A LVQ-based neural network anti-spam email approach

Zhan Chuan Lu Xianliang Hou Mengshu Zhou Xu
College of Computer Science and Engineering of UEST of China, Chengdu, China 610054
zhanchuan@uestc.edu.cn xlu@uestc.edu.cn

Abstract: Along with wide application of e-mail nowadays, many spam e-mails flood into people's email inboxes and bring catastrophe to their study and work. This paper presents a novel anti-spam e-mail filter based-LVQ network in terms of spam e-mails which are mainly made up of several kinds commercial or political spam emails at present. Our experiment has proved that the filter based on LVQ is superior to Bayes-based and BP-based approaches in total performances apparently.
Keyword: LVQ, anti-spam e-mail filtering, mutual information, vector space model

1. Introduction

Along with wide application of the Internet, e-mail has been used widely with its characteristics of high-speed, convenient, low cost and become an efficient and popular communication medium nowadays. However, a large number of spam e-mails flood into people's mailboxes and bring catastrophes into their study and life. Spam e-mail is annoying to most users, as they waste users time, money, network bandwidth as well as, meanwhile, clutter users' mailboxes, even be harmful, e.g. pornographic content. It was reported an American received 2200 pieces spam e-mail on average in 2002. Increasing by 2% per month, it will reach 3600 pieces spam e-mails in 2007. A survey [1] by CNNIC found that every email user in china received 13.7 piece emails per week in 2004, including 7.9 piece spam emails. In America, spam emails make enterprises to be loss up to 9 billions per year [2]. A study was reported that spam messages constituted approximately 60% of the incoming messages to a corporate network. Without appropriate counter-measures, the situation will continue worsening and spam email will eventually undermine the usability of email.

In terms of content-based anti-spam email filtering, emails are usually regarded as particular texts. Cohen [3] used RIPPER algorithm to classify emails. Sahami et al. [4] used Bayes theorem to filter spam emails. It was proved that Bayes-based filtering approach outperforms keywords-based approach in performance. Xavier Carreras et al. [5] used Boosting algorithm to filter spam emails. After testing public email sample PU1 corpus, they found that their approach outperforms Bayes-based and decision tree-based methods. Duhong Chen et al. [6] compared four algorithms, Bayes, decision tree, neural networks, Boosting, and drew a conclusion that neural network algorithm has higher performance. James clark et al. [7] designed a 3 layers BP neural networks. It was shown that a BP network with IG has rather good effect of identifying spam email in their experiment.

This paper uses a LVQ network, which combines subclasses into a single class and forms complex class boundaries, to design an anti-spam email neural network model and identify spam emails which are mainly composed of commercial and political emails. Experiments have proved that the LVQ-based anti-spam email filter has better performance than Bayes-based and BP neural network.-based approaches.

The remaining of this paper is organized as follows: section 2 introduces vector weight and feature extraction based on MI, section 3 introduces spam email classification and describes anti-spam email LVQ algorithm and parameters setting, Section 4 shows our experiments and results, finally, section 5 is some conclusions.

2. Email sample and data preprocessing

2.1 Email representation

Vector space model [8] is a text representing approach which is widely used and has good performance in TC. Email is regarded as a vector space which is composed of a group of orthogonal key words. Let the dimension of vector space be n, email d represents by V (d) $=(x_1, x_2, \cdots, x_n)$, with the value of elements of vector being weight of each feature key word in email d..

We use TFIDF [9] approach to calculate feature weight. In TFIDF approach, the frequency of a key word in document is directly proportional to the frequency which the word appears in the document and is inversely proportional to the number of documents which contain the word. Therefore, TFIDF of word t_i in document is (1):

$$TFIDF_i = TF_i \times \log (N / DF_i) \quad (1)$$

Where, TF_i is the frequency that word t_i appears in document d, N are the total numbers of training documents, DF_i represents the numbers of documents which contain word t_i.

2.2 Feature extraction

Feature of email may be based on word (e.g. price, adult and shop) or phrase (e.g. on sale, be over 21) as well as non-textual properties (e.g. whether or not a mail contains attachments or html tag). We adopt feature based on word in our experiments in order to focus on our algorithm performance and simplify our test.

