Possibility Theory-Based Approach to Spam Email Detection

Dat Tran[(1)], Wanli Ma[(1)], Dharmendra Sharma[(1)], and Thien Nguyen[(2)]
[(1)]*University of Canberra, Australia,* [(2)]*De Anza College, CA, USA*
dat.tran@canberra.edu.au

Abstract

Most of current spam email detection systems use keywords in a blacklist to detect spam emails. However these keywords can be written as misspellings, for example "baank", "ba-nk" and "bankk" instead of "bank". Moreover, misspellings are changed from time to time and hence spam email detection system needs to constantly update the blacklist to detect spam emails containing such misspellings. However it is impossible to predict all possible misspellings for a given keyword to add those to the blacklist. We present a possibility theory-based approach to spam email detection to solve this problem. We consider every keyword in the blacklist along with its misspellings as a fuzzy set and propose a possibility function. This function will be used to calculate a possibility score for an unknown email. Using a proposed if-then rule and this core, we can decide whether or not this unknown email is spam. Experimental results are also presented.

1. Introduction

Spam emails are one type of the cyber nuisances we have to put up with everyday. The industry and the research community have been investing significant effort in fighting spam emails. Several methods have been proposed to detect spam emails, but the volume of those mails still continues to grow [1]. A server-based or client-based spam email detection system is currently used. For organizations, a server-based spam email detection system seems to be a good solution. It has more advantages and gives administrators more control to eliminate unsolicited emails sent in bulk [2]. However, this system should not eliminate emails based on their contents. A client-based solution allows the user to control the information to which they wish to be exposed and to determine which email is spam. This solution can reduce the number of spam messages received. Several methods for detecting spam emails have been proposed in the literature. They are based on address lists, headers, keyword lists and content statistical analysis known as Bayesian filter [3]. The address list-based method can be implemented at both client and server level. The address blacklist contains addresses from which all emails are blocked. The normal address list contains addresses of people the user knows and wishes to communicate with. When the user receives a new spam email, he or she will add the address

Similar to the address list-based system, the keyword list-based system has a blacklist of keywords which are the words used to detect spam emails. Keywords are collected from the subject, header or body of spam emails. The keyword list-based system is effective if it is made specifically for each user, therefore it is only appropriate for client-based systems or server-based systems for organizations who wish to control the information to which their staff members have access [2]. It takes time and effort to create a good keyword list and this list needs to be regularly updated in order to make them as effective as possible. Although there are not many keywords found in spam emails, but the problem for detection is that these keywords are written as misspelling words and change their misspellings from time to time. Users can understand the content of the email containing such misspellings but the keyword list-based system is unable to update the blacklist with those misspellings. For example, an email is regarded as a spam email because it contains the keyword "*banking*". After updating the blacklist with this keyword, the system is still unable to detect spam emails containing "*baanking*", "*bank_ing*", "*bbanking*", or "*bank1ng*". Since there are numerous ways to produce misspellings for a given word, the email detection system becomes ineffective. Moreover, recent spam emails have messages in image format rather than text format. It was estimated that 38% of spam emails contains images [4]. Given the fact that image based spam can successfully circumvent spam filters, the situation can only get worse in the future.

In this paper, we propose a possibility theory-based approach to spam email detection systems that use a blacklist of keywords to detect spam emails. We

0-7695-3032-X/07 $25.00 © 2007 IEEE
DOI 10.1109/GrC.2007.123

consider every keyword in the blacklist and its misspellings as a fuzzy set and determine a possibility distribution function. These functions for all keywords will be used to calculate a possibility score for an unknown email. Using an if-then rule and this core, we can decide whether or not this unknown email is spam. Experimental results are also presented to evaluate the proposed approach. We also compare experimental results with those obtained from Trigram approach.

The rest of the paper is as follows. Section 2 presents the Trigram approach to detect spam emails. Section 3 presents the possibility theory-based approach to spam email detection. Section 4 presents experimental results. Finally, we conclude the paper in Section 5.

2. Trigram Approach

Current methods for text analysis are based on the n-gram approach [5-7]. The popular trigram-based method analyzes a text document as a set of trigrams, i.e. sequences of three letters. For example, the trigram method analyses the string "*text document*" as the following sequence: *tex, ext, xt_, doc, ocu, cum, ume, men, ent*, and *nt_*.

