International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

Filtering Spam: Current Trends and Techniques

Geerthik S.[1*] and Anish T. P.[2]

[1]Department of Computer Science, Sree Sastha College of Engineering, India
[2]Department of Computer Science, St.Peter's College of Engineering &tech, India
*Corresponding Author's E-mail: *geerthiks@gmail.com*

Abstract

This article gives an overview about latest trend and techniques in spam filtering. We analyzed the problems which is introduced by spam ,what spam actually do and how to measure the spam .This article mainly focuses on automated, non-interactive filters, with a broad review ranging from commercial implementations to ideas confined to current research papers. The solutions using both machine and non –machine learning approaches are reviewed and taxonomy of different approaches is presented. While a range of different techniques have and continue to be evaluated in academic research, heuristic and Bayesian filtering, along with its variants provide the greatest potential for future spam prevention.

Keywords: spam filter; Non-machine learning filters; machine learning filters; Bayesian-filters.

1. Introduction

In recent years, the increasing use of e-mail has led to the emergence and further escalation of problems caused by unsolicited bulk e-mail messages, commonly referred to as Spam. Evolving from a minor nuisance to a major concern, given the high circulating volume and offensive content of some of these messages, Spam is beginning to diminish the reliability of e-mail [1]. Personal users and companies are affected by Spam due to the network bandwidth wasted receiving these messages and the time spent by users distinguishing between Spam and normal (legitimate or ham) messages.

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

Dealing with spam and classify it is a very difficult task. A single model for classifying spam is also a difficult task since new spams are constantly evolving. Further, spammers often actively tailor their messages to avoid detection adding further impediment to accurate detection. Proposed solutions to spam can be separated into three broad types: legislation, protocol change and filtering. At current situation, legislation has appeared to have little effect on spam volumes, with some arguing that the law has contributed to an increase in spam by giving bulk advertisers permission to send spam, as long as certain rules are followed. Protocol changes have proposed to change the way in which we send email, including the required authentication of all senders, a per email charge and a method of encapsulating policy within the email address [2].

Such proposals, while often providing a near complete solution, generally fail to gain support given the scope of a major upgrade or replacement of existing email protocols. Interactive filters, often referred to as 'challenge-response'(C/R) systems, intercept incoming emails from unknown senders or those suspected of being spam. These messages are held by the recipient's email server, which issues a simple challenge to the sender to establish that the email came from a human sender rather than a bulk mailer. The underlying belief is that spammers will be uninterested in completing the 'challenge' given the huge volume of messages they sent; furthermore, if a fake email address is used by the sender, they will not receive the challenge.

On-interactive filters classify emails without human interaction and such filters often permit user interaction with the filter to customize user-specific options or to correct filter misclassifications; however, no human element is required during the initial classification decision. Such systems represent the most common solution to resolving the spam problem, precisely because of their capacity to execute their task without supervision and without requiring a fundamental change in underlying email protocols.

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(8), Jul, 2013, pp 208-223, ISSN: 2305-0543
Available online at: http://www.aeuso.org

2. Statistical Filter Classification and Evaluation

The experimental measures in spam include: spam recall (SR), spam precision (SP), F1 and accuracy (A). Spam recall is effectively spam accuracy. A legitimate email classified as spam is considered to be a 'false positive'; conversely, a spam message classified as legitimate is considered to be a 'false negative'. The mathematical terms for experimental measures for the evaluation of spam filters are given below.

$$\mathrm{A} = \frac{\#\text{ email correctly classified}}{\text{Total \# of emails}}$$

$$SR = \frac{\#\text{ spam correctly classified}}{\text{Total \# of spam messages}}$$

$$SP = \frac{\#\text{ spam correctly classified}}{\text{Total \# of messages classified as spam}}$$

$$F_1 = \frac{2 \times \mathrm{SP} \times \mathrm{SR}}{\mathrm{SP} + \mathrm{SR}}$$

The accuracy measure, while often quoted by product vendors, is generally not useful in evaluating anti-spam solutions. The level of misclassifications consists of both false positives and false negatives; clearly a 99% accuracy rate with 1% false negatives (and no false positives) is preferable to the same level of accuracy with 1% false positives (and no false negatives). The level of false positives and false negatives is of more interest than total system accuracy. [3] Suggests an alternative measurement technique - Receiver Operating Characteristics.

