Proceedings of the Fifth International Conference on Machine Learning and Cybernetics, Dalian, 13-16 August 2006

AN EFFICIENT SVM-BASED SPAM FILTERING ALGORITHM

ZI-QIANG WANG, XIA SUN, XIN LI, DE-XIAN ZHANG

School of Information Science and Engineering, Henan University of Technology, ZhengZhou 450052, China
E-MAIL: wzqagent@xinhuanet.com, wzqagent@126.com

Abstract:

The electronic mail (e-mail) concept makes it possible to communicate with many people in an easy and cheap way. Though email brought us such huge convenience, it also caused us trouble of managing the large quantities of spam mails received everyday. Without appropriate counter-measures, the situation seems to be worsening and spare email will eventually undermine the usability of email. To efficiently solve the above problems, a spam mail filtering method using feature selection and support vector machine is proposed in this paper. The experimental results show that the proposed method outperforms other conventional spam filtering method.

Keywords:

SVM; Spam filtering; Feature selection; Text classification

1. Introduction

With the popularization of the Internet, low cost, and fast delivery of message, email has become an indispensable method for people to communicate each other. Though email brought us such huge convenience, it also caused us trouble of managing the large quantities of spam mails received everyday. Spam mails, which are unsolicited commercial emails or junk mails, flood mailboxes, exposing minors to unsuitable content, and wasting network bandwidth. A study was reported that spam messages constituted approximately 10% of the incoming messages to a corporate network. Without appropriate counter-measures, the situation seems to be worsening and spare email will eventually undermine the usability of email[1] . Therefore it is challenging to develop spam filters that can effectively eliminate the increasing volumes of unwanted mails automatically before they enter a user's mailbox.

To efficiently solve the above problems, spam mail filtering system has been developed for recent years, and there have been lots of studies to increase the performance of the system. For example, rule-based system[2] is suggested to classify spam mails, but this system can be strongly dependent on the existence of key terms, therefore, specific terms can cause the failure of filtering. In addition, Naïve Bayesian classifier is traditionally very popular method for document classification and mail filtering system [3]. It uses probabilistic method to compute test document's possible categories, and achieves high performances of precision and recall. But it exists the following drawbacks. First, it has a cold start problem; second, the cost of spam mail filtering is higher than rule-based systems; third, if an e-mail has only few terms those represent its contents, the filtering performance is fallen. To alleviate those problems, many researchers suggested new spam classification systems which use other methods such as Bayesian network that enhancing the performance of Bayesian classification [4], Weighted Bayesian Classification (WBC)[5], Neural Networks[6] and support vector machine[7], etc.

In its simple form, spam filtering can be recast as text categorization task where the classes to be predicted are spam and legitimate. The success of statistical learning techniques in text categorization[8] has recently led researchers to explore the applicability of statistical learning algorithms in anti-spam filtering. Especially, support vector machine (SVM) is a powerful supervised learning paradigm based on the structured risk minimization principle from computational learning theory[9]. SVM has been reported remarkable performance on text categorization task with many relevant features[10]. It has also been applied to spam filtering task with better filtering accuracy [11]. To further improve spam classification performance, we proposed a spam filtering method via feature selection based on SVM. The rest of this paper is outlined as follows: We begin with a E_mail data pre-processing and feature selection, followed by a discussion on spam filtering method using SVM classifier and performance comparison.

1-4244-0060-0/06/$20.00©2006 IEEE

2. E-mail data pre-processing and feature selection

2.1. Definition of spam filtering

Spam filtering based on the textual content of email messages can be seen as a special case of text categorization(TC), with the categories being spam and legitimate (non-spam). Although the task of text categorization has been researched extensively, its particular application to email data and, especially, detection of spam is relatively recent. Following the description used in Ref. [8], we now give a definition of automated spam filtering.

Definition 1. Spam Filtering. Given message set $D = \{d_1, d_2, \cdots, d_{|D|}\}$ and category set $D = \{c_1 = spam, c_2 = legitimate\}$, where d_1 is the *j*th mail in *D* and *C* is the possible label set. The task of automated spam filtering is to build a Boolean categorization function $\psi(d_j, c_i): D \times C \to \{True, false\}$. When $\psi(d_j, c_i)$ is True, it indicates message d_j belongs to category c_i; when $\psi(d_j, c_i)$ is False, it means d_j does not belong to c_i.

In the setting of spam filtering there exist only two category labels: spam and legitimate. Each message $d_j \in D$ can only be assigned to one of them, but not both. Therefore, we can use a simplified categorization function $\psi_{spam}(d_j): D \to \{True, false\}$ instead. A message is classified as spam when $\psi_{spam}(d_j)$ is True, and legitimate otherwise.

