

A Comparative Study for Email Classification

Seongwook Youn and Dennis McLeod
University of Southern California,
Los Angeles, CA 90089
USA

Abstract - Email has become one of the fastest and most economical forms of communication. However, the increase of email users have resulted in the dramatic increase of spam emails during the past few years. In this paper, email data was classified using four different classifiers (Neural Network, SVM classifier, Naïve Bayesian Classifier, and J48 classifier). The experiment was performed based on different data size and different feature size. The final classification result should be '1' if it is finally spam, otherwise, it should be '0'. This paper shows that simple J48 classifier which make a binary tree, could be efficient for the dataset which could be classified as binary tree.

I. INTRODUCTION

Email has been an efficient and popular communication mechanism as the number of Internet users increase. Therefore, email management is an important and growing problem for individuals and organizations because it is prone to misuse. The blind posting of unsolicited email messages, known as spam, is an example of misuse. Spam is commonly defined as the sending of unsolicited bulk email - that is, email that was not asked for by multiple recipients. A further common definition of a spam restricts it to unsolicited commercial email, a definition that does not consider non-commercial solicitations such as political or religious pitches, even if unsolicited, as spam. Email was by far the most common form of spamming on the internet.

Text classification including email classification presents challenges because of large and various number of features in the dataset and large number of documents. Applicability in these datasets with existing classification techniques was limited because the large number of features make most documents undistinguishable.

In many document datasets, only a small percentage of the total features may be useful in classifying documents, and using all the features may adversely affect performance. The quality of training dataset decides the performance of both the text classification algorithms and feature selection algorithms. An ideal training document dataset for each particular category will include all the important terms and their possible distribution in the category.

The classification algorithms such as Neural Network (NN), Support Vector Machine (SVM), and Naïve

Bayesian (NB) are currently used in various datasets and showing a good classification result.

The problem of spam filtering is not a new one and there are already a dozen different approaches to the problem that have been implemented. The problem was more specific to areas like Artificial intelligence and Machine Learning. Several implementations had various trade-offs, difference performance metrics, and different classification efficiencies. The techniques such as decision tree (J48), Naive Bayesian classifiers, Neural Networks, Support Vector Machine, etc had various classification efficiencies. The remainder of the paper is organized as follows: Section 2 describes existing related works; Section 3 introduces four spam classification methods used in the experiment; Section 4 discusses the experimental results; Section 5 concludes the paper with possible directions for future work.

II. RELATED WORKS

[17] compared a cross-experiment between 14 classification methods, including decision tree, Naïve Bayesian, Neural Network, linear squares fit, Rocchio. KNN is one of top performers, and it performs well in scaling up to very large and noisy classification problems.

[14] showed that bringing in other kinds of features, which are spam-specific features in their work, could improve the classification results. [11] showed a good performance reducing the classification error by discovering temporal relations in an email sequence in the form of temporal sequence patterns and embedding the discovered information into content-based learning methods. [13] showed that the work on spam filtering using feature selection based on heuristics.

Approaches to filtering junk email are considered [2, 5, 14]. [6] and [7] showed approaches to filtering emails involve the deployment of data mining techniques. [3] proposed a model based on the Neural Network to classify personal emails and the use of Principal Component Analysis (PCA) as a preprocessor of NN to reduce the data in terms of both dimensionality as well as size. [1] compared the performance of the Naïve Bayesian filter to an alternative memory based learning approach on spam filtering.

[15] and [18] developed a algorithm to reduce the feature space without sacrificing remarkable classification

accuracy, but the effectiveness was based on the quality of the training dataset.

In the classification experiment for spam mail filtering, J48 showed better result than NB, NN, or SVM classifier.

III. SPAM CLASSIFICATION METHODS

Generally, the main tool for email management is text classification. A classifier is a system that classifies texts into the discrete sets of predefined categories. For the email classification, incoming messages will be classified as spam or legitimate using classification methods.

