Feature-Fusion Framework for Spam Filtering Based on SVM

Qinqing Ren

Department of Control Science and Engineering Zhejiang University

Hangzhou, 310027, China

renqinqing@zju.edu.cn

ABSTRACT

In this paper, we propose a novel framework based on support vector machine (SVM): feature-fusion framework. SVM is proved to be a good method to filter spam on large-scale text collections with state-of-the-art performance by many researchers in the world. Feature extraction is the key step to SVM classification. The feature-fusion the framework based on SVM is a feature level method and has advantages to solve some practical problems of spam filtering and lead to high performance, especially reduces the computational cost. The feature-fusion method proposed in this paper can fuse information of words in an email by using TF-IDF formula and the words in the subject header field of an email by increasing its weight, and improve the performance of spam filtering. Experimental results show that the feature-fusion framework outperforms pure SVM and other SVM-based methods for large-scale spam filtering in terms of ROC curves.

Keywords

Spam filtering, framework, support vector machine (SVM)

1. INTRODUCTION

Email has become one of the most important, efficient and popular communication mechanisms through Internet. At the same time, misuse of email is becoming an increasingly serious problem for both individuals and organizations. One of the examples of such misuse is spam, which is commonly known as unsolicited bulk email (UBE) or unsolicited commercial email (UCE) and has generated a need for robust high-performance anti-spam filters.

Generally, spam filtering is text classification problem, which is an inter-discipline across machine learning (ML) and information retrieval (IR) [14]. The problem of email spam filtering is also a linear separable problem. By volume, spam filtering is easily the most important application of text classification [3]. Hence, using statistical machine learning methods to automatically filter out spam has drawn many researchers' attention. And a large amount of research has already been done in this field, which can be roughly divided into works of feature level and model level. The feature level includes feature selection and weighting [2,9,10], and the model level includes rule-based methods [11], statistical models [5, 12,16], or even ensemble learning methods [13]. Fumera *et al.* [4] have exploited the usefulness of text information embedded into images attached to spam emails. In this paper we will focus on textual information of email's subject header field and its body field while ignoring its address and attachment

information.

In the spam filtering field, most work is based on generative models, and little is done using discriminative models (like support vector machines and logistic regression) [6]. Hulten and Goodman [17] found that discriminative methods typically beat generative methods in large-scale problems. And in the literature of text categorization [14], the discriminative model SVM [19] also outperforms any single model, including generative model (like Naïve Bayes and Gaussian mixture model), or other discriminative models (like logistic regression). Though ensemble learning [13, 14] is usually better than SVM in aspects, it is too time-consuming. Hence, we choose the discriminative SVM model in our approach as the basic classifier.

In this paper, facing the challenging problem of large-scale email spam filtering, we present an efficient method based on SVM to classify the content-based dataset. And then, we propose a new feature fusion framework. These methods are considered in a proposed general framework of feature-fusion for feature extraction and information fusion afterwards.

The organization of this paper is as follows: Section 2 describes the feature-fusion framework and the proposed methods. Section 3 gives experiments on public corpus and the evaluation results. And in Section 4 the conclusion and future work are presented.

2. FEATURE-FUSION SPAM FILTERING FRAMEWORK

Some academic researchers have advocated the use of support vector machines (SVMs) for content-based filtering, as this machine learning methodology gives state-of-the-art performance for text classification [8], especially on non-English datasets.

However the spam filter based on SVM still has many problems in practice, such as high cost, low speed and inflexible model.

Most systems address the English spam filtering problem [1]. The most difficult issue is the system complexity with the increasing number of languages it supports. The charset type of an email can help a lot to elementary classify. Taking Chinese as an example, most non-simple-Chinese emails are spam for common Chinese users. Another important problem is feature extracting. Feature extracting is the key step of spam filtering based on SVM. A better feature extracting method leads to more accurate and quicker speed classification.

Our approach to English and non-English emails adopts information fusion theory to obtain feature extraction, and then classifies the emails by SVM. We optimize the feature extraction procedure to exploit fast implementation by high dimension feature vectors.

