Spam Detection Using Text Clustering

Minoru Sasaki
Department of Computer and Information Sciences,
Faculty of Engineering, Ibaraki University
4-12-1 Naka-narusawa-cho, Hitachi, Ibaraki, Japan
sasaki@cis.ibaraki.ac.jp

Hiroyuki Shinnou
Department of Computer and Information Sciences,
Faculty of Engineering, Ibaraki University
4-12-1 Naka-narusawa-cho, Hitachi, Ibaraki, Japan
shinnou@dse.ibaraki.ac.jp

Abstract

We propose a new spam detection technique using the text clustering based on vector space model. Our method computes disjoint clusters automatically using a spherical k-means algorithm for all spam/non-spam mails and obtains centroid vectors of the clusters for extracting the cluster description. For each centroid vectors, the label('spam' or 'non-spam') is assigned by calculating the number of spam email in the cluster. When new mail arrives, the cosine similarity between the new mail vector and centroid vector is calculated. Finally, the label of the most relevant cluster is assigned to the new mail. By using our method, we can extract many kinds of topics in spam/non-spam email and detect the spam email efficiently. In this paper, we describe the our spam detection system and show the result of our experiments using the Ling-Spam test collection.

1. Introduction

In recent years, spam email or more properly, Unsolicited Bulk Email (UBE) is a widespread problem on the Internet. Spam email is so cheap to send that unsolicited messages are sent to a large number of users indiscriminately. When a large number of spam messages are received, it is necessary to take a long time to identify spam or non-spam email and their email messages may cause the mail server to crush.

To solve the spam problem, there have been several attempts to detect and filter the spam email on the client-side. In previous research, many Machine Learning(ML) approaches are applied to the problem, including Bayesian classifiers as Naive Bayes[1, 3, 7, 11], C4.5[10], Ripper[4] and Support Vector Machine(SVM)[6, 9] etc. In these approaches, Bayesian classifiers obtained good results by many researchers so that it widely applied to several filtering softwares. However, almost approaches learn and find the distribution of the feature set in only the spam and the non-spam messages. Today, there are many type of spam email, for example, advertisements for the purpose of making money or selling something, urban legends for the purpose of spreading hoaxes or rumors etc. Moreover, there are HTML mails contains web bug which is a graphic in an email message designed to monitor who is reading the message. Therefore, some of spam mails are judged to be non-spam email even if we use the existing filtering techniques.

In this research, we propose a new spam detection technique using the text clustering based on vector space model. This method construct the spam detection model by the contents of various kinds of mail and find spam more efficiently. The system computes disjoint clusters automatically using a spherical k-means algorithm[5] for all spam/non-spam mails and obtains centroid vectors of the clusters for extracting the cluster description. For each centroid vectors, the label('spam' or 'non-spam') is assigned by calculating the number of spam email in the cluster. When new mail arrives, the cosine similarity between the new mail vector and centroid vector is calculated. Finally, the label of the most relevant cluster is assigned to the new mail. By using our method, we can extract many kinds of topics in spam/non-spam email and detect the spam email efficiently. In this paper, we describe the our spam detection system and show the result of our experiments using the Ling-Spam[1, 2, 12] test collection.

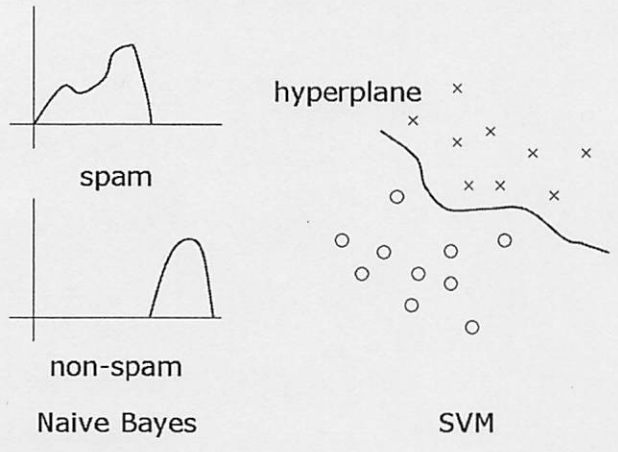

Figure 1. spam detection model using Naive Bayes and SVM

2. Motivation

Spammers send out various kinds of spam so that no system can detect all spam with 100% reliability. In past days, we have got a lot of ads to sites showing pornography. But in recent days, there are some kinds of spam such as drug ads, chain letters and urban legend. Using Bayesian classifiers[3] which widely applied to several filtering softwares, they learn and find the only two distribution of the spam and the non-spam. Therefore, we obtain good results for typical spam content, but they does not work with most spam which rarely comes twice. Consequently, it becomes more difficult to detect all spam using one distribution of spam messages growing increasingly diverse.

