Coupled Behavioral Analysis for User Preference-based Email Spamming

Frank Jiang*, Jin Gan#, Yuanyuan Xu#, Guandong Xu*
*Faculty of Engineer of IT, University of Technology Sydney, Australia;
#College of Electronics and Engineering, Guangxi Normal University
Email: Frank.Jiang@uts.edu.au

Abstract. In this paper, we develop and implement a new email spamming system leveraged by coupled text similarity analysis on user preference and a virtual meta-layer user-based email network, we take the social networks or campus LAN networks as the spam social network scenario. Fewer current practices exploit social networking initiatives to assist in spam filtering. Social network has essentially a large number of accounts features and attributes to be considered.

Instead of considering large amount of users accounts features, we construct a new model called meta-layer email network which can reduce these features by only considering individual user's actions as an indicator of user preference, these common user actions are considered to construct a social behavior-based email network. With the further analytic results from text similarity measurements for each individual email contents, the behavior-based virtual email network can be improved with much higher accuracy on user preferences. Further, a coupled selection model is developed for this email network, we are able to consider all relevant factors/features in a whole and recommend the emails practically to the user individually. The experimental results show the new approach can achieve higher precision and accuracy with better email ranking in favor of personalised preference.

Keywords Email filtering; Top k selections; Naïve Bayesian classifier, Support vector machines

1. Introduction

Nowadays, billions of internet end-users and device to device connections contribute to the significant data growth in recent years. Not only the volume of the data is increased but also the relationships among the items and objects. The current hot term Big Data reflects these observations, which are characterized by the unstructured, multi-dimensional and complex measures. The existence of adversaries and intruders indicates that future data can be deliberately constructed to maliciously increase the error rate of prediction models, where words and images are intelligently transformed by the senders of spam in an effort to deceive spam filters.

The specific issues for the new spam system include: 1) Not personalised; 2) comparatively static association rules defined in the firewalls, or gateways; 3) cannot identify the extremely hidden information that mixed in the syntax or semantics.

Moreover, the conventional i.i.d. ness-based learning methods cannot handle the new data effectively any more, for instance, these methods cannot efficiently find the inter-relationship and intra-relationship with such scale [1] [2], and they do not scale well and nor do they perform well under highly unstructured, unpredictable conditions (data volume, data variety, data categories etc.).

The most recent approaches for email spam can be seen in these work [4-7]. The approaches can be generally grouped into two categories: (1) based on email content, (2) based on email header. The first approach has low error but high computational cost. The second approach has lower computational cost but with higher error rate. In the first category, statistical methods such as Bayesian Classification or Support Vector Machines (SVM) are used to filter emails Bayesian Classification. It was introduced by Paul Graham in 1998 [8], and further addressed by Androutsopoulos [9]. In Bayesian Classification, the scores for keywords in the email content are used to judge the email as Spam or Ham. This method depends highly on the quality of the training dataset. In this method, contents of training emails will be extracted into tokens and store in a database. In the presence of the current data volume, variety and categories nowadays, the first category of techniques often show higher correct rate but low performance. The second category of techniques often have higher performance but also higher error rate.

In this paper, we propose a new framework of email recommender system using user actions and statistical methods. Instead of labeling emails as SPAM or HAM, we label emails with the personalized importance ranking based on user actions and preferences. The possible number of user's actions may vary but mainly falls into the following general categories: (a) reply, (b) read, (c) forward, (d) delete or mark as spam. By using this approach, we can not only filter spam, but also suggest the action for users and prioritize emails by different actions based on user preferences. This method remarkably improves the time of processing new emails and is based on user preferences. Not only limited to this, we also consider the text mining results from the main contents of the email body, whose results are used to further fine tune the user action network for recommender purposes.

This paper is organized as follows: Section 2 introduces the preliminaries of four spam filtering algorithms including our own algorithm. Section 3 further presents the prioritisation theory and its experiment validation. A description of the dataset is also included. Section 4 investigate the classification problem in our new user-action based meta-email network, the classifiers are considered by use of Naïve Bayesian Classifier and Support Vector Machines. Experimental results show the promising performance of the new meta-email network by use of these two classifiers. Section 5 concludes and discusses the future work of this research.