2.1 Feature-Fusion Framework for Spam filtering

Figure 1 shows the proposed framework for spam filtering. It includes the following four components:

1) **Tokenization of English and Chinese emails.** When an email arrives, to make the problem simple, we put aside the attachment, pictures and audio. We use different program to tokenize the email. We use Tianwang Chinese tokenization program [1] for Chinese email dataset.

2) **Information fusion of emails.** The words information in email's header field and the email's content word information are fused as the email's information.

3) **Feature extraction.** By putting weighting value of each word in email, the email can be represented in a vector space model (VSM).

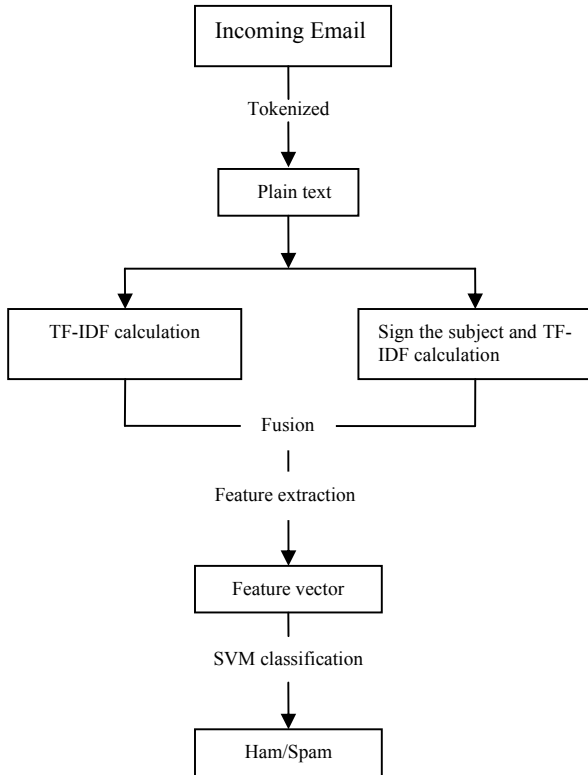4) **SVM classification.** Classify the feature vector using support vector model.

Figure1. Feature-Fusion Framework for Spam Filtering

Although the framework in Figure 1 is designed for English and Chinese target users, it is universal and can be applied efficiently to other language users without much person-effort by different tokenization programs.

2.2 Feature Level Methods

In this section, we describe two different feature schemes in our approach to the spam filtering problem. The baseline considers the common feature weighting scheme, TF-IDF (Term Frequency and Inverse Document Frequency); and then an improved one with extra weighting on subject header field of the email is proposed.

2.2.1 TF-IDF Weighting

Feature level work is usually more important than model work, especially for a sophisticated machine learning method. We use the most widely adopted feature weighting scheme in IR, TF-IDF [15], to represent the email in a vector space model, and it is calculated as follows:

$$a_{ij} = \frac{tf_{ij} \cdot \log\frac{|D|}{DF_i}}{\sqrt{\sum_k \left(tf_{kj} \cdot \log\frac{|D|}{DF_k} \right)^2}} \tag{1}$$

Where tf_{ij} is emails in the training set, and DF_i is the number of emails containing the term i. The importance of a term in an email is measured by the term frequency and its inverse document frequency. The more times an item appears in an email, the more important it is; and the more times it appears in the training email set, the less poorly discriminative it becomes. Often the logarithm of tf_{ij} or DF_i are taken in order to de-emphasize the increases in weighting for larger values [15]. The TF-IDF weighting of each email is calculated in the feature extraction procedure of the proposed framework and we take this weighting scheme as the baseline.

TF-IDF has many advantages such as convenient to use and suitable to other language, such as Chinese. Because the word segmentation of Chinese is more complex than that of English, we need a more efficient method to extract the feature, which is TF-IDF we provide in this paper as a part of the method. However feature extraction method based on TF-IDF does not pay much attention to the semantic of the word. And the TF-IDF is constructed based on the following hypothesis: the word occurs many times in one document will occur little in another document of the same type. This hypothesis does not always hold true, resulting that the accuracy of the filter only using TF-IDF is not high. Therefore, we need other information of the emails to be complementary.

