage Forecast, 2002-2006: Know What's Coming Your Way", IDC, pp 1-37, Sept 2002.

[3] M.Y. Schaub, "Unsolicited Email" Computer Law & Security Report vol. 18 no. 2, 2002.

[4] Internet Market, Retrieved 9th January 2004 (Accessed from: http://europa.eu.int/comm/internal_market)

[5] Eugene Schultz "Security views" Computers & security 0167- 4048/03.

[6] Cournane, A., & Hunt, R. 2004. "An Analysis of the tools used for the generation and prevention of spam." Computer & Security, 23 (2), 154-166.

[7] Moore D., Shannon C., Voelker G. M., and Savage S. "Internet Quarantine: Requirements for Containing Self- Propagating Code" IEEE INFOCOM April 2, 2003.

[8] Eric Allman "Features: Spam, Spam, Spam, Spam, Spam, the FTC, and Spam" Queue- Vol. 1 Issue 6, pages 62 - 69, September 2003.

[9] Androutsopoulos I., Koutsias J., Chandrinos K. V., Paliouras G., and Spyropoulos C. D. "An Evaluation of Naïve Bayesian Anti-Spam Filtering" 11th European Conference on Machine Learning- Barcelona, Spain, pp 9- 17, 2000.

[10] Tarek Hassan, Peter Cole and Chun Che Fung "Towards Eradication of SPAM: A Study on Intelligent Adaptive SPAM Filters" Proceedings of the 5th PEECS Symposium, Perth, Western Australia, pp 203-206, September 2004

[11] Bayesian Technique. Retrieved 10th September 2004 (Accessed from http://classifier4j.sourceforge.net)

[12] Prospect Mailer. Retrieved 20th March 2005 (Accessed from: http://www.marketing-2000.net/pm.htm)

[13] SPAM Archive. Retrieved 1st April 2005 (Accessed from http://www.spamarchive.org/)

[14] Email Protect, and SPAM Eater. Retrieved 1st April 2005 (Accessed from http://spam-filter-review.toptenreviews.com/)