Such curves show the tradeoff between true positives and false positives as the classification threshold parameter within the filter is varied. If the curve corresponding to one filter is uniformly above that corresponding to another, it is reasonable to infer that its performance exceeds that of the other for any combination of evaluation weights and external

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

factors [4]; the performance differential can be quantified using the area under the curves. The area represents the probability that a randomly selected spam message will receive a higher ‘score' than a randomly selected legitimate email message, where the ‘score' is an indication of the likelihood that the message is spam. Filter classification strategies can separated into two categories: those based on machine learning (ML) principles and those not based on ML (Fig 1). ML approaches are capable of extracting knowledge from a set of messages supplied, and using the obtained information in the classification of newly received messages.

Non-machine learning techniques, such as heuristics, blacklisting and signatures, have been complemented in recent years with new, ML-based technologies. In the last few years, substantial academic research has taken place to evaluate new ML-based approaches to filtering spam. ML filtering techniques can be further categorized into complete and complementary solutions. Complementary solutions are designed to work as a component of a larger filtering system, offering support to the primary filter (whether it be ML or non-ML based). Complete solutions aim to construct a comprehensive knowledge base that allows them to classify all incoming messages independently.

Such complete solutions come in a variety of flavors: some aim to build a unified model, some compare incoming email to previous examples (previous likeness), while others use a collaborative approach, combining multiple classifiers to evaluate email (ensemble). Filtering solutions operate at one of two levels: at the mail server or as part of the user's mail program. Server-level filters examine the complete incoming email stream, and filter it based on a universal rule set for all users. Advantages of such an approach include centralized administration and maintenance, limited demands on the end user, and the ability to reject or discard email before it reaches the destination. User-level filters are based on a user's terminal, filtering incoming email from the network mail server as it arrives. They often form a part of a user's email program. ML-based solutions often work best when placed at the user

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

level [5], as the user is able to correct misclassifications and adjust rule sets. Software-based filters comprise many commercial and most open source products, which can operate at either the server or user level. Many software implementations will operate on a variety of hardware and software combinations. Appliance (hardware-based) on-site solutions use a piece of hardware dedicated to email filtering. These are generally quicker to deploy than a similar software-based solution, given that the device is likely to be transparent to network traffic. The appliance is likely to contain optimized hardware for spam filtering, leading to potentially better performance than a general-purpose machine running a software-based solution. Furthermore, general-purpose platforms, and in particular their operating systems, may have inherent security vulnerabilities while appliances may have pre-hardened operating systems.