2.2.2 Weighting on Subject header field of Emails

Since TF-IDF weighting does not pay much attention to the semantic of the word. One method to complement the shortcoming is to consider the weighting on subject header field of emails.

The number of words/characters in the subject header field: these features help build a basic profile of the user's writing characteristics. Most virus text is randomly chosen, and spam messages have been found to share certain characteristic. Average number of words/characters per subject, body; average word length: These features capture trends in email wording that can separate normal email from malicious activity, and among users [18].

To make some improvement on the thoughts provided above, we try to emphasize the subject header field of an email by setting heavier weighting to subject header field items than the body field

---

[1] We use the Tianwang Chinese tokenization program, which is available at http://net.pku.edu.cn/~webg/src/ChSeg/.

items. The motivation for this is that the subject header field usually represents an email summarization and contains more semantic information for judgment whether it is a junk email.

2.3 Feature Fusion and Classification

2.3.1 Feature Fusion

While information technology can transform a data poor situation into a data rich environment, the fact remains that data need to be fused and analyzed effectively and efficiently, in order to provide appropriate information for intelligent decision making [20]. Nowadays, information fusion (IF) has been widely used in many fields, such as control, computer science, and aeronautics to obtain optimal estimate. In this paper, information fusion is used to combine with SVM to complete the content-based classification with the idea of fusion first and then filter.

By the fusion of feature information, we can capture specific elements of emails and improve our technique more effectively. The feature information fused in this paper from two sources: the words in body field of email and the words in subject header field of the email. Because the feature information of subject header field of emails can remedy the shortcoming of feature information of only using TF-IDF of email's body field words, it shows better results compared with the results that fuses irrelevant feature information.

2.3.2 Support Vector Marchine

Support vector machine is an algorithm inspired from statistical learning theory. It has been successfully employed in the text classification with its excellent generalization ability through maximum margin approach. As we know, confidence score contains more information than simple binary classification result. Hence, we use SVM first to classify the VSM represented email, and obtain the spam confidence score as the input for further judgment.

SVM can be described as follows: There is training data set T.

$$T = \{(x_1, y_1),.....(x_l, y_l)\} \in (X \times Y)^l,$$

$$x_i \in X = R^n, y_i \in Y = \{1, -1\}, i = 1,....l$$

And each x_i is a vector containing features describing example i,

and each y_i is the class label for that example. In this paper the label of the sample is +1 or -1. Now we import a pair of parameters w and b, here w is a hypothesis vector and b is bias term. Then the SVM problem can be simplified as an optimization problem.

$$\min_{w,b,\xi} \quad \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}\xi_i$$

$$s.t. \quad y_i((w \cdot x_i) + b) \geq 1 - \xi_i$$

$i = 1, 2,.....n, \xi_i \geq 0$, C is a tradeoff parameter. ξ_i is a slack variable. The pair of parameters, w and b, help to generate the predicted class label y_i.

Kernel trick is another important point to the success of SVM, and polynomial kernel, Radial basis function (RBF) kernel and sigmoid kernel are three typical kernels. There is no work ever reported on how to choose one for a specific problem and RBF kernel is usually recommended [19]. In our experiment, we find that there is no substantial difference between linear kernel and RBF kernel. And it makes sense that the text is linearly separable in high dimension space. So we choose the linear kernel for its lower time complexity.