2. PRELIMINARIES

In this section, we focus on reviewing new methods to detect spam based on the theory of the complex networks. Each method, obviously, has advantages and disadvantages. Firstly, we introduce three popular spam filtering methods and one improved spam detection algorithm developed by authors in the past [28] [29] [30]. In the next theory section of this paper, this improved filtering scheme is integrated in part with the meta-email network for user ranking particularly. Secondly, we summarize the current text mining methods.

2.1 Method based on Clustering Coefficient

Boykin and Roychowdhury [19] propose a solution to detect spam based on the clustering coefficient. The authors collect emails from their own personal mailbox to build an email network, in which email addresses are nodes and links on the sender-receiver nodes are considered as edges. In this method, the email exchanges between the set of users are modeled as a social network. Based on the two specific characteristics of the social network (and the email network), the free-scale degree [24] and the small-world degree [25], the clustering coefficient of node i in the email network is calculated by the following formula:

$$C_i = \frac{2 * E_i}{k_i (k_i - 1)} \quad (1)$$

In which, Ci is the clustering coefficient; ki is the number of nodes that link to node i; Ei is the number of edges between neighboring vertices of node i. The authors identify that the higher clustering coefficient of the node is, the lower possibility of the email address corresponding to that node spams, or in other words, it is a normal user. However, calculating the clustering coefficient of the nodes by this formula (1) has some restrictions. Firstly, it ignores all the vertices with k = 1. Secondly, more importantly, the calculation results do not distinguish the nodes which have the same Ei, equals to 0, while having different value of ki (C = 0 when E = 0).

2.2 Method based on PageRank Algorithm

The WWW network is a kind of complex network in which the nodes are webpages and the edges are the links from this webpage to others. Brin and Page, in 1998, proposed the PageRank algorithm [21] used to rank the webpage. The remarkable idea of the algorithm is that a webpage is considered as "important" if there are plenty of "important" webpage links to it. Here is the PageRank formula:

Assume that webpages T1… Tn have links to the webpage A. C(A) is defined as the number of link-outs from the webpage A; then the page rank of webpage A can be calculated by the formula:

$$PR(A) = (1-d) + d\left(\frac{PR(T1)}{C(T1)} + ... + \frac{PR(Tn)}{C(Tn)}\right) \quad (2)$$

In the formula (2), d is the damping which is the probability that a user clicks on a link available on the site. According to Brin and Page's calculations, this damping is set equal to 0.85.

Brin and Page, then, founded Google with the leading search engine (i.e., www.gooogle.com). The success of Google search engine proves the correctness of the PageRank algorithm to rank the vertices of the complex network. Later on, many scholars applied the PageRank algorithm to solve other ranking problems - Hromada used PageRank algorithm to rank the concepts of culture among countries and achieved promising results [26].

2.3 Method based on Weighted PageRank Algorithm

Xing and Ghorbani [22] proposed the Weighted PageRank algorithm in 2004. The ranking score of a webpage (rank) is divided for webpages having link-in(s) from that page with different weights, instead of equally sharing as in the original PageRank algorithm [21].

The weighted PageRank algorithm offers two values $W^{in}_{(v,u)} = \frac{I_u}{\sum_{p \in R(v)} I_p}$ and $W^{out}_{(v,u)} = \frac{O_u}{\sum_{p \in R(v)} O_p}$, in which Iu and Ip are respectively the number of link-in(s) to webpage u and p; Ou and Op are respectively the number of link-out(s) from the webpage u and p. R(v) is the set of webpages with links from webpage v. Here is the formula for the Weighted PageRank:

$$PR(u) = (1-d) + d \sum_{v \in B(u)} PR(v) W^{in}_{(v,u)} W^{out}_{(v,u)} \quad (3)$$

In particular, the damping index d has the same meaning as the formula (4).

Reviews on the author's Web data shows that weighted PageRank algorithm is better than the original PageRank algorithm. However, there is not any review on the application of weighted PageRank algorithm on spam data set. It is very important to use the same dataset to test weighted PageRank algorithm.