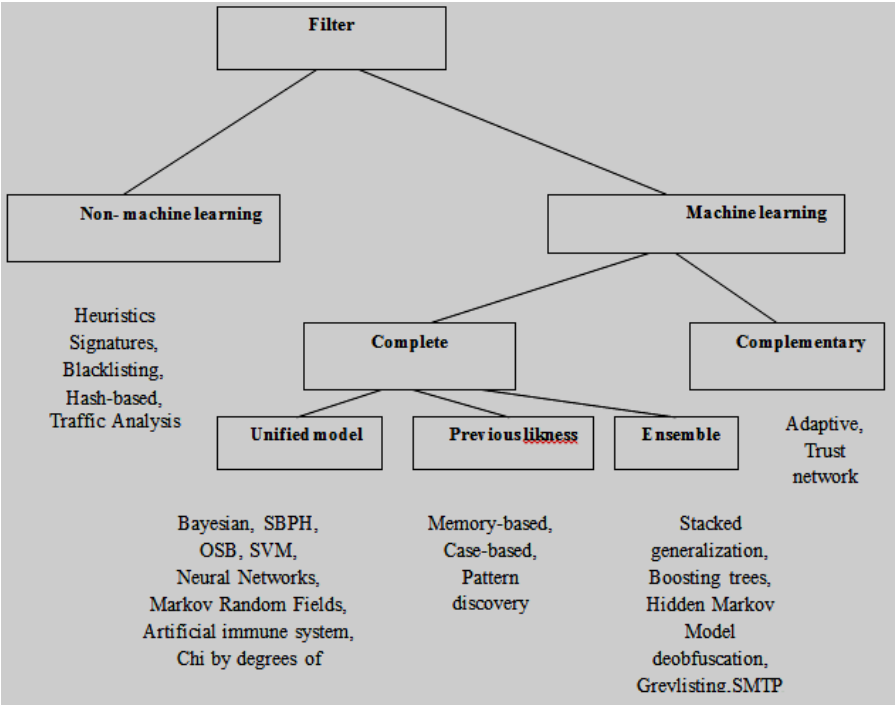

Figure 1. Classification of the various approaches to spam filtering

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

- -

3. Filter Technologies

3.1 Non-machine learning filters

3.1.1 Heuristics:

Heuristic, or rule-based, analysis uses regular expression rules to detect phrases or characteristics that are common to spam; the quantity and seriousness of the spam features identified will suggest the appropriate classification for the message. A simple heuristic filtering system may assign an email a score based upon the number of rules it matches. If an email's score is higher than a pre-defined threshold, the email will be classified as spam. The historical and current popularity of this technology has largely been driven by its simplicity, speed and consistent accuracy. Furthermore, it is superior to many advanced filtering technologies in the sense that it does not require a training period.

However, in light of new filtering technologies, it has several drawbacks. It is based on a static rule set: the system cannot adapt the filter to identify emerging spam characteristics. This requires the administrator to construct new detection heuristics or regularly download new generic rule sets. If a spammer can craft a message to penetrate the filter of a particular vendor, their messages will pass unhindered to all mail servers using that particular filter. Open source heuristic filters; provide both the filter and the rule set for download, allowing the spammer to test their message for its penetration ability.

3.1.2 Signatures:

Signature-based techniques generate a unique hash value (signature) for each known spam message. Signature filters compare the hash value of an incoming email against all stored hash values of previously identified spam emails. Signature generation techniques make it statistically improbable for a legitimate email message to have the same hash as a spam message. This allows signature filters to achieve a very low level of false positives. However,

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

signature-based filters are unable to identify spam emails until such time as the email has been reported as spam and its hash distributed. Furthermore, if the signature distribution network is disabled, local filters will be unable to catch newly created spam messages. Simple signature matching filters are trivial for spammers to work around. By inserting a string of random characters in each spam message sent, the hash value of each message will be changed. This has led to new, advanced hashing techniques, which can continue to match spam messages that have minor changes aimed at disguising the message. Spammers do have a window of opportunity to promote their messages before a signature is created and propagated amongst users.

Furthermore, for the signature filter to remain efficient, the database of spam hashes has to be properly managed. Commercial signature filters typically integrate with the organization's mail server and communicate with a centralized signature distribution server to receive and submit spam email signatures. Distributed and collaborative signature filters require sophisticated trust safeguards to prohibit the network's penetration and destruction by a malicious spammer while still allowing users to contribute spam signatures. Advances on basic signatures have been developed by [9] (combining hashing with document space density), [10] (use message digests, addresses of the originating mail servers and URLs within the message to improve spam identity) and [11] (personalized collaborative filters in conjunction with P2P networking).

Tokenization attacks in spam, insert characters that create divisions within words, causing incorrect representations of e-mails. [12] Introduce a new method that reverses the effects of tokenization attacks. This method processes e-mails iteratively by considering possible words, starting from the first token and compares the word candidates with a common dictionary to which spam words have been previously added. It provides an empirical study of how tokenization attacks affect the filtering capability of a Bayesian classifier and we show that our method can reverse the effects of tokenization attacks.