2.3.3 Pseudo-code of the feature extraction

Input: Dataset $U = (X_1, y_1),........,(X_n, y_n)$,
parameter α, β, δ
Initialize $w = 0$; $b = 0$; $y = \{-1, 1\}$;
FOR each $x_i \in U$ **DO**
   change x_i to text file x_i';
   read each word $p_k \in x_i'$;
   IF occurrence of $p_k < \alpha$ **OR** occurrence of $p_k > \beta$
   THEN
      prune p_k;
   END
   FOR each $p_k \in x_i'$ **DO**
     IF p_k is subject of the email
     THEN
       sign p_k;
       calculate p_k's weighting q_k using TF-IDF;
       $P_k = q_k * \delta$
       put q_k to feature vector set $\vec{V}$;
     ELSE
       calculate p_k's weighting q_k using TF-IDF;
       put q_k to feature vector set $\vec{V}$;
     END
   END
END

Figure2: Pseudo-code of the feature extraction

After the feature extraction, do the classification

train and test $\vec{v}$;
FOR each x_i **DO** classify x_i using
$f(x_i) = sign(<w, x_i> + b)$;
END

Figure3: Pseudo-code of the classification

2.3.4 .The Situation to Be Avoid

The logistic average misclassification percentage is defined as:

$$lam\% = \log it^{-1}(\frac{\log it(hm\%) + \log it(sm\%)}{2})$$

where $\log it(x) = \log(\frac{x}{100\% - x})$, *hm%* is the ham misclassification percentage and *sm%* is the spam misclassification percentage [7].

From this formula, we can see if the *hm%* or *sm%* close to 0 or close to 1, the value of *lam%* will be scattered. So when we adjust parameters, we should avoid this happen.

3. EXPERIMENTS

In Section2, a new feature selection for spam filter with SVM is proposed. This feature selection is called feature-fusion framework. In this section, we test this framework on the benchmark data sets to see if this method may outperform other methods which are based on SVM by ROC curve. It can be found that this novel method can also reduce computational cost and save running time. Our main tests on content-based spam detection are performed on large benchmark sets of email data.

3.1 Experiment Setting

We establish several experiments on public datasets: TREC 2005~2007 Spam filtering[2] which has been frequently used by many researchers and has gotten many results by various filter methods[3].

Spam filter is a binary test classification problem, so accuracy of the filter is not good performance method. Sometimes, when the accuracy of filter is very high, problems such as over-fitting happens. Because ROC curve reflects the comprehensive index of variables in aspects such as accuracy, flexibility and specificity, we use ROC curve as standard of comparison. And now ROC curve is becoming a typical method to evaluate spam filtering worldwide.

In our method, we do not care about the attachment, picture and audio of the emails. So if there are attachment, pictures, and audios in the email, they will be pruned before email sets are dealt with. And the email sets will become datasets like the corpus of TREC 2005~2007.

To prove our method proposed in this paper is reliable, the results of our method are compared with ROSVM (Sculley and Wachman 2007) in aspects of area under ROC (or value of $(1 - ROCA)\%$), running time of filtering program, and its memory consumed. We also provide the filter accuracy on each corpus to show our method is practical and good.

First, the tests are conducted on the corpus of TREC 2005~2007 by different parameter setting. Second, we try to find out a set of parameters which can obtain maximal area under ROC curve. At last, we use this set of parameters to do subsequent experiments and obtain the other results besides area under ROC curve.

Our classification code is developed with Java and the cross-validation part is completed in Python. All the experiments are all done on a typical PC with Intel Core2 Duo E8500 3.16GHz and 2GB memory , which runs Microsoft Windows XP with service

[2] We take the public corpus of the Anti-Spam Track in. http://plg.uwaterloo.ca/~gvcormac/spam/

[3] We can use the tools by download them in the site of : http://www.csie.ntu.edu.tw/~cjlin/libsvm/, http://www.python.org/ and ftp://ftp.gnuplot.info/pub/gnuplot/

pack2 as operation system and eclipse-3.2.1 with JDK v1.6.0_17. In this Windows XP system, there is VMware station on which installed Fedora 10. The running time and the memory consumption are obtained by jprofiler 5.1.4. Besides, we use python 2.6.1[3], libsvm 2.89[3] and gunplot1.32[3] to do cross-validation to obtain the optimal parameters of SVM, i.e. c and gammar. (Libsvm is a tool provide by Chih-Chung Chang and Chih-Jen Lin from Taiwan University, which can be used for text classification research.) The procedure is simulated by canonical order on each corpus and the feedback is made immediately after the mails of corpus are classified. This allows meaningful comparison between our method and the methods provided before on these content-based spam detection tasks.