The probability of a given trigram is the ratio of its frequency to the sum of the frequencies of all the trigrams. The probability set of all the trigrams obtained from a training document is stored and regarded as a trigram model for that document. For other trigrams that are not in the model, their probability will be equal to 0. In order to identify an unknown text, this text is also analyzed into a sequence of trigrams and the probability of the sequence is calculated using the probability set in the trigram model. The trigram-based system [5] was achieved good performance when test strings were about 50 through 700 words using the n-gram frequency of 400.

The learning and detection procedures for the trigram approach are summarized as follows.

Learning:

- Given N keywords in a blacklist.
- Analyze the N keywords to obtain a list of trigrams then calculate the probability for each trigram.
- The list of trigrams and their probability is regarded as a blacklist model.

Detection

- Given an unknown email and a preset threshold.
- Analyze the unknown email to obtain a list of trigrams.
- Use the probabilities in the blacklist model to calculate a score for the email.
- If the score is greater than a preset threshold, the unknown email is regarded as a spam email.

3. Possibility Theory-Based Approach

Possibility theory has been proposed by Zadeh [8], where fuzzy variables are associated with possibility distributions in the similar way that random variables are associated with probability distributions [9, 10]. The development of possibility theory has led to a theory framework similar to that of probability theory. Possibility theory offers a simple, non-additive modeling of partial belief.

3.1. Possibility measure

Let Ω be the universe of discourse Ω and be a finite set. Assume that all subsets are measurable. A distribution of possibility is a function $\Pi(A)$ from Ω to [0, 1] and satisfies the following conditions

$\Pi(\varnothing) = 0$
$\Pi(\Omega) = 1$ (normalization)
$\Pi(A \cup B) = \max(\Pi(A), \Pi(B))$ for any disjoint subsets A and B

3.2. Possibility functions for keyword and blacklist

Consider a blacklist consisting of N keywords used to detect spam emails. Let $B = \bigcup_{k=1}^{N} W_k$ be a fuzzy set representing the blacklist and W_k is a fuzzy subset consisting of the k-th keyword in the blacklist and misspelling words of that keyword.

The possibility function $\Pi(B)$ is determined as follows

$$\Pi(B) = \max_{W_k \subset B} \Pi(W_k) \quad (1)$$

Consider a fuzzy subset W. Let w be the keyword in W and x be any misspelling word of w. Let $w = w_1 w_2 ... w_{T_w}$ where w_i is the i-th alphabetical letter in w and T_w is word length. Similarly, let $x = x_1 x_2 ... x_{T_x}$ be a word of length T_x. The possibility of x to be a misspelling word of w can be determined as follows

$$\Pi(x \mid W) = \frac{q(x_1, w_1) + \sum_{t=1}^{T_x} p((x_t, x_{t+1}), (w_k, w_{k+1}))}{\max\{T_x, T_w\}} \quad (2)$$

where $q(x_1, w_1) = 1$ if $x_1 = w_1$ and $q(x_1, w_1) = 0$ if $x_1 \neq w_1$; and $p((x_t, x_{t+1}), (w_k, w_{k+1})) = 1$ if pair $(x_t, x_{t+1}) = (w_k, w_{k+1})$ and $p((x_t, x_{t+1}), (w_k, w_{k+1})) = 0$ if $(x_t, x_{t+1}) \neq (w_k, w_{k+1})$ for any integer k such that $1 \leq k < T_w$.

An example for the possibility function in (1) is as follows. Consider the keyword "*bank*" and two misspellings "*baank*" and "*aank*". The keyword "*bank*" has the first letter '*b*' and letter pairs "*ba*", "*an*", and "*nk*". Using (1) we obtain the following $\Pi(baank, W) = 4/5 = 0.8$, $\Pi(aank, W) = 2/4 = 0.5$ and $\Pi(bank, W) = 1$.

3.3 Possibility function for an email message

An email message E can be regarded as a set of words. This message can be in text or image format. If the message is in image format, an optical character recognition (OCR) tool [11] will be used to extract text in the message.

In order to decide the email E to be a spam email, we calculate the possibility function value of the email E given the fuzzy set B of the blacklist. Let x be any word in E, we have

$$\Pi(E \mid B) = \max_{x \in E} \Pi(x \mid B) = \max_{x \in E} \{ \max_{W_k \subset B} \Pi(x \mid W_k) \} \quad (3)$$

Let θ be a preset threshold. The following if-then rule is used to decide if the email E is as a spam email

$$\text{If } \Pi(E \mid B) > \theta \text{ then } E \text{ is a spam email} \quad (4)$$

For example, assume θ = 0.7 and "*bank*" is a keyword in the blacklist, according to the example considered above, an email is regarded as a spam email if it contains "*bank*" or "*baank*".