Vol. 3(8), Jul, 2013, pp 208-223, ISSN: 2305-0543
Available online at: http://www.aeuso.org

- -

3.1.3 Blacklisting:

Blacklisting is a simplistic technique that is common within nearly all filtering products. Also known as block lists, black lists filter out emails received from a specific sender. White lists, or allow lists, perform the opposite function, automatically allowing email from a specific sender. Such lists can be implemented at the user or server level, and represent a simple way to resolve minor imperfections created by other filtering techniques, without drastically overhauling the filter. Given the simplistic nature of technology, it is unsurprising that it can be easily penetrated. The sender's email address within an email can be faked, allowing spammers to easily bypass blacklists. Further, such lists often have a notoriously high rate of false positives, making them dangerous to use as a standalone filtering system.

3.1.4 Traffic analysis:

Characterization of spam traffic patterns is provided in [13]. By examining a number of email attributes, they are able to identify characteristics that separate spam from nonspam traffic. Several key workload aspects differentiate spam traffic; including the email arrival process, email size, number of recipients per email, and popularity and temporal locality among recipients.

3.2 Machine learning filters

3.2.1 Unified model filters:

Bayesian filtering now commonly forms a key part of many enterprise-scale filtering solutions as it addresses many of the shortcomings of heuristic filtering. No other machine learning or statistical filtering technique has achieved such widespread implementation and therefore represents the ‘state-of-the-art’ approach. Tokens and their associated probabilities are manipulated according to the user's classification decisions and the types of email received. Therefore each user's filter will classify emails differently, making it impossible for

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

a spammer to craft a message that bypasses a particular brand of filter. Bayesian filters can adapt their rule sets based on user feedback, which continually improves filter accuracy and allows detection of new spam types. Bayesian filters maintain two tables: one of spam tokens and one of 'ham' (legitimate) mail tokens. Associated with each spam token is a probability that the token suggests that the email is spam, and likewise for ham tokens. Probability values are initially established by training the filter to recognize spam and legitimate email, and are then continually updated based on email that the filter successfully classifies.

Incoming email is tokenized on arrival, and each token is matched with its probability value from the user's records. The probability associated with each token is then combined, using Bayes' Rules, to produce an overall probability that the email is spam. An example is provided in Fig. 2. Bayesian filters perform best when they operate on the user level, rather than at the network mail server level. Each user's email and definition of spam differs; therefore a token database populated with user-specific data will result in more accurate filtering [5].

Given the high levels of accuracy that a Bayesian filter can potentially provide, it has unsurprisingly emerged as a standard used to evaluate new filtering technologies. Despite such prominence, few Bayesian commercial filters are fully consistent with Bayes' Rules, creating their own artificial scoring systems rather than relying on the raw probabilities generated [6]. Furthermore, filters generally use 'naive' Bayesian filtering, which assumes that the occurrence of events is independent of each other. For example such filters do not consider that the words 'special' and 'offers' are more likely to appear together in spam email than in legitimate email.

International Journal of Mechatronics, Electrical and Computer Technology

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

- -

For example, the following set of keywords were extracted from an unseen email:

`prescription` (0.9) `tomorrow` (0.1) `student` (0.1) `james` (0.01) `quality` (0.85)

A value of 0.9 for prescription indicates 90% of previously seen emails that included that word were ultimately classified as spam, with the remaining 10% classified as legitimate email.

To calculate the overall probability of an email being spam (P):

$$\begin{aligned} P &= \frac{x_1 \cdot x_2 \cdots x_n}{x_1 \cdot x_2 \cdots x_n + (1-x_1) \cdot (1-x_2) \cdots (1-x_n)} \\ &= \frac{0.9 \cdot 0.1 \cdot 0.1 \cdot 0.01 \cdot 0.85}{0.9 \cdot 0.1 \cdot 0.1 \cdot 0.01 \cdot 0.85 + (1-0.9) \cdot (1-0.1) \cdot (1-0.1) \cdot (1-0.01) \cdot (1-0.85)} \\ &= 0.006 \text{ (to three decimal places)} \end{aligned}$$

This value indicates that it is unlikely that the email message is spam; however, the ultimate classification decision would depend on the decision boundary set by the filter.