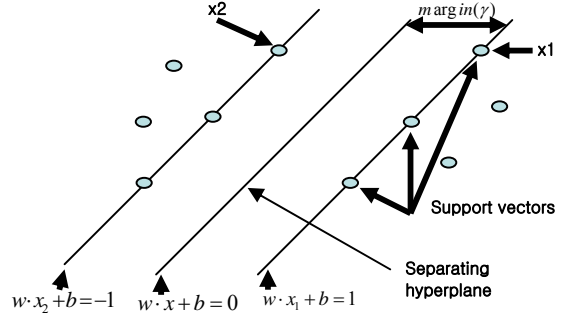
A. Neural Network (NN)

Classification method using a NN was used for email filtering long time ago. Generally, the classification procedure using the NN consists of three steps, data pre-processing, data training, and testing. The data pre-processing refers to the feature selection. Feature selection is the way of selecting a set of features which is more informative in the task while removing irrelevant or redundant features. For the text domain, feature selection process will be formulated into the problem of identifying the most relevant word features within a set of text documents for a given text learning task. For the data training, the selected features from the data pre-processing step were fed into the NN, and an email classifier was generated through the NN. For the testing, the email classifier was used to verify the efficiency of NN. In the experiment, an error BP (Back Propagation) algorithm was used.

B. Support Vector Machines (SVM) Classifier

SVMs are a relatively new learning process influenced highly by advances in statistical learning theory. SVMs have led to a growing number of applications in image classification and handwriting recognition. Before the discovery of SVMs, machines were not very successful in learning and generalization tasks, with many problems being impossible to solve. SVMs are very effective in a wide range of bioinformatic problems. SVMs learn by example. Each example consists of a m number of data points (x_1, \dots, x_m) followed by a label, which in the two class classification we will consider later, will be +1 or -1. -1 representing one state and 1 representing another. The two classes are then separated by an optimum hyperplane, illustrated in figure 1, minimizing the distance between the closest +1 and -1 points, which are known as support vectors. The right hand side of the separating hyperplane represents the +1 class and the left hand side represents the -1 class.

This classification divides two separate classes, which are generated from training examples. The overall aim is to generalize well to test data. This is obtained by introducing a separating hyperplane, which must maximize the margin (γ) between the two classes, this is known as the optimum separating hyperplane



Let's consider the above classification task with data points $x_i, i=1, \dots, m$, with corresponding labels $y_i = \pm 1$, with the following decision function:

$$f(x) = \text{sign}(w \cdot x + b)$$

By considering the support vectors x_1 and x_2 , defining a canonical hyperplane, maximizing the margin, adding Lagrange multipliers, which are maximized with respect to α :

$$W(\alpha) = \sum_{i=1}^m \alpha_i - \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)$$

$$\left(\sum_{i=1}^m \alpha_i y_i = 0, \alpha_i \geq 0 \right)$$

C. Naïve Bayesian (NB) Classifier

Naïve Bayesian classifier is based on Bayes' theorem and the theorem of total probability. The probability that a document d with vector $\vec{x} = \langle x_1, \dots, x_n \rangle$ belongs to category c is

$$P(C = c | \vec{X} = \vec{x}) = \frac{P(C = c) \cdot P(\vec{X} = \vec{x} | C = c)}{\prod_{k \in \{\text{spam}, \text{legit}\}} P(C = k) \cdot P(\vec{X} = \vec{x} | C = k)}$$

However, the possible values of \vec{X} are too many and there are also data sparseness problems. Hence, Naïve Bayesian classifier assumes that X_1, \dots, X_n are conditionally independent given the category C . Therefore, in practice, the probability that a document d with vector $\vec{x} = \langle x_1, \dots, x_n \rangle$ belongs to category c is

$$P(C = c | \vec{X} = \vec{x}) = \frac{P(C = c) \cdot \prod_{i=1}^n P(X_i = x_i | C = c)}{\prod_{k \in \{\text{spam}, \text{legit}\}} P(C = k) \cdot \prod_{i=1}^n P(X_i = x_i | C = k)}$$

$P(X_i | C)$ and $P(C)$ are easy to obtain from the frequencies of the training dataset. So far, a lot of

researches showed that the Naïve Bayesian classifier is surprisingly effective.

D. J48 Classifier

J48 classifier is a simple C4.5 decision tree for classification. It creates a binary tree.

IV. RESULTS

In this section, four classification methods (Neural Network, Support Vector Machine classifier, Naïve Bayesian classifier, and J48 classifier) were evaluated the effects based on different datasets and different features. Finally, the best classification method was obtained from the training dataset. 4500 emails were used as a training dataset. 38.1% of dataset were spam and 61.9% were legitimate email. To evaluate the classifiers on training dataset, we defined an accuracy measure as follows.

$$Accuracy(\%) = \frac{Correctly_Classified_Emails}{Total_Emails} * 100$$

Also, Precision and Recall were used as the metrics for evaluating the performance of each email classification approach.