The learning and detection procedures for the proposed approach are summarized as follows:

Learning:

- Given N keywords in a blacklist.
- Analyze the N keywords to obtain a list of first letters and pairs of letters.

Detection

- Given an unknown email and a preset threshold.
- Analyze the unknown email to obtain a list of first letters and pairs of letters.
- Use (3) to calculate possibility function value for each email
- Use (4) to decide if the unknown email is a spam email.

4. Experimental Results

It is obvious that spam email detection rate is very high for email text messages containing exact keywords. The challenge is spam emails containing messages as image attachments.

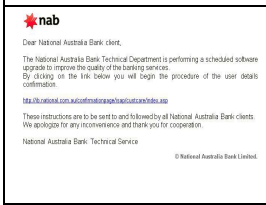
Picture	*Text extracted*
	_ _ ____ ____ _ _ VIAGRAsoft __ CIALISsoft _ $ 3.66 _ $ 3.78 per I * mg per * mg LEVITRA __ VIAGRA m: $ 4.90 t $ 3.OO per * mg per I* mg
	_ ____ nab Dear Natlonal Australla Bank client The Natlonal Australla Bank Technlcal Depa_ment Is performl ng a scheduled so_ware upgrade to Improve the quallty of the banklng services By cllcklng on the ll nk
	_,,,,,,,,,_,, ,,,,,_,,,,, g.,,,,,_;,_,,gu,H ,,,, ,,,,,,,,, _, Q,.,,,,.,a,.,,,,,,,, /,M, g_

Fig. 1: Images and recognized texts obtained by OCR

We collected 100 emails and there were 33 spam emails containing image attachment. We used the OCR tool [11] to extract text messages in these 33 spam emails. It was first pre-processed to remove all special, common characters and punctuation marks such as commas, columns, semi-columns, quotes, stops, exclamation marks, question marks, signs, etc. The next step was to convert all the characters into lowercases.

There were 50 keywords collected and about 10,000 normal words found in the total 100 emails. Using the possibility function in (3) and the decision rule in (4) to detect spam email, the spam email detection rate we could achieve was 91%.

We also tested those emails using the Trigram method and the detection rate obtained was the same. This preliminary experiment is very encouraging, and we will anticipate better results in the near future.

5. Conclusion

We have presented a possibility theory-based approach to spam email detection problem. We have tested on 67 normal emails and 33 spam emails. We have also compared this approach with the Trigram method. Our next step is to conduct a large scale testing on the effectiveness of the approach based on spam emails available from SpamArchive corpus and our own private collections.

Finally, we are aware of that the spammers are trying to sabotage OCR tools by obfuscating images, such as introducing noises to the images and skewing and rotating the text on the images etc. We believe that we have found a solution to recognize these images used for spam purposes. We are conducting more experiment to test the solution. And on the other hand, as the final note, spammers do not spam for fun. They do it for financial return. The obfuscated images, although still humanly readable, will not bring the expected return to the spammers, as human readers are less likely to respond to this type of images. Spammers try to circumvent the spam email filters, but still prefer that the emails look normal so that they will not raise human alarm. In essence, the purpose of spam emails is to deliver messages, and in a nice format. This is the Achilles' heel of spam emails, and it is also the key for us to fight spam emails.

10. References

[1] S. L. Pfleeger and G. Bloom, "Canning Spam: Proposed Solutions to Unwanted Email", in IEEE Security & Privacy, (2005) 40-47.

[2] L. Pelletier, J. Almhana, V. Choulakian, "Adaptive Filtering of SPAM", in Proceedings of the Second Annual Conference on Communication Networks and Services Research (CNSR'04).

[3] P. Graham, "Better Bayesian Filtering", retrieve from the website http://paulgraham.com/better.html (2003)

[4] Wu, C.-T., K.-T. Cheng, et al. Using visual features for anti-spam filtering. in IEEE International Conference on Image Processing, 2005 (ICIP 2005). 2005.

[5] Cavnar, W.B. and J.M. Trenkle. "N-gram-based text categorization", in the 3rd Annual Symp. Document Analysis and Information, Retrieval, 1994.

[6] Muthusamy, Y.K. and A.L. Spitz, Automatic language identification, in Survey of the state of the art in human language technology, R.A. Cole, et al., Editors. Cambridge University Press, 1998.