Figure 2. A simple example of Bayesian filtering

In attempt to address this limitation of standard Bayesian filters, [7,8] introduced sparse binary polynomial hashing (SBPH) and orthogonal sparse bigrams (OSB). SBPH is a generalization of the naive Bayesian filtering method, with the ability to recognize mutating phrases in addition to individual words or tokens, and uses the Bayesian Chain Rule to combine the individual feature conditional probabilities. The method reported results that exceed 99.9% accuracy on real-time email without the use of white lists or blacklists. An acknowledged limitation of SBPH is that the method may be too computationally expensive; OSB generates a smaller feature set than SBPH, decreasing memory requirements and increasing speed. A filter based on OSB, along with the non-probabilistic Winnow algorithm as a replacement for the Bayesian Chain rule, saw accuracy peak at 99.68%, outperforming SBPH by 0.04%; however, OSB used just 600,000 features, substantially less than the 1,600,000 features required by SBPH. Support vector machines (SVMs) are generated by

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

mapping training data in a nonlinear manner to a higher-dimensional feature space, where a hyper plane is constructed which maximizes the margin between the sets. The hyper plane is then used as a nonlinear decision boundary when exposed to real-world data. [14] applied the technique to spam filtering, testing it against three other text classification algorithms. Both boosting trees and SVMs provide acceptable performance, with SVMs preferable given their lesser training requirements. A SVM-based filter for Microsoft Outlook has also been tested and evaluated [15].

This would allow a significant proportion of spam to be effectively filtered. This technique, when compared with a Bayesian filter, was found to provide equally good or better results. [16] presents a spam classifier based on a Markov Random Field (MRF) model. This approach allows the spam classifier to consider the importance of the neighborhood relationship between words in an email message. The inter-word dependence of natural language can therefore be incorporated into the classification process which is normally ignored by naive Bayesian classifiers. [17] introduces a unified spam filtering model for multi-source information, and proposes an approximate estimate method for the model portability. The model can be applied to multi-source information, and the classification algorithm achieved encouraging performance.

3.2.2 Previous likeness based filters

Memory-based, or instance-based, machine learning techniques classify incoming email according to their similarity to stored examples (i.e. training emails). Defined email attributes form a multi-dimensional space, where new instances are plotted as points. New instances are then assigned to the majority class of its k closest training instances, using the k-Nearest-Neighbor algorithm, which classifies the email. [18, 19] use a k-NN spam classifier, implemented using the TiMBL memory-based learning software.

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

Case-based reasoning (CBR) systems maintain their knowledge in a collection of previously classified cases, rather than in a set of rules. Incoming email is matched against similar cases in the system's collection, which provide guidance towards the correct classification of the email. The final classification, along with the email itself, then forms part of the system's collection for the classification of future email. [20] construct a case-based reasoning classifier that can track concept drift. They propose that the classifier both adds new cases and removes old cases from the system collection, allowing the system to adapt to the drift of characteristics in both spam and legitimate mail.

An initial evaluation of their classifier suggests that it outperforms naive Bayesian classification. [21] apply the Tiresias pattern discovery algorithm to email classification. Given a large collection of spam email, the algorithm identifies patterns that appear more than twice in the corpus. Experimental results are based on a training corpus of 88,000 items of spam and legitimate email. Spam precision was reported at 96.56%, with a false positive rate of 0.066%.

3.2.3 Ensemble filters

Stacked generalization is a method of combining classifiers, resulting in a classifier ensemble. Incoming email messages are first given to ensemble component classifiers whose individual decisions are combined to determine the class of the message. Improved performance is expected given that different ground-level classifiers generally make uncorrelated errors. [22] create an ensemble of two different classifiers: a naive Bayesian classifier [19, 23] and a memory based classifier [18]. Analysis of the two component classifiers indicated they tend to make uncorrelated errors. Unsurprisingly, the stacked classifier outperforms both of its component classifiers on a variety of measures. The boosting process combines many moderately accurate weak rules (decision stumps) to induce one accurate, arbitrarily deep, decision tree.[24] use the AdaBoost boosting algorithm and

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

compare its performance against spam classifiers based on decision trees, naive Bayesian and k-NN methods. They conclude that their boosting based methods outperform standard decision trees, naive Bayes, k-NN and stacking, with their classifier reporting F1 rates above 97% (Section 2). The AdaBoost algorithm provides a measure of confidence with its predictions, allowing the classification threshold to be varied to provide a very high precision classifier.Spammers typically use purpose-built applications to distribute their spam. Greylisting tries to deter spam by rejecting email from unfamiliar IP addresses, by replying with a soft fail.