2.2. Data pre-processing and feature selection

Vector space model is a text representing approach, which is widely used and has good performance in TC. In its simple form, spam filtering can be recast as text categorization task where the classes to be predicted are spam and legitimate. Therefore, Email can be regarded as a vector space, which is composed of a group of orthogonal key words.

For each email, its textual portion was represented by a concatenation of the subject line and the body of the message. Due to the prevalence of html and binary attachments in modern email a degree of pre-processing is required on messages to allow effective feature extraction. Therefore, we adopt the following data pre-processing steps: 1)To avoid treating forms of the same word as different attributes, a lemmatizer was applied to the corpora to convert each word to its base form (e.g., "got" becomes "get"). 2) The stopping process is adopted to remove the high frequent words with low content discriminating power in a email document such as "to", "a","and","it", etc. Removing these words will save spaces for storing document contents and reduce time taken during the subsequent processes.3) Words were defined as contiguous strings of characters delimited by whitespace. All characters were converted to lowercase, but if a word consisted of all capital letters, it was effectively treated as two words: all lowercase all uppercase. Zipf's law was applied to eliminate word features whose frequency in either class was less than 3.

All the algorithms applied in TC(including spam filtering) need the documents to be represented in a way suitable for the inducing of the classifier. In our experiment, each message is converted into a vector $x = < x_1, x_2, \cdots, x_n >$, where $x_1, x_2, \cdots, x_n$ are the values of attributes $X_1, X_2, \cdots, X_n$,as in the vector space model. All attributes are binary: $X_i = 1$ if some characteristic represented by X_i is present in the message; otherwise $X_i = 0$. In this paper, attributes represent words, i.e., each attribute shows if a particular word (e.g., "people") occurs in the message.

To reduce the high dimensionality of the instance space, feature selection was performed. First, words occurring in less than 4 messages were discarded, i.e., they were not considered as candidate attributes. Then, the information gain (IG) of each candidate attribute X with respect to variable C, denoting the category, was computed as the following equation (1), and the attributes with the m highest IG-scores were selected, with m varying in our experiments from 50 to 700 by 50. The probabilities were estimated from the training corpora using m-estimates[12].

$$IG(X,C) = \sum_{x \in \{0,1\}, c \in \{spam, legit\}} P(X = x, C = c) \cdot \log_2 \frac{P(X = x, C = c)}{P(X = x) \cdot P(C = c)} \quad (1)$$

3. Spam filtering algorithm based on SVM

The textual and non-textual features representing a email, obtained through the method mentioned previously, are as the input to the spam email filtering algorithm. In the approach, the filtering algorithm is represent by support vector machine(SVM).

Support vector machine (SVM) is a powerful supervised learning paradigm based on the structured risk

minimization principle from statistical learning theory, which is currently placed among of the best-performing classifiers and have a unique ability to handle extremely large feature spaces (such as text), precisely the area where most of the traditional techniques fail due to the "curse of the dimensionality". SVM has been reported remarkable performance on text categorization task. In our evaluation, we used the Library for Support Vector Machines (LIBSVM, version 2.33)[13] to build SVM models. In the following, we give a brief introduction to the theory and implementation of SVM classification algorithm.

Consider the problem of separating the set of training set vectors belonging to two separate classes in some feature space. Given one set of training example vectors:

$$(x_1, y_1), \cdots, (x_l, y_l),\ x_i \in R^n,\ y_i \in \{-1, +1\} \tag{2}$$

we try to separate the vectors with a hyperplane

$$(w \cdot x) + b = 1 \tag{3}$$

so that

$$y_i[(w \cdot x) + b] \geq 1 \quad , (i = 1,2,\cdots,l) \tag{4}$$

The hyperplane with the largest margin is known as the optimal separating hyperplane. It separates all vectors without error and the distance between the closest vectors to the hyperplane is maximal. The distance is given by

$$d(w,b) = \frac{2}{\|w\|} \tag{5}$$

Hence the hyperplane that separates the data optimally is the one that minimizes the following equation:

$$Minimize\ \frac{1}{2}\|w\|^2 \tag{6}$$

subject to the constraints of (4).

To solve above problem, introduce lagrange multipliers $\alpha_i, i = 1,2,\cdots,l$ and define

$$w(\alpha_i) = \sum_{i=1}^{l} \alpha_i y_i x_i \tag{7}$$

With Wolfe theory the problem can be transformed to its dual problem:

$$\max W(\alpha) = \sum_i \alpha_i - \frac{1}{2} w(\alpha) \cdot w(\alpha),\ s.t.\ \ \alpha_i \geq 0 \tag{8}$$

$$\sum_i \alpha_i y_i = 0 \tag{9}$$

With the optimal separating hyperplane $(w_0 \cdot x) + b_0 = 0$ found, the decision function can be written as:

$$f(x) = (w_0 \cdot x) + b_0 \tag{10}$$

Then the test data can be labeled with

$$label(x) = \mathrm{sgn}(f(x)) = \mathrm{sgn}((w_0 \cdot x) + b_0) \tag{11}$$

Training vectors that satisfy $y_i[(w_0 \cdot x) + b_0] = 1$ are termed support vectors, which are always corresponding to nonzero α_i. The region between the hyperplane through the support vectors on each side is called the margin band.