A mail contains many different words. A large part of words contribute little to the classification, what's more, some words may play a negative role in classifying process. Hence, it is necessary to select some important words as features. We select features by MI [10] (Mutual Information) method to compress features and reduce dimensionality. MI is widely used in text categorization. MI of word t corresponding to class is calculated (2):

$$MI(t,s) \approx \log \frac{A \times N}{(A+B) \times (A+C)} \quad (2)$$

where A are the numbers of emails which contain word t and belong to class s, B are that of emails which contain word but not belong to class s. C are that of emails which belong to class s but not contain word t. N is the total email number in training corpus. For multi-class, we compute MI of the word corresponding to every class respectively and then select the maximum according to formula (3),

$$MI_{\max}(t) = \max_{i=1}^{m} MI(t,s_i) \quad (3)$$

where m denote the numbers of classes, we will choose feature words whose MI are bigger than a threshold so as to decrease the dimensionality of vector space.

3. Anti-spam email LVQ model

3.1 Spam email category.

Spam emails, which mainly are commercial and political emails at present, vary significantly in content. Related statistic [11] shows that spam emails are mainly composed of shopping online, promoting IT products, get-rich, adult products , vacation, political

information, business information, pornography/violence as well as other in content, shown in figure 1, therefore, category of spam emails is rather wide . Clustering centers of different subclasses of spam emails are different, while feature words of the whole spam emails are sparse, so that it is difficult to distinguish spam emails from legitimate emails. If we divide spam emails into several subclasses according to content, feature words of every subclass is closer and more related so as to identify spam emails easily. Hence, in our approach, if a mail belongs to one of subclasses of spam emails, due to legitimate emails being usually rather different from above subclasses of spam email in content, the email is regard as a spam email.

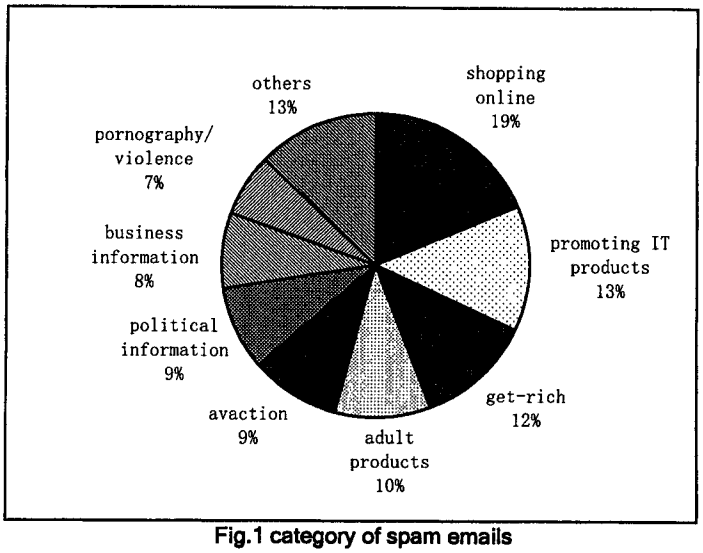

Fig.1 category of spam emails

3.2 Learning vector quantization neural network model

LVQ network[12] is a hybrid network, which form classification through supervise and unsupervised learning. The model is divided into two layers. The first layer is competitive layer, in which each neuron represents a subclass, and the second is output layer, in which each neuron represents a class. A class may be composed of several subclasses. The second layer combines several subclasses into a class through W2 matrix. So LVQ network may create complex boundaries through combining several subclasses into a class. Therefore, LVQ is suited to classify spam emails which have several subclasses.