It is built on the premise that the so-called 'spamware' does little or no error recovery, and will not retry to send the message. Careful system design can minimize the potential for lost legitimate email and greylisting is an effective technique for rejecting spam generated by poorly implemented spamware.SMTP Path Analysis [25] learns the reputation of IP addresses and email domains by examining the paths used to transmit known legitimate and spam email. It uses the 'received' line that the SMTP protocol requires that each SMTP relay add to the top of each email processed, which details its identity, the processing timestamp and the source of the message.

3.2.4 Complementary filters

Adaptive spam filtering [26] targets spam by category. It is proposed as an additional spam filtering layer. It divides an email corpus into several categories, each with a representative text. Incoming email is then compared with each category, and a resemblance ratio generated to determine the likely class of the email. When combined with Spamihilator, the adaptive filter caught 60% of the spam that passed through Spamihilator's keyword filter. [4] Identify a user's trusted network of correspondents with an automated graph method to distinguish between legitimate and spam email. The classifier was able to determine the class of 53% of all emails evaluated, with 100% accuracy. The authors intend this filter to be part of a more

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

comprehensive filtering system, with a content-based filter responsible for classifying the remaining messages. [7] constructed a similar network from 'trust' scores, assigned by users to people they know. Trust ratings can then be inferred about unknown users, if the users are connected via a mutual acquaintance(s).

Conclusion

Most modern spam-filtering solutions discussed are deployed on the receiver side. They are good at filtering spam for end users, but spam messages still keep wasting Internet bandwidth and the storage space of mail servers. Bloom filters [11] is intended to detect and nip spamming bots in the bud. By the technique it is found that account cracking events on 14 legitimate mail servers, on which some user accounts are cracked and abused for spamming. The method can effectively detect and curb the spamming bots with the precision and the recall up to 0.97 and 0.96.

[26] designed a method of dynamically updating the greylist and block list based on the delivery behavior of spammers to block spam sessions in time. This method is implemented on a mail gateway in a senior high school.The method can block 70.29% and 69.21% of the known and possible messages in the spam sessions with greylisting and block listing techniques. Therefore, the required system resources for further spam filtering on mail servers can be greatly saved by blocking most of the spam sessions in time

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

References

[1] B. Hoanca, "How good are our weapons in the spam wars", IEEE Technology and Society Magazine, pp 22–30, 2006.

[2] J. Ioannidis, "Fighting spam by encapsulating policy in email addresses", In Network and Distributed System Security Symposium, pp 6-7, Feb 2003.

[3] J. M. G. Hidalgo, "Evaluating cost-sensitive unsolicited bulk email categorization", In SAC '02: Proceedings of the ACM symposium on Applied computing, ACM Press, pp 615-620, 2002.

[4] Pin-Ren Chiou ,Po-Ching Lin , Chun-Ta Li, "Blocking spam sessions with greylisting and block listing based on client behavior ", IEEE Conference Publications, Jan 2013.

[5] F.D. Garcia, J.H. Hoepman, and J. van Nieuwenhuizen, "Spam filter analysis", In Proceedings of 19th IFIP International Information Security Conference, WCC2004-SEC, Toulouse, France,Kluwer Academic Publishers, Aug 2004.

[6] S. Vaughan-Nichols, "Saving private e-mail Spectrum", IEEE, pp 40-44, Aug 2003.

[7] J. Golbeck and J. Hendler, "Reputation network analysis for email filtering", In Conference on Email and Anti-Spam, 2004.

[8] Siefkes, F. Assis, S. Chhabra, and W. Yerazunis, "Combining winnow and orthogonal sparse bigrams for incremental spam filtering", In Proceedings of ECML/PKDD 2004, LNCS, Springer Verlag, 2004.