3. THEORETICAL FRAMEWORK - New Spam-Filtering Methods Based On Meta-Email Networks

Unlike the two classes spam and ham classifications, the problem is therefore transformed into a multi-class classification problem for the email network. The overall email recommender system produces a ranking list denoted as Rank (E, P), that is defined for the purpose of email prioritisation. It considers both the global ranking and the personalised ranking, where the global rank is calculated by using the Extended Coefficient Clustering [23], and the personalised rank is calculated independently on the individual user's preferences. We adopt the top - k selection as the way to generate the final important email lists for individual users. It is denoted in a general form as follows.

$$Rank\ (E, P) = C1 * G(s) + C2 * P(c, p) \quad (6)$$

where "E" represents "email", "P" stands for "person", "s" is "sender", "c" is the content of the email", and C1 and C2 are two constant weight numbers. G(s) represents the Global rank(s), and P(c, p) represents Personalized rank (c,p).

Rank (E,P) is the rank (or prioritization) of an email E to a person P, this is the rank we would like to compute as describe in the objective). We compute the overall rank by considering two components: Global rank (s) and Personalized rank (c,p) C1 and C2 are adjustment weights.

The global rank evaluates the importance of the emails sender(s) via the calculated ranks, while the personalised rank identifies the content of this email is of the email recipients' interest or not according to the users' actions.

3.1 Coupled Content Classification

The most general form of text data is string, and the most common representation for texts is the vector-space representation. The vector-space model represents the texts for each document as a "bag-of-words". Though the vector-space representation is very simple and efficient, it loses information about the structural information of the words in the document, especially when the text is short.

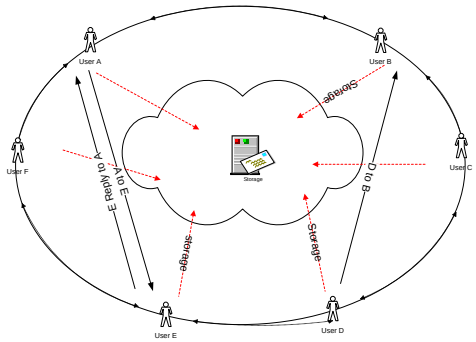

Figure 1. Meta-email network

In many applications, the "unordered bag of words" representation is insufficient for finding the analytical insights, especially in the case for fine-grained applications, where the structure of the documents affects the underlying semantics. Intuitively, the advantage of the vector-space representation is in that the simplicity lends itself to straightforward processing. However, the vector-space representation is inaccurate because it does not include any information about the ordering of the word in the document and assume the words are independent with each other. Actually, in the real word, the document's

semantic meaning is heavily by the coupled relation between words to words. Additionally, the vector-space model implements the word frequency–inverse document frequency (TF-IDF) of each word. It only uses one scalar to represent the feature of one word in the document. The discriminative power of this approach is not strong especially on the low frequency word because many low frequency words share a relative same TF-IDF value. However, most of the documents semantic means are presented by these low frequency words. Therefore, it is very hard to distinguish the similarity among those documents, regardless using Euclidean distance or Cosine distance.

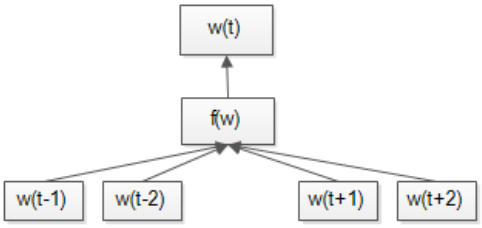

Figure 2. Continuous Bag-of-words Model

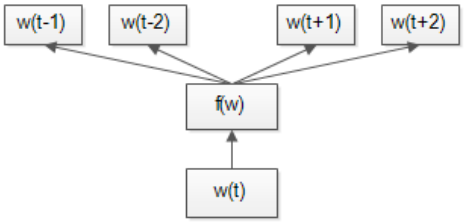

Figure 3. Continuous Skip-gram Model

Both of the two built a vector representation of the words by evolving the coupled relation from its surrounding neighbors. Every word w represents by a vector V =$\{v_1, \cdots, v_n\}$ where n is the window size of the surrounding neighbors. Increasing the range of windows improves the prediction result but costlier. Initially every word start by a random number vectors, and optimizes the value by the learning process. Continuous Bag-of-word model predicts a given word by its past and future neighbors, running a log-linear classifier on the averaged vector to get the resultant word, while the Skip-gram model used the given word as an input to a log-linear classifier to predict surrounding words.