In the case of linearly non-separable training data, by introducing slack variables the primal problem can be rewritten as:

$$Min\left(\frac{1}{2}\|w\|^2 + C\sum_i \xi_i\right) \tag{12}$$

subject to $y_i[(w \cdot x) + b] \geq 1 - \xi_i,\ \xi_i \geq 0$.

Similarly, we can get the corresponding dual problem

$$\begin{aligned} &\max W(\alpha) = \sum_i \alpha_i - \frac{1}{2} w(\alpha) \cdot w(\alpha), \\ &s.t.\ \ 0 \leq \alpha_i \geq C\ ,\ \sum_i \alpha_i y_i = 0 \end{aligned} \tag{13}$$

Problems described as in Equation(8) and Equation(13) are typical quadratic optimization questions, and have been approached using a variety of computational techniques. Recent advances in optimization methods have made support vector learning in large-scale training data possible[14].

All the training vectors corresponding to nonzero α_i are called support vectors, which form the boundaries of the classes. The maximal margin classifier can be generalized to nonlinearly separable data via transforming input vectors into a higher dimensional feature space by a map function φ, followed by a linear separation there. The expensive computation of inner products can be reduced significantly by using a suitable kernel function $K(x_i, x_j) = (\varphi(x_i), \varphi(x_j))$. We implemented the SVM classifier using the LIBSVM library[13] and adopted radial basis function (RBF) defined as the kernel $K(x_i, x_j) = \exp\left(-\gamma \|x_i - x_j\|^2\right)$.

A practical difficulty of using SVM is the selection of parameters, i.e., the soft margin parameter C and kernel parameter γ in our case. Although SVM is not sensitive to different choices of parameter γ RBF kernel [15], it is desirable to obtain optimal values for both of these parameters given a special dataset. Automatic model selection for SVM has been studied extensively in the field of machine learning. Several upper bounds of the generalization errors were defined [16], and efficient

algorithms were designed to search the best values for these parameters. Accuracy in this study is satisfactory using $C = 300$ and $\gamma = 0.5$, the optimal values computed by leave-one-out model selection (LOOMS) algorithm [17].

4. Performance measure

To evaluate the performance of the proposed system, we now introduce the performance measures used in this paper. Let *S* and *L* stand for spam and legitimate message, respectively. $n_{L\to L}$ and $n_{S\to S}$ denote the numbers of legitimate and spam messages correctly classified by the system. $n_{L\to S}$ represents the number of legitimate messages misclassified as spam (false positive), and $n_{S\to L}$ is the number of spam messages wrongly treated as legitimate (false negative). Then spam precision(P), spam recall(R), and F_1-measure are defined as follows:

$$\Pr ecision = \frac{n_{S\to S}}{n_{S\to S} + n_{L\to S}} \tag{14}$$

$$\mathrm{Re}\, call = \frac{n_{S\to S}}{n_{S\to S} + n_{S\to L}} \tag{15}$$

$$F_1 = \frac{2 \times \Pr ecision \times \mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \tag{16}$$

Apart from the above commonly used evaluation measures, cost must also be taken into account when defining evaluation measures. The total cost ratio (TCR) [3] allows the performance of a filter to be compared easily to that of the baseline:

$$TCR = \frac{n_S}{\lambda \cdot n_{L\to S} + n_{S\to L}} \tag{17}$$

Here greater TCR values indicate better performance. If TCR is less than 1.0, then the baseline (not using the filter) is better. An effective spam filter should be able to achieve a TCR value higher than 1.0 in order to be useful in real-world applications.

5. Experiment comparison

Unlike general text categorization task where many standard benchmark collections exist, relatively few spam corpora are available, and the sizes are often small. This is probably because while it is easy to collect spam messages, it is much harder to collect legitimate mails for the reason of protecting personal privacy. In this experiment, we used lingspam mail set which introduced by Androutsopoulos [18]. Lingspam includes: 2412 legitimate messages from a linguistic mailing list and 481 spam messages collected by the author with a 16.63% spam rate, which has been used in a considerable number of publications. This mail set contains 4 types of mails-'bare', 'stop-list', 'lemmatizer', 'lemmatizer+stop-list'. Each type of mails is created by pre-processing procedure. And in this experiment, we used the last type of mails, lemmatized and stop-list deleted mails for training and testing the system.