To make it possible to detect various kinds of spam, the system does not only construct a static model by all the training data, but also it is really desirable to modify the model dynamically for the newly received mail. If the system make a wrong judgment of the newly received mail using the existing model, it is necessary to learn the new mail to judge it correctly. However, the system spend much time to learn all the mail containing the new mail and construct the new model. If the number of all the mail is comparatively small, the model can be reconstructed in the short period o the time. But generally, the number of mail is too large to construct the model so fast. Moreover, the user does not necessarily have all spam in the previous construction of the model. So it is highly possible that the previous model can not updated properly. Therefore, we consider that the model needs easily updatable to improve performance such as incremental text classification and relevance feedback[13] and so on.

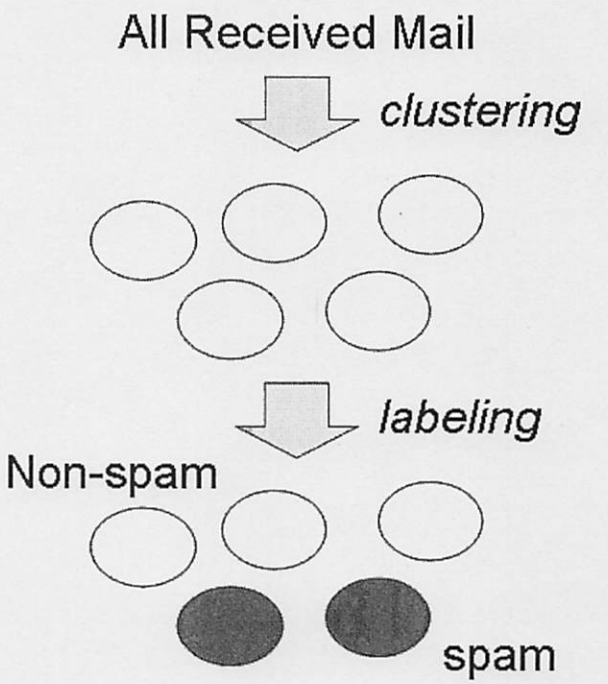

Figure 2. our spam detection model

3. Spam Detection System

In this section, we propose a new spam detection technique using the text clustering based on vector space model. In this method, the system automatically construct the spam detection model by the contents of various kinds of mail and find spam more efficiently. To obtain the spam detection model, we use the clustering algorithm called the spherical k-means algorithm[5] for all the received mail. This algorithm divide the mail set into the predefined number of clusters.

For each clusters, cluster centroid vectors are calculated as Cluster Representative. By obtaining the clusters, Similarity calculation between a new mail and the clusters can be performed easily. In the previously proposed methods such as Naive Bayes classifier and SVM filter, contents of spam are represented as one term statistic. However, using our method, the contents of various kinds of mail are represented as several term statistics as the centroid vectors.

By obtaining the centroid vectors,the label('spam' or 'non-spam') is assigned by calculating the number of spam mail in the cluster. If the ratio of spam mail to all mail in the cluster is higher than the ratio which consisted of 70% to 85%, we consider a cluster as spam. Thus, a set of clusters can be partitioned into spam and non-spam clusters.

When we obtain centroid vectors of spam and non-spam clusters, the system judges whether a new mail is spam. First, new received mail is transformed into the vector in

Filter	Cluster	Spam Prec.	Non-Spam Prec.
bare	50	91.84%	99.17%
bare	100	89.80%	99.59%
lemm	50	95.92%	98.76%
lemm	100	95.92%	97.52%
stop	50	93.88%	99.17%
stop	100	95.92%	98.35%
lemm+stop	50	97.96%	98.76%
lemm+stop	100	100%	96.28%

Table 1. Experimental results of our system

the same way of the vector space model for information retrieval. After obtaining the vector, we can calculate the cosine similarity between the new mail vector and centroid vector for each clusters. Finally, the label of the most relevant cluster is assigned to the new mail.

4. Experiment

When each mail document is represented by a vector, the elements of the vector d are assigned two-part values [14]

$$w_{ij} = L_{ij} \times G_i.$$

In our experiments, the factor L_{ij} is a local weight that reflects the weight of term i within document j and the factor G_i is a global weight that reflects the overall value of term i as an indexing term for the entire document collection as follows:

$$w_{ij} = L_{ij} \times G_i = f_{ij} \cdot \log \frac{n}{df_i},$$

where n is the number of documents in the collection, f_{ij} is the frequency of the i-th term in the j-th document, and F_i is the number of documents containing the i-th term throughout the entire document collection.

To evaluate efficiency of our system, we experiment with spam detection using freely available test collection Ling-Spam. The Ling-Spam collection consists of 2412 non-spam messages and 481 spam messages by hand categorization. By using stop-list and lemmatizer, this collection consists four collections: bare(untreated), lemm(using lemmatizer), stop(using stop-list) and lemm+stop(using lemmatizer + stop-list). In our experiments, the data set contains 2170 non-spam messages and 432 spam messages and the test set contains 242 non-spam messages and 49 spam messages.