Maximum likelihood learning

Though those methods restricted the range of surrounding context to reduce the computational cost, computing the gradient of log-likelihood related the vocabulary size which is always large. The optimization can be given by:

$$\frac{\partial}{\partial\theta}\log P_\theta(w|h) = \frac{\partial}{\partial\theta}s_\theta(w,h) - \sum_{w'}P(w'|h)\frac{\partial}{\partial\theta}s_\theta(w',h)$$

The computation $s_\theta(w,h)$ require all words in the vocabulary, hence the learning could be very slow.

Recently introduced a noise-contrastive estimation which can perform a more stable and efficient importance sampling for training. Applied this method to build a new model which can train the vector representation of the words by using the formula:

$$\frac{\partial}{\partial\theta}J^{h,w}(\theta) = \frac{kP_n(w)}{P_\theta(w|h)+kP_n(w)}\frac{\partial}{\partial\theta}logP_\theta(w|h) - \sum_{i=1}^{k}[\frac{kP_n(x_i)}{P_\theta(x_i|h)+kP_n(x_i)}\frac{\partial}{\partial\theta}logP_\theta(x_i|h)]$$

where $x_1, \ldots, x_k$ are the k noise samples.

By doing this, the computation of $s_\theta(w,h)$ is unnecessary and reduce the computation cost enormously.

Finally we computed the vector representation of the words and we can obtain the intra relation similarity (CS) between two words straightforwardly:

$$RS^{Intra}(w_\alpha, w_\beta) = \frac{V_\alpha \cdot V_\beta}{||V_\alpha||||V_\beta||}$$

Where V_n is the vector representation for the word w_n.

Document similarity by using Word to word Coupled relations

This work proposes a novel method by considering the coupled relations between words and words. The intra relation between two words is defined by the Cosine distance of two words' vector presentations which only captured the direct relation by the word surrounding neighbors. This work go one step further, because it is possible that two words are not appeared in a same sliding window but still have strong correlations. Hence, we developed a novel metric to capture the indirect relations between those words. We define an inter-coupling relation between words as follows. If two words appeared in the same sliding window with range of n, we define them are direct neighbors (DN) where DN(w_α, w_β)=True. If two words shared at least on direct neighbors, we call them indirect neighbors (IDN).

IDN($w_\alpha, w_\beta | w_\gamma$)= True if DN($w_\alpha, w_\gamma$)=True and DN($w_\beta, w_\gamma$)=True.

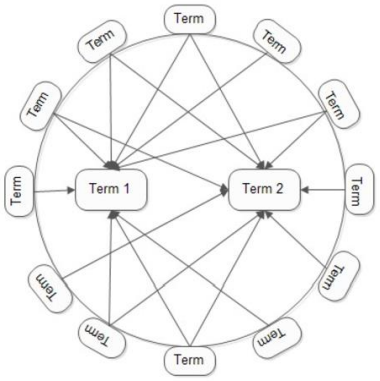

Figure 4. Illustration of the indirect neighbors in texts

The inter relation similarity between words defined as follows:

$$RS^{Inter}(w_\alpha, w_\beta) = \frac{1}{n}\sum_{i=1}^{n} CS^{Intra}(w_\alpha, w_i)\cdot CS^{Intra}(w_i, w_\alpha)$$

Where n is the number of shared neighbors by both w_α and w_β. We assume that the inter relation should be smaller than the inter pass through one shared neighbor, after accumulate all the intra relation through shared neighbors we applied a normalization term to control the final weight.

After the intra relation and inter relation has been defined, we developed comprehensive metric call coupled relation similarity (CRS) defined as follows:

$$CRS(w_\alpha, w_\beta) = \mu \cdot RS^{Intra}(w_\alpha, w_\beta) + (1-\mu)\cdot RS^{Inter}(w_\alpha, w_\beta)$$

Document Similarity

After we have the coupled similarity between two words, we could compute the similarity between two documents by applying the generalized vector space model (GVSM). The original form is:

$$k'(w_\alpha, w_\beta) = V_\alpha W^T \times W V_\beta^T$$

where V_n is the vector representation of w_n and W is the document-word matrix and W^TW reflects the similarity between words which measured by their frequency of co-occurrence across the document set. We adapted the coupled similarity metric to the GVSM by next few steps. As the CRS between all the word pairs in the vocabulary can be computed before the document similarity performed, we can get the coupled relation matrix W_{CRS}. Hence we define the coupled document similarity as:

$$\text{CDS}(w_\alpha, w_\beta) = V_\alpha W_{CRS}^T \times W_{CRS} V_\beta^T$$

4. EXPERIMENT AND EVALUATION

The experiment is based on the Enron email data set. The Enron Corpus is a large database of over 600,000 emails generated by 158 employees of the Enron Corporation and acquired by the Federal Energy Regulatory Commission during its investigation after the company's collapse. We use the proposed model to do the email classification task. When a new email came, based on the proposed model, this experiment predicts the actual mailbox it should belong to. More precisely, this experiment the experiment decides whether the incoming mail belongs to the deleted folder or the inbox folder.