3.3 Anti-spam email LVQ algorithm

- initialize weight vectors $W=\{ w_1, w_2, \cdots, w_n \}$, and learning rate $\alpha \in [0, 1]$
- select an example from training email corpus, and calculate distance between weight vectors and it respectively. We take the place of Euclidean distance in formula with Cosine distance, which represents similarity of two texts. cosine distance is defined as follows:

$$Sim(U,V) = \frac{\sum_{k=1}^{n} w_{uk} \cdot w_{vk}}{\sqrt{\sum_{k=1}^{n} w_{uk}^2} \sqrt{\sum_{k=1}^{n} w_{vk}^2}} \quad (4)$$

then compare similarities between the example and each weight vector, in the result, the neuron with maximum similarity, wins and outputs 1, other neuron of hidden layer output 0

$$a^1 = \max(Sim(x, x^i)) \quad (5)$$

- adjust weight, if a input example belongs to class r, the neuron c which wins in competitive learning belongs to class s, we will adjust weight in accordance with formula (6)

$$\begin{cases} w_c(t+1) = w_c(t) + u(t)[x(t) - w_c(t)]; & r = s \\ w_c(t+1) = w_c(t) - u(t)[x(t) - w_c(t)]; & r \neq s \\ w_i(t+1) = w_i(t); & i \neq c \end{cases} \quad (6)$$

- modify learning rate u(t), decrease u(t) when iteration increasing
- check stopping condition, whether is iterative times enough.

3.4 Parameter setting

Figure 2 shows our anti-spam email LVQ network model, in input layer, we choose 100 feature words as input nodes. According to previous experiments, when the number of feature is set 100, the filter has a better performance, if the number continues to increase, the performance improves slightly, but calculation increases sharply. We choose 10 neurons in hidden layer with competitive function. On output layer, two neurons (legitimate and spam) and linear function are set.

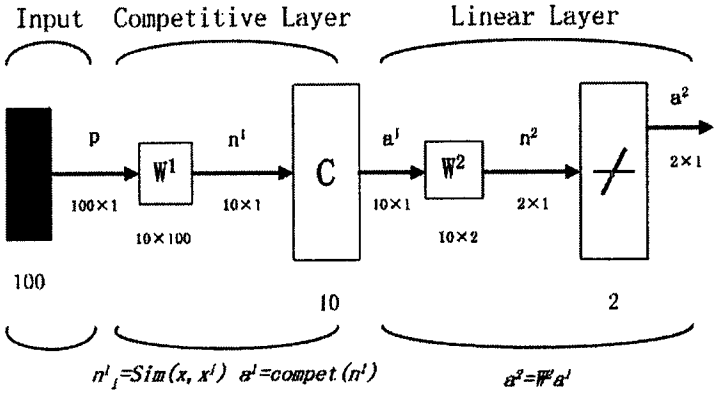

Fig2. Anti-spam emails LVQ model

In order to accelerate learning and converge as early as possible, we divide learning into two steps. The first is rapid learning. In the phase, we choose a learning rate which is bigger than 0.05. When receiving stable weight vectors, we transfer the first phase to the second phase, slow learning. We set learning rate to 0.05 in the second phase.

In order to avoid weight of neurons in the competitive layer be not able to converge during learning process, we select a prototype vector from each subclass of spam emails and legitimate emails as an initial weight vector respectively.

4. Experiments and result

This project makes use of email corpus from http://www.spamassassin.org/publiccorpus, which is open available source. We select 1000 pieces e-mails randomly from the corpus, including 580 spam e-mails, 420 legitimate e-mails. The corpus consists of all English emails whose attachments, html tags and email headers except the subject line have been stripped off.

Anti-spam email filter performance is often measured in terms of spam precision (SP) and spam recall (SR):

$$SP = \frac{n_{spam \to spam}}{n_{spam \to spam} + n_{legit \to spam}} \tag{7}$$

$n_{spam \to spam}$ are the numbers of spam emails that the filter classified as spam emails,

$n_{legit \to spam}$ are the numbers of legitimate emails that the filter classified as spam emails mistakenly.