To sum up, the whole experiment procedure includes four parts as follows:
1) Conduct cross-validation experiments to obtain optimal parameters to train.
2) Run main program to obtain feature vectors.
3) Train and test the feature vectors, then classify the emails.
4) Draw ROC curve on Fedora.

3.2 Parameter Adjustment

There is a set of parameters in this filter system. They are discussed as follows:

The optimal parameters of SVM are as follows: c is the parameter of cost for C-SVC and γ is the parameter of kennel function setting. The optimal values of *c* and γ are obtained by cross-validation using python and LIBSVM, which have been mentioned in the above 3.1. The optimal value of *c* is 31.73, and the optimal value of γ is 0.0078. We use these parameters to train the train set. When we fuse the information feature of emails, we use TF-IDF formula mentioned in 2.2.1. First, we set two parameters α, β, which are used to filter those words of exceptional occurrence: low occurrence and high occurrence. By trying the parameters many times and checking the area under the ROC curve, we determine the value of α, and β as α=3, and β=40 respectively. This means if the times of a word in email occurs lower than 3, or more frequent than 40, it will be ignored. When we increase the weight of subject header field information of an email, we set weight multiple on words in email's subject header field. The parameter of multiple is δ. Under the prerequisites of α=3, β=40, after many experiments, δ is determined as 5. When δ=5, (α=3, β=40), the area under ROC curve is maximal.

Table1. The optimal and empirical parameters of feature-fusion framework for spam filtering

Optimal parameters		Empirical parameters		
c	γ	α	β	δ
31.73	0.0078	3	40	5

3.3 ROC Curves

Now we chose TREC05p-1[2], TREC06p[2] and TREC07p[2] for further experiment. In TREC05p-1 corpus, there are 92,189 emails in which 52,790 emails are labeled spam while 39,399 emails are labeled ham. The corpus of TREC06p contains 37,822 messages in which 12,910 messages are ham and 24,912 messages are spam. The corpus of TREC07p contains 75,419 messages in which 25,220 massages are ham and 20,199 are spam.

According to the ROC curve evaluation, if the area under ROC curve (ROCA) of feature-fusion framework spam filter is larger, then the value of $(1-ROCA)\%$ is less. So if we want to compare the performance, we can also compare the value of $(1-ROCA)\%$.

ROC curve has been considered as the accurate evaluation method on text classification in the world as we mentioned in Experiment Setting. The ROC curves of the feature-fusion framework spam filter on the three corpuses are shown in Figure 4.
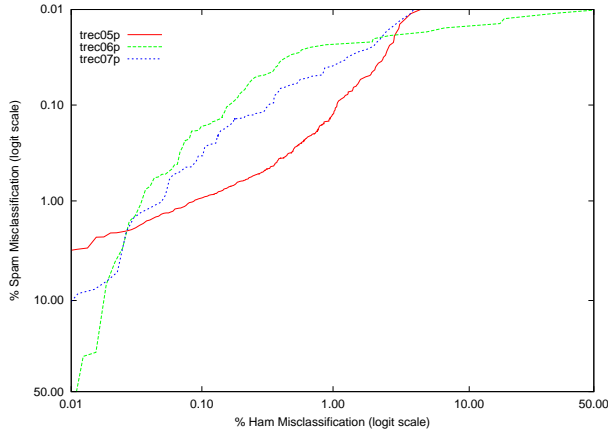

Figure 4: ROC Curves of feature-fusion framework based on SVM

3.4 Comparison with ROSVM

In 2007, Sculley and Watchman presented a relaxed online SVM (ROSVM) method for spam filtering *in Relaxed On-line SVMs for Spam Filtering*. Both the method of Sculley and Watchman and ours are based on SVM, so we can compare these two methods in terms of ROC curves or values of $(1-ROCA)\%$, where 0 is optimal.

Table2. Results compared between feature-fusion framework (FFF) spam filter and ROSVM filter.