First, this experiment calculates the user preference score UP for each user, based on the aforementioned definition. For the computation efficiency, this experiment sets the maximum neighbors amount for each user to 100. Secondly, for the text content pre-process, this experiment set the minimum term frequency to 10 and ignored the top 50 highest frequent terms when calculating the TFIDF values.

The experiment runs on the entire Enron data set in the first stage, and focuses separately on each user. Due to the limitation of the space, we only select 34 representative users' results.

Figure 5 shows the classification accuracy on the whole data set, comparing the two most widely used classifiers. In all the following figures, we use NB to represent naïve Bayesian classifier, and use SVM to represent the Support Vector Machines classifier, and use CS to represent the Coupled Similarity classifier, and use UP to represent the User preference score. Figure 5 shows the user preference score can significantly enhance the accuracy performance of the classification task by merging it with the classic classification method. Meanwhile, the proposed coupled similarity classifier also outperforms the traditional classification method. Finally, when combined the coupled similarity classifier with the user preference score, it get the best performance of all the experiment tasks.

Figure 6 is the F-measure comparison of the proposed method with the classic method. The result confirmed the advantage of the performance of the proposed method.

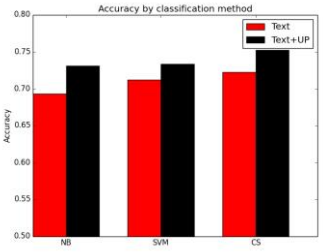

Figure 5. The Accuracy Comparison

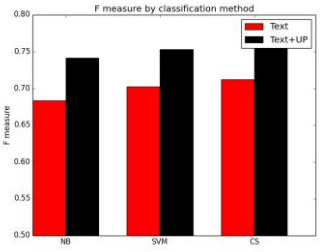

Figure 6. The F-measure Comparison

Figure 7 and 8 are the comparisons of the recall and precession which proved the performance of the proposed method comprehensively.

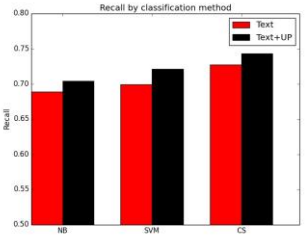

Figure 7. The Recall Comparison

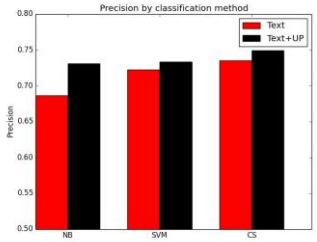

Figure 8. The Precision Comparison

5. ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers to improve the quality of this paper.

6. REFERENCES

[1]Steve, W. (1996). Email overload: exploring personal information management of email. *CHI '96 Proceedings of the SIGCHI conference on Human factors in computing systems: common ground* . 96 (1), p276 - 283

[2]Nicholas, K. (2005). Automated email activity management: an unsupervised learning approach. *IUI '05 Proceedings of the 10th international conference on Intelligent user interfaces* . 05 (1), p67-74.

[3]Anirban, D. (2011). Enhanced email spam filtering through combining similarity graphs. *WSDM '11 Proceedings of the fourth ACM international conference on Web search and data mining* . 11 (1), p785-794.

[4]Khurum, N.J. (2007). Automatic Personalized Spam Filtering through Significant Word Modeling. *ICTAI '07 Proceedings of the 19th IEEE International Conference on Tools with Artificial Intelligence*. 2 (1), p291-298.

[5]Yiming, Y. (2010). Personalized Email Prioritization Based on Content and Social Network Analysis. *IEEE Intelligent Systems*. 25 (4), p12-18.

[6]Paul-Alexandru, C; Jörg, D; Wolfgang, N. (2005). MailRank: using ranking for spam detection. *CIKM '05 Proceedings of the 14th ACM international conference on Information and knowledge management* . 05 (1), p373 - 380.