Table 1 shows the results of the experiments. In this figure, our system provides the high-performance for both spam and non-spam messages. The spam precision is more than about 90% and the non-spam precision is more than 96% for all collections. Moreover, to make an objective evaluation of our method, precision of our method is compared with that of other methods. In this comparison, we use Support Vector Machine(SVM)[8] and bogofilter[3]. SVM is one of the most powerful machine learning method and bogofilter is a Bayesian spam filter. We show the result in the table 2. This results show that the precision using our method is better than the bogofilter and is approximately equivalent to the SVM. So it can be concluded that using the spam and non-spam clusters based on the unsupervised clustering is a effective method for detecting spam.

Filter	SVM		bogofilter	
	Spam Prec.	Non-Spam Prec.	Spam Prec.	Non-Spam Prec.
bare	97.96%	100%	36.73%	100%
lemm	97.96%	100%	42.86%	100%
stop	97.96%	100%	36.73%	100%
lemm+stop	100%	100%	40.82%	100%

Table 2. Experimental results using SVM and bogofilter

However, we define the threshold value of spam cluster as 70% so that the non-spam precision is not 100%. Thus we define the greater threshold value than 70% and calculate the precision of spam on the condition that the non-spam precision is nearly 100%. the table 3 and 4 show the results of these experiments using TF-IDF (Text Frequency · Inverse Document Frequency) and TF respectively as term weighting. The spam precision is about 90% so that our method provides the high-performance for spam messages. Additionally, the spam precision using TF is better than that using TF-IDF except result of lemm_stop.

5. Conclusion

In this paper, we propose a new spam detection technique using the text clustering based on vector space model. This method construct the spam detection model by the contents of various kinds of mail and find spam more efficently. The experimental results show that the precision using our method is better than the bogofilter and is approximately equivalent to the SVM. So it can be concluded that using the spam and non-spam clusters based on the unsupervised clustering is a effective method for detecting spam.

Further work would be required to refine the spam and non-spam clusters using dynamic updating such as relevance feedback.

Filter	Ratio	Spam Prec.	Non-Spam Prec.
bare	0.8	95.92%	99.59%
bare	0.85	87.76%	100.00%
lemm	0.7	95.92%	98.76%
lemm	0.8	71.43%	100.00%
stop	0.8	91.84%	99.59%
stop	0.85	89.80%	100.00%
lemm_stop	0.7	97.96%	98.76%
lemm_stop	0.75	83.67%	100.00%

Table 3. Experimental results using some threshold values(TFIDF)

Filter	Ratio	Spam Prec.	Non-Spam Prec.
bare	0.8	95.92%	99.59%
bare	0.85	95.92%	100.00%
lemm	0.7	97.96%	99.17%
lemm	0.8	93.88%	100.00%
stop	0.8	95.92%	99.59%
stop	0.85	95.92%	99.59%
lemm_stop	0.8	89.80%	99.17%
lemm_stop	0.85	75.51%	99.17%

Table 4. Experimental results using some threshold values(TF)

References

[1] I. Androutsopoulos, J. Koutsias, K. Chandrinos, G. Paliouras, and C. Spyropoulos. 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)*, pages 9–17, 2000.

[2] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C. Spyropoulos, and P. Stamatopoulos. Learning to filter spam e-mail: A comparison of a naive bayesian and a memory-based approach. In *Proceedings of the workshop Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD-2000)*, pages 1–13, 2000.

[3] *Bogofilter*. http://bogofilter.sourceforge.net/.

[4] W. W. Cohen. Learning rules that classify e-mail. In *Proceedings of the 1996 AAAI Spring Symposium on Machine Learning in Information Access*, pages 203–214, 1996.

[5] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. Technical report, IBM Almaden Research Center, 1999.

[6] H. Druker. Support vector machines for spam categorization. In *Proceedings of the IEEE Transaction on Neural Networks*, volume 10, pages 1048–1054, 1999.

[7] P. Graham. *Better Bayesian Filtering*. http://www.paulgraham.com/better.html.

[8] T. Joachims. *Learning to Classify Text Using Support Vector Machines*. Dissertation, Kluwer, 2002.

[9] A. Kolcz and J. Alspector. Svm-based filtering of e-mail spam with content-specific misclassification costs. In *Proceedings of the TextDM' 01 Workshop on Text Mining, IEEE International Conference on Data Mining*, pages 1048–1054, 2001.

[10] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, San Mateo, CA, 1993.

[11] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A bayesian approach to filtering junk e-mail. In *Proceedings of the AAAI-98 Workshop on Learning for Text Categorization*, pages 1048–1054, 1998.

[12] G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C. Spyropoulos, and P. Stamatopoulos. Stacking classifiers for anti-spam filtering of e-mail. In *Proceedings of the 6th Conference on Empirical Methods in Natural Language Processing(EMNLP 2001)*, pages 44–50, 2001.

[13] G. Salton and C. Buckley. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41:288–297, 1990.

[14] I. H. Witten, A. Moffat, and T. C. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Van Nostrand Reinhold, New York, 1994.