Table 1. Performance comparison on lingspam

Method	Precision	Recall	F1	TCR
NB	89.21%	77.58%	82.94%	2.81
NN	91.43%	78.62%	84.58%	3.47
SVM	94.75%	89.37%	91.46%	5.42

To test the performance of the proposed method, we compared the proposed SVM method with naive Bayes-based(NB)[4] and neural network-based(NN)[6] approaches in terms of precision, recall, F_1 and TCR measure. The proposed method is implemented using Delphi, and MS-SQL Server. All the experiments are performed on a 2.2GHz Pentium PC with 512M RAM, running Microsoft Window XP. Table 1 lists the performance comparison between our proposed method with naive Bayes-based and neural network-based approaches on the lingspam corpora. The experimental results show that the proposed method outperforms other conventional spam filtering method.

6. Conclusions

The electronic mail (e-mail) concept makes it possible to communicate with many people in an easy and cheap way. But, many spam mails are received by users without their desire. As time goes on, a higher percentage of the e-mails is treated as spam. To efficiently solve the above problems, we proposed a spam mail filtering method with feature selection and support vector machine(SVM).The experimental results show that the proposed method outperforms other conventional spam filtering method.

Acknowledgements

This paper is supported by the National Natural Science Foundation of China under Grant No.90412013-3, the Natural Science Foundation of Henan Province under Grant No.0511011700, and the Natural Science Foundation of Henan Province Education Department under Grant No.200510463007.

References

[1] L.F. Cranor, and B.A. LaMacchia, Spam, Communications of ACM, Vol.41, No.8, pp. 74-83, August 1998.

[2] W.W.Cohen, Learning rules that classify E-mail, Proceedings of 1996 AAAI Spring Symposium on Machine Learning in Information Access, Palo Alto, pp.18-25, April 1996.

[3] I. Androutsopoulos, J. Koutsias, and K. V. Chandrinos, Proceedings of 2000 Workshop on Machine Learning in the New Information Age, Barcelona, pp.9-17, May 2005.

[4] M.Sahami, S.Dumais, D.Heckerman and E.Horvitz, A Bayesian approach to filtering junk E-Mail, Proceedings of the Fifteenth National Conference on Artificial Intelligence, Madison, pp.55-62, July 1998.

[5] T.Gartner, and P.A.Flach, WBCsvm: weighted bayesian classification based on support vector machine, Proceedings of the 18th International Conference on Machine Learning, Williamstown, pp.154-161, June 2001.

[6] J.Clark, I.Koprinska and J.Poon, A neural network based approach to automated e-mail classification, Proceedings of the 2003 IEEE/WIC international conference on web intelligence, Halifax, pp. 702-705, October 2003.

[7] A. Kolcz, and J. Alspector, SVM-based filtering of e-mail spam with content-specific misclassification costs, Proceedings of ICDM-2001 Workshop on Text Mining, San Jose, pp.53-58,November 2001.

[8] F.Sebastiani, Machine learning in automated text categorization, ACM Computing Surveys, Vol.34, No.1, pp.1-47, March 2002.

[9] V.N.Vapnik,The Nature of Statistical Learning Theory, Springer-Verlag, New York,1995.

[10] T.Joachims,Text categorization with support vector machines: learning with many relevant features, Proceedings of the 10th European Conference on Machine Learning,Chemnitz,pp.137-142,April 1998.

[11] H.Drucker, D.Wu and V.N.Vapnik, Support vector machines for spam categorization, IEEE Transactions on Neural networks,Vol.10,No.5,pp.1048-1054, May 1999.

[12] T.M.Mitchell, Machine Learning, McGraw-Hill, New York, 1997.

[13] C.C.Chang, and C.J.Lin, LIBSVM-A library for support vector machines (version 2.33), http: // www. csie.ntu.edu.tw/~cjlin/libsvm/index.html,2002.

[14] J.Platt,Sequential minimal optimization: a fast algorithm for training support vector machines, Technical Report 98-14, Microsoft Research, Redmond, Washington, pp.1-21,April 1998.

[15] B.Schoelkopf, K. Sung,C.Burges,F.Girosi, P.Niyogi, T. Poggio, and V.Vapnik, Comparing support vector machines with Gaussian kernels to radial basis function classifiers, IEEE Transactions on Signal Processing, Vol.45, No.11,pp.2758-2765, November 1997.

[16] V.Vapnik, and O.Chapelle, Bounds on error expectation for support vector machines, Neural Computation, Vol.12,No.9, pp.2013-2036, September 2000.

[17] O.Chapelle, V.Vapnik , O.Bousquet and S.Mukherjee, Choosing multiple parameters for support vector machines, Machine Learning, Vol.46, No.1, pp. 131-159 , January 2002.

[18] G.Sakkis, I. Androutsopoulos, and G.Paliouras, A memory-based approach to anti-spam filtering for mailing lists, Information Retrieval, Vol.6, No.1, pp. 49-73, January 2003.