[7]Mingjun, L; Wanlei, Z . (2005). Spam Filtering based on Preference Ranking. CIT '05 Proceedings of the The Fifth International Conference on Computer and Information Technology . 05 (1), p223-227.

[8]Graham, P., 2002. A plan for spam. Web document, URL: http://www.paulgraham.com/spam.html.

[9]Androutsopoulos I., Koutsias, J., Chandrinos, K.V., Paliouras, G., Spyropoulos, C.D., 2000. An evaluation of Naive Bayesian anti-Spam filtering. *Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning*, Barcelona, Spain, pp 9–17.

[10] Thorsten J, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features" in *Kunstliche Intelligenz-1997*.

[11] Marti A. Hearst (1998). Support Vector Machines. *IEEE Intelligent System*.

[12] Tom Mitchell. (1997). Bayesian Learning. In: Tom, M *Machine Learning*. USA: McGrawhill. p179.

[13] Christopher D. Manning. (2008). *Naive Bayesian text classification.* Available: http://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html. Last accessed 10th April 2012.

[14] Christopher D. Manning. (2008). *Text classification and Naive Bayes.* Available: http://nlp.stanford.edu/IR-book/html/htmledition/text-classification-and-naive-bayes-1.html. Last accessed 13th April 2012.

[15] Ion, A. (2000). An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages. *SIGIR '00 Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval* . 00 (1), p160-167.

[16] Manu, K (2008). *Building Search Applications: Lucene, LingPipe, and Gate*. US: Mustru Publising. p22.

[17] LIBSVM. 2012. *LIBSVM - A Libary for Support Vector Machines*. [ONLINE] Available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm/. [Accessed 10 July 12].

[18] Chih-Wei, H, 2002. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 2/13, 415 - 425.

[19] Boykin P.O. and Roychowdhury V.. "Leveraging social networks to fight spam", IEEE Computer, 38(4):61-68, 2005; "Sorting e-mail friends from foes", Nature news, 19 Feb. 2004.

[20] Hanoi University website. URL: http://www.hanu.vn

[21] Brin S. and Page L. "The Anatomy of a Large-Scale Hypertextual Web Search Engine". Proceedings of the 7th

international conference on World Wide Web (WWW). Brisbane, Australia. 1998, pp. 107–117.
[22] Xing W., Ghorbani A., “Weighted PageRank Algorithm”, Proceedings of the Second Annual Conference on Communication Networks and Services Research, 2004, pp 305 – 314.
[23] Bui N.L., Tran Q.A., Ha Q.T., "User’s authentic rating based on email networks," The First International Conference on Mobile
[24] H. Ebel, L-I. Mielsch and S. Bornholdt (2002). Scale-free topology of email networks, Phys. Rev. E, 66, 035103 (R), Sept. 2002
[25] Newman, M. E. J. and Watts, D. J. (1999). Renormalization group analysis of the small-world network model. Physics Letters A 263, pp. 341–346.
[26] Hromada D., 2010, Quantitative Intercultural comparison by means of parallel page ranking of diverse national wikipedias, Proceedings of JADT (Journées d'analyse des données textuelles) 2010 Computing, Communications and Applications (ICMOCCA 2006), pp. 144-148
[27] Chirita P., Diederich J., Nejdl W., "MailRank: using ranking for spam detection", Proceedings of the 14th ACM international conference on Information and knowledge management, 2005, pp. 373-380
[28] Q. A. Tran, M. T. Vu, F. Jiang, "Email User Ranking Based on Email Networks", American Institute of Physics, Conf. Proc. 1479, pp. 1512 - 1517, doi: 10.1063/1.4756451 ICNAAM 2012, KoS, Greece, Sept., 2012.
[29] M. H. Quang, V. D. Phung, F. Jiang, Q. L. Nguyen, "Image Spam Filtering based on Maximum Entropy Segmentation method", Proceeding of 7th International Conference on Broadband Communications and Biomedical Applications (IB2COM 2012), pp. 147-151, 2012.
[30] M. H. Tuan Vu, Q. A. Tran, F. Jiang, V. Q. Tran, "Multilingual Rules For Spam Detection", in Proceeding of 7th International Conference On Broadband Communications and Biomedical Applications (IB2COM 2012), pp. 106-110, 2012.