A. Effect of dataset on performance

An experiment measuring the performance against the size of dataset was conducted using dataset of different sizes listed in Fig.1. The experiment was performed with 55 features from TF/IDF. For example, in case of 1000 dataset, Accuracy was 95.80% using J48 classifier.

Data Size	NN	SVM	Naïve Bayesian	J48
1000	93.50%	92.70%	97.20%	95.80%
2000	97.15%	95.00%	98.15%	98.25%
3000	94.17%	92.40%	97.83%	97.27%
4000	89.60%	91.93%	97.75%	97.63%
4500	93.40%	90.87%	96.47%	97.56%

With 55 features
Fig. 1. Classification result based on data size

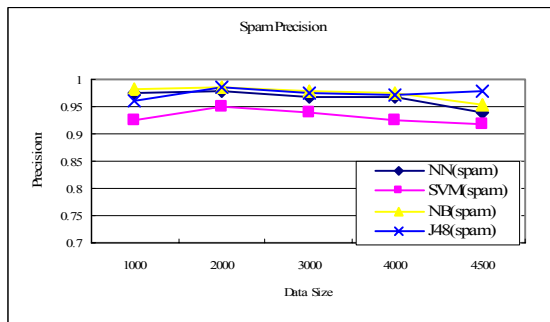


Fig. 2. Spam precision based on data size

A few observations can be made from this experiment. As shown in Fig. 1, the average of correct classification rate

for both J48 and NB was over 95%. Dataset size was not an important factor in measuring precision and recall. The results show that the performance of classification was not stable. For four different classification methods, precision of spam mail was shown in Fig. 2, likewise, precision of legitimate mail was shown in Fig. 3.

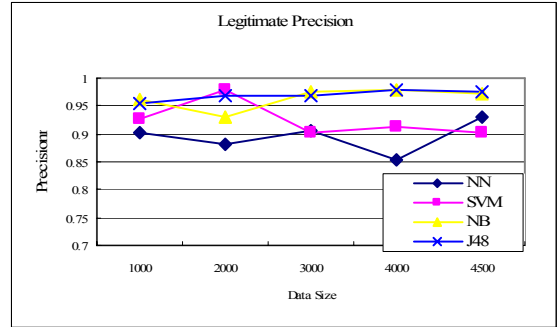


Fig. 3. Legitimate precision based on data size

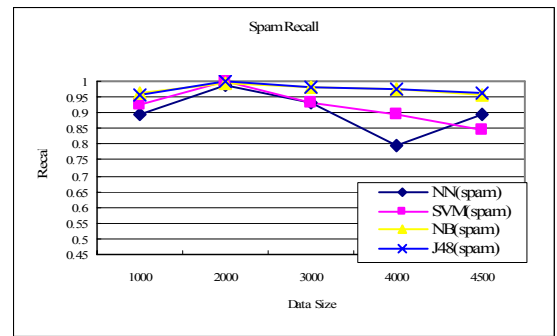


Fig. 4. Spam recall based on data size

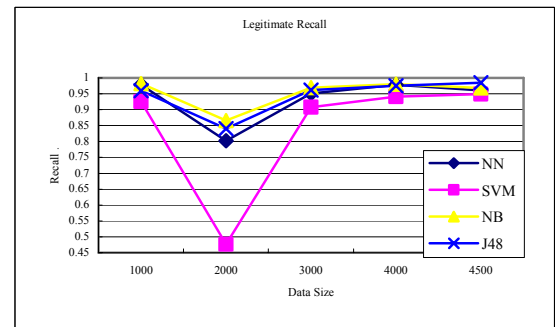


Fig. 5. Legitimate recall based on data size

As shown in Fig. 2, 3, 4, and 5, the precision and recall curves of J48 and NB classification were better than the ones of NN and SVM. Also, the average precision and recall for both J48 and NB was over 95%. In Fig. 5, legitimate recall values were sharply decreased at the data size 2000. The increase of spam mail in the training dataset between 1000 and 2000 result in a sharp decrease of legitimate recall values for all classifiers

B. Effect of feature size on performance

The other experiment measuring the performance against the size of dataset was conducted using different features listed in Fig. 6. 4500 email dataset was used for the experiment. For example, in case of 10 features, Accuracy was 94.84% using J48 classifier. The most frequent words in spam mail were selected as features. Generally, the result of classification was increased for all classification methods according to the feature size increased.