[7] Schmitt, J.C., "Trigram-based method of language identification", in U.S. Patent number: 5062143, 1991

[8] Zadeh L. A. "Fuzzy sets as a basis for a theory of possibility", Fuzzy Sets and Systems, vol. 1, no. 1, pp. 3-28, 1978.

[9] Dubois D. and Prade H. Possibility Theory: An Approach to Computerized Processing of Uncertainty Plenum Press, New York, 1988.

[10] Tanaka H. Possibilistic Data Analysis for Operations Research, Physica-Verlag, A Springer-Verlag Company, Germany, 1999.

[11] Ma, W., D. Tran, and D. Sharma. Detecting Image Based Spam Email by Using OCR and Trigram Method. in International Workshop on Security Engineering and Information Technology on High Performance Network (SIT2006). 2006. Cheju Island, Korea.

[12] Cormack, G. and T. Lynam. TREC 2005 Spam Track Overview. in The Fourteenth TExt Retrieval Conference (TREC 2005). 2005. Gaithersburg, MD, USA.

[13] Sahami, M., S. Dumais, et al. A Bayesian Approach to Filtering Junk E-mail. in AAAI-98 Workshop on Learning for Text Categorization. 1998.

[14] Sakkis, G., I. Androutsopoulos, et al., A Memory-Based Approach to Anti-Spam Filtering for Mailing Lists. INFORMATION RETRIEVAL, 2003. **6**(1): p. 49-73.

[15] Carreras, X. and L. Marquez. Boosting Trees for Anti-Spam Email Filtering. in 4th International Conference on Recent Advances in Natural Language Processing (RANLP-2001). 2001.

[16] Zhang, L. and T.-s. Yao. Filtering Junk Mail with A Maximum Entropy Model. in 20th International Conference on Computer Processing of Oriental Languages (ICCPOL03). 2003.

[17] Drucker, H., D. Wu, and V.N. Vapnik, Support vector machines for spam categorization. IEEE Transactions on Neural Networks, 1999. **10**(5): p. 1048-1054.

[18] Chuan, Z., L. Xianliang, et al., A LVQ-based neural network anti-spam email approach. ACM SIGOPS Operating Systems Review, 2005. **39**(1): p. 34 - 39.

[19] Zhou, Y., M.S. Mulekar, and P. Nerellapalli. Adaptive Spam Filtering Using Dynamic Feature Space. in 17th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'05). 2005.

[20] Graham-Cumming, J. The Spammers' Compendium. 2006 15 May 2006 [cited 2006 May]; Available from: http://www.jgc.org/tsc/.

[21] Cockeyed. There are 600,426,974,379,824,381,952 ways to spell Viagra. 2006 [cited 2006 October 2006]; Available from: http://cockeyed.com/lessons/viagra/viagra.html.

[22] Damiani, E., S.D.C.d. Vimercati, et al. P2P-based collaborative spam detection and filtering. in 4th IEEE

International Conference on Peer-to-Peer Computing (P2P'04). 2004. Zurich, Switzerland.

[23] Albrecht, K., N. Burri, and R. Wattenhofer. Spamato - An Extendable Spam Filter System. in 2nd Conference on Email and Anti-Spam (CEAS'05). 2005. Stanford University, Palo Alto, California, USA.

[24] Yerazunis, W.S., S. Chhabra, et al., A Unified Model of Spam Filtration. 2005, Mitsubishi Electric Research Laboratories, Inc: 201 Broadway, Cambridge, Massachusetts 02139, USA.

[25] Postel, J.B. Simple Mail Transfer Protocol. 1982 [cited 2006 May]; Available from: http://www.ietf.org/rfc/rfc0821.txt.

[26] Freed, N. and N. Borenstein. Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types. 1996 [cited 2006 May]; Available from: http://www.ietf.org/rfc/rfc2046.txt.

[27] Ma, W., D. Tran, et al. Detecting Spam Email by Extracting Keywords from Image Attachments. in Asia-Pacific Workshop On Visual Information Processing (VIP2006). 2006. Beijing, China.

[28] Tran, D., W. Ma, and D. Sharma. Fuzzy Normalization for Spam Email Detection. in Proceedings of SCIS & ISIS. 2006.

[29] Tran, D., W. Ma, and D. Sharma. A Noise Tolerant Spam Email Detection Engine. in the 5th Workshop on the Internet, Telecommunications and Signal Processing (WITSP'06). 2006. Hobart, Australia.

[30] Tran, D., W. Ma, et al. A Proposed Statistical Model for Spam Email Detection. in Proceedings of the First International Conference on Theories and Applications of Computer Science (ICTAC 2006). 2006.