[9] K. Yoshida, F. Adachi, T. Washio, H. Motoda, T. Homma, A. Nakashima, H. Fujikawa, and K. Yamazaki, " Density-basedspam detector", In KDD '04: Proceedings of the 2004 ACM SIGKDD international conference on Knowledge discovery and data mining, ACM Press, pp 486-493, 2004.

[10] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati, " P2P-based collaborative spam detection and filtering", In P2P '04: Proceedings of the Fourth International Conference on Peer-to-Peer Computing (P2P'04), IEEE Computer Society, pages 176-183, 2004.

[11] Po-Ching ,Lin Ping-Hai Lin , Pin-Ren Chiou , Chien-Tsung Liu, "Detecting spamming activities by network monitoring with Bloom filters ", IEEE Conference Publications, Jan 2013.

[12] I. Santos, Laorden. C,Sanz. B, Bringas. P.G , "JURD: Joiner of Un-Readable Documents to reverse tokenization attacks to content-based spam filters ", IEEE Conference Publications, Jan 2013.

[13] L.H. Gomes, C. Cazita, J. Almeida, V. Almeida, and Jr. W. Meira, "Characterizing a spam traffic", In IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement,ACM Press, pp 356-369, 2004.

Vol. 3(8), Jul, 2013, pp 208-223, **ISSN**: 2305-0543
Available online at: http://www.aeuso.org

[14] H. Drucker, D. Wu, and V.N. Vapnik, "Support vector machines for spam categorization", IEEE Transactions on Neural Networks, vol 10(5), pp1048-1054, Sep 1999.

[15] M. Woitaszek, M. Shaaban, and R. Czernikowski, " Identifying junk electronic email in Microsoft outlook with a support vector machine", Symposium on Applications and the Internet, pp 166-169, Jan 27-31, 2003.

[16] S. Chhabra, W. Yerazunis, and C. Siefkes, " Spam filtering using a Markov random field model with variable weighting schemas", Fourth IEEE International Conference on Data Mining, pp 347-350, Nov 1-4, 2004.

[17] Wuying Liu ,Ting Wang, "Unimodel-based Multi-source Portable Spam Filtering", IEEE Conference Publications, Oct 2008.

[18] G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C. Spyropoulos, and P. Stamatopoulos, "A memory-based approach to anti-spam filtering. Technical report, DEMO 2001

[19] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C. Spyropoulos, and P. Stamatopoulos, "Learning to filter spam email: A comparison of a naive Bayesian and a memory-based approach, In Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases, 2000.

[20] P. Cunningham, N. Nowlan, S. Delany, and M. Haahr, "A casebased approach to spam filtering that can track concept drift", In ICCBR'03 Workshop on Long-Lived CBR Systems, June 2003.

[21] I. Rigoutsos and T. Huynh. Chung-kwei, " A pattern-discovery based system for the automatic identification of unsolicited email messages spam",Conference on Email and Anti-Spam, 2004.

[22] G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C.D. Spyropoulos, and P. Stamatopoulos, " Stacking classifiers for anti-spam filtering of e-mail", In Empirical Methods in Natural Language Processing, pp 44-50, 2001.

[23] I. Androutsopoulos, J. Koutsias, K. Chandrinos, and C. Spyropoulos, "An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages", In SIGIR '00: Proceedings of the 23rd annualinternational ACM SIGIR conference on Research and development in information retrieval, pp 160-167, ACM Press, 2000.

[24] X. Carreras and L. Marquez, "Boosting trees for anti-spam email filtering", In Proceedings of RANLP-01, 4th International Conference on Recent Advances in Natural Language Processing, Tzigov Chark, BG, 2001

[25] B. Leiba, J. Ossher, V. Rajan, R. Segal, and M. Wegman," SMTP path analysis", 2005.

[26] L. Pelletier, J. Almhana, and V. Choulakian ,"Adaptive filtering of spam", 2nd Annual Conference on Communication Networks and Services Research, pp 218-224, May 19-21, 2004.