Filters	Corpus	Time (h-m-s)	Memory (MB)	(1-ROCA)%
FFF Spam filter	TREC05p-1	39m27s	86.79	0.0088
	TREC06P	6m46s	97.20	0.0228
	TREC07p	9m9s	99.80	0.0092
ROSVM Filter	TREC05p-1	6h52m	--	0.0090
	TREC06P	5h9m	--	0.0240
	TREC07p	--	--	0.0093

We can also compare these two filters in other aspects, such as running time (the total time from input the TREC05p-1 to the classification results), memory cost on TREC05p-1, TREC06p and TREC07p.

In Table 2, for feature-fusion framework, the running time is the main program running time, the train, test and classifying time by libsvm[3] are much less, only about a few second, which can be ignored. The main program includes the procedure from inputting emails to outputting the feature vectors.

The time of cross validation is 20 minutes for TREC05p-1, 9minutes and 11minutes for TREC06p and TREC07p successively.

Also in Table 2, for feature-fusion framework, the memory refers to the main program memory. The memory of other procedure is much less than main program memory, which can be ignored.

From Table 2, it is obviously, our feature-fusion frame spam filter is much quicker and does not use more memory. Since the time of ROSVM reported is CPU time, the actually running time of ROSVM is even more. Furthermore, the $(1-ROCA)\%$ values of our experiments are better than those of ROSVM. The above results show that the performance of our method is better. We will do experiments on TREC06C[2] which is an email set in Chinese and add smoothing factor to make the ROC curve more smoothing in our later work.

3.5 Accuracy of Feature-Fusion Framework Spam Filter

We do experiments on TREC05p-1, TREC06p and TREC07p by feature-fusion framework based on SVM. And the accuracy is 98.8307%, 99.6414% and 99.6327% respectively. From the values of accuracy, it can be seen that results are steadily and not having much fluctuation.

4. CONCLUSIONS AND FUTURE WORK

SVM is believed to give state-of-the-art performance for text classification [8] by some researchers. But compared with other spam filters based on naïve Bayes and other filters, its computational cost is much higher and the speed is slow. Nowadays, some researchers are working on online SVM to reduce the computational cost and increase the speed of classification without decreasing its accuracy. However, the train model of online SVM will change when it is classifying the emails and it cannot deal with the situation that margin support set is empty. Besides, in practice, there are also other problems, such as non-English email set and various kinds of information in emails. These problems are ubiquitous for all classifiers. To solve the above problems, in this paper a feature-fusion framework based on SVM is presented for filtering spam, which is in the content-based email sets.

The experimental results show that a judicious selection of features and suitable feature extraction method can significantly improve the performance of classification techniques. Although some feature extraction methods have been proposed before, such as term occurrence, term frequency, and binary occurrence etc, the feature extraction based on the concept of information fusion is first proposed in this paper. This feature extraction method is to fuse the information of words in email's body field and words in email's subject field, using TF-IDF to calculate their weights. Experiments demonstrate that this framework of feature extraction method increases the effectiveness of text detection and creating an overall high performance. Furthermore, this framework based on SVM leads to less computational cost. Though the experiments in this paper are conducted on English datasets, the framework can also be used for email sets in other languages, such as Chinese, Japanese etc.

In the future, we will continue to find ways to deal with the attachment, pictures etc, as a complement feature, not only put them aside at the beginning. We will also consider some special

contents of email's subject header field which may give more information of emails and expect to further improve the framework by fuse more features of useful information and help to overcome the shortcomings of only using TF-IDF or using a changing model for online SVM.

5. ACKNOWLEDGMENTS

6. REFERENCES

[1] Lynam, T. R., and Cormack, G. V. On-line Spam Filter Fusion. In *Proceedings of 29th Annual International ACM SIGIR Conference on Research and Development on Information Retrieval (SIGIR 2006)*, Seattle, Washington, August 6-11, 2006

[2] Yeh, C.Y., Wu, C. H., and Doong, S. H. *Effective Spam Classification based on Meta-Heuristics. In Proceedings of IEEE International Conference on Systems, Man and Cybernetics (SMC 2005)*, Vol. 4, 3872-3877, Waikoloa, Hawaii, October 10-12, 2005,