$$SR = \frac{n_{spam \to spam}}{N_{spam}} \tag{8}$$

N_{spam} are the total numbers of spam emails

spam recall measures the percentage of spam email that the filter manages to block (intuitively its effectiveness), while spam precision measures the degree to which the blocked emails are indeed spam (the filter's safety), which is more important factor in the performance evaluation. On our intuitiveness, it is difficult to compare the performance of different filters using spam recall and precision: each filter (or filter configuration) yields a pair of spam recall and precision results; without a single unifying measure. Therefore, we introduce a criterion F1, which incorporates spam precision and spam recall. It is defined:

$$F1 = \frac{SP \times SR \times 2}{SP + SR} \tag{9}$$

In the experiments, first of all, we compare performance of the LVQ network in different training times on both open set and close set, shown in table 1. Then, we compare performances of LVQ-based approach, Bayes-based as well as BP neural network-based, shown in table 2

Table 1. Result of Test 1

Training times	Open set		Close set	
	SP(%)	SR(%)	SP(%)	SR(%)
500	90.64	87.45	91.06	88.96
1000	95.83	92.36	96.74	95.29
1500	98.97	93.58	99.51	96.86

Table 2. Result of Test 2

	SP(%)	SR(%)	F1(%)
Naïve Bayes	97.63	86.48	91.72
ANN—BP	98.42	91.26	94.70
ANN—LVQ	98.97	93.58	96.20

When the numbers of training of our LVQ network reache 500, SP and SR of the filter are not very ideal, after coming to 1000, the performances improve apparently, when reaching 1500, the performances only have a little improvement than before. In open set, SP is 98.97%, SR is 93.5%, in close set, SP is 99.51%, SR is 96.86%

In table 2, we list the performances of three algorithms, both algorithms based on neural network, which have slight improvement in SP whereas improve SR apparently, are superior to Bayes-based approach. In two neural networks, LVQ-based is better than BP-based in both SP and SR. In the terms of F1, Naïve Bayes, ANN-BP, ANN-LVQ increase in sequence.

5. Conclusion

ANN-LVQ approach further classifies spam email into several subclasses according to spam emails' content in order that emails are easy to identify. Then it combines the subclasses into complex classes by LVQ neural network so as to identify spam emails. It is proved by experiments

The numbers of neural network training affect the performance of the filter, when the numbers of training are not enough, the performance of the filter is not ideal, when the numbers reache 1500, the performance gets to stability.

Both neural network-based algorithms are usually better than that based on Bayes. Because neural networks take account of relationship between each feature words on the whole, and yet Bayse-based algorithm simply thinks that feature words are independent.

LVQ-based method outperforms that based on BP, because we classify spam emails into several subclasses in content so that the feature words of each subclass of spam email is more related and closer as well as characteristics of each subclass of spam emails are easier to identify.

References

[1] CNNIC, the 13th China Internet development status report, 2004, 1

[2] IResearch Inc. China Anti-Spam Market Research Report, 2003, 11

[3] William W. Cohen. Learning rules that classify e-mail. In proceedings of the 1996 AAAI Spring symposium in information access, 1996

[4] Sahami, M, S. Dumais, et al. A Bayesian Approach to Filtering Junk E-Mail. Learing for Text Categorization –Papers from the AAAI Workshop, Madison Wisconsin. 1998.

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

[6] Duhong chen, Tongjie et al. Spam Email Filter Using Naive Bayesian, Decision Tree, Neural Network and AdaBoost, http://www.cs.iastate.edu/~tongjie/spamfilter/paper.pdf

[7] James Clark, Irna Koprinska, Josiah Poon, A neural network based approach to automated e-mail classification , Proceedings of the IEEE/WIC international conference on web intelligence.

[8] Salton G , Wong A, Yang C S . A vector space model for automatic indexing . Communication s of the ACM , 1975.

[9] Salton G. Introduction to modern information retrieval . New York McGraw-Hill Book company. 1983.

[10] Kenneth Ward Church and Tatrick Hanks. Workd association norms, mutual information and lexicography. In proceedings of ACL27, Wancouver , Canada, 1989.

[11] China anti-spam market research report in 2004, IResearch Inc. 2004

[12] Martin T.Hagan, Howard B.Demuth, Nark H. Beale, Neural network design, China Machine Press, 2002,8