[3] Lewis, D. D. Invited talk: (Naive) Bayesian Text Classification for Spam Filtering. Minimizing Market MAYhem: Statistical Applications in Direct Marketing, *The Spring Conference of the Chicago Chapter of the American Statistical Association*, Loyola University. Chicago, IL USA. May 7, 2004

[4] Fumera, G., Pillai, I., and Roli, F. Spam Filtering Based On The Analysis Of Text Information Embedded Into Images. *Journal of Machine Learning Research (JMLR)*, *7*, (2006), 2699-2720

[5] Bratko, A., Filipic, B., Cormack, G.V., Lynam, T.R., and Zupan, B. Spam Filtering Using Statistical Data Compression Models. *Journal of Machine Learning Research (JMLR)*, *7*, (2006), 2673-2698

[6] Goodman, J., and Yih, W. T. Online Discriminative Spam Filter Training. In *Proceedings of 3rd Conference on email and Anti-Spam (CEAS 2006)*, Mountain View, CA, 2006.

[7] Gordon Cormack. TREC 2007 Spam Track Overview. In *Proceeding of the TREC 2007,* (Gaithersburg, Maryland, USA, November 5-9, 2007). National Institute of Standards and Technology (NIST) Special Publication, 500-274, 2007

[8] D.Sculley and Grbriel M.Wachman. Relaxed Online SVMs for Spam Filtering. In *Proceeding of the SIGIR conference* (Amsterdam, The Netherlands, July 23-27, 2007). ACM Press, New York, NY, 2007

[9] Sahami，M., S. Dumais, S., Heckerman, D., and Horvitz, E. A Bayesian Approach to Filtering Junk E-mail. In *Proceedings of the AAAI workshop on learning for Text* Categorization, 1998.

[10] Kiritchenko, S., Matwin, S., and Abu-Hakima, S. email Classification with Temporal Features. In *Proceedings of Intelligent Information Systems (IIS 2004)*, 523-533, 2004.

[11] Cohen, W. Fast Effective Rule Induction. In *Proceedings of 12th International Conference of Machine Learning (ICML 1995)*, 115-123, Lake Tahoe, CA, 1995.

[12] Drucker, H., Wu, D., and Vapnik, V.N. Support Vector Machine for Spam Categorization. *IEEE Trans. on Neural Networks*, *10*, (1999), 1048-1054

[13] Carreras, X., and Marquez, L. Boosting Trees for Anti-Spam email Filtering. In *Proceedings of European Conference on Recent Advances in NLP (RANLP 2001)*, 58-64, Sep. 2001.

[14] Sebastiani, F. Machine Learning in Automated Text Categorization. *ACM Computing Surveys, 34*, 1 (2002), 1-47

[15] Lang, K. NewsWeeder: learning to filter news. In *Proceedings of the 12th International Conference on Machine Learning (ICML 1995)*, 331-339, Lake Tahoe CA, 1995.

[16] Androutsopoulos, I., Koutsias, J., Chandrinos, K.V., Paliouras G., and Spyropoulos, C. D. An Evaluation of Naive Bayesian Anti-Spam Filtering. In *Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000)*, 9-17, 2000,

[17] Hulten, G., and Goodman, J. Tutorial on junk e-mail filtering. In *Proceedings of 21st International Conference of Machine Learning (ICML 2004)*, Banff, Alberta, Canada, July 4-8, 2004.

[18] Steve Martin, Anil Sevani, Blaine Nelson, Karl Chen,and Anthony D. Joseph Analyzing Behaviorial Features for Email Classificaiton. In Proceedings of 3rd Conference on email and Anti-Spam (CEAS 2005), Stanford University, CA, 2005.

[19] Scholkopf, B., and Smola, A. J. Learning with kernels. MIT Press, Cambridge, MA, 2002.

[20] Tien, James M. Toward a Decision Informatics Paradigm: A Real-Time, Information-Based Approach to Decision Making. *Journal of Systems, Man, and Cybernetics –Part C: Application and Reviews. 33*, 1, (2003), 102-113.