Feature Size	NN	SVM	Naïve Bayesian	J48
10	83.60%	81.91%	92.42%	94.84%
20	89.87%	85.73%	95.60%	96.91%
30	93.31%	88.87%	95.64%	97.56%
40	92.13%	89.93%	97.49%	97.13%
50	93.18%	90.27%	96.84%	97.67%
55	93.10%	90.84%	97.64%	97.56%

Fig. 6. Classification result based on feature size

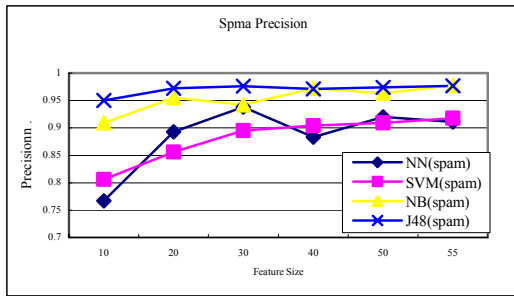


Fig. 7. Spam precision based on feature size

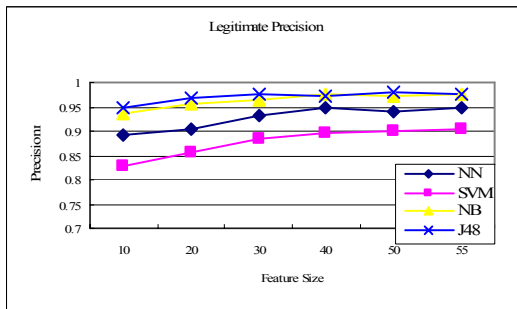


Fig. 8. Legitimate precision based on feature size

As shown in Fig. 7, 8, 9, and 10, good classification result order in the experiment was J48, NB, NN, and SVM for all cases (spam precision, legitimate precision, spam recall, and legitimate recall). The overall precision and recall for email classification increase and become stable according to the increase of the number of feature. Gradually, the accuracy increase and finally saturated with the increased feature size. As shown in Fig. 7 and 8,

J48 classifier provided the precision over 95% for every feature size irrespective of spam or legitimate. Also, J48 classifier supported over 97% of classification accuracy for more than 30 feature size. For the recall, J48 and NB showed better result than NN and SVM for both spam and legitimate mail, but J48 was a little bit better than NB.

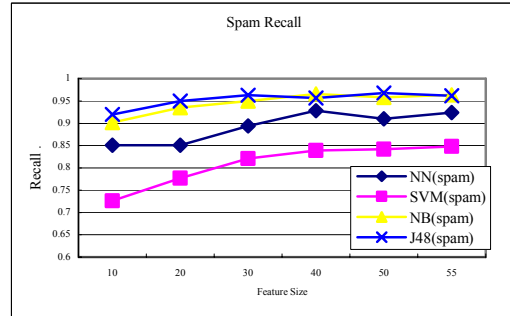


Fig. 9. Spam recall based on feature size

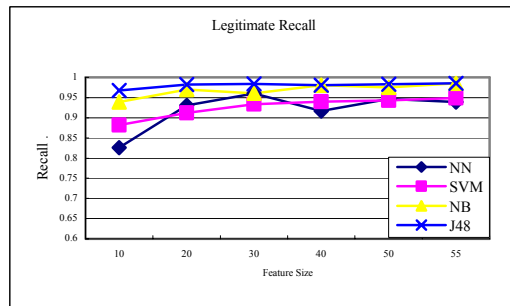


Fig. 10. Legitimate recall based on feature size

V. CONCLUSION AND FUTURE WORK

In this paper, four classifiers including Neural Network, SVM, Naïve Bayesian, and J48 were tested to filter spams from the dataset of emails. All the emails were classified as spam (1) or not (0). That was the characteristic of the dataset of email for spam filtering. J48 is very simple classifier to make a decision tree, but it gave the efficient result in the experiment. Naïve Bayesian classifier also showed good result, but Neural Network and SVM didn't show good result compared with J48 or Naïve Bayesian classifier. Neural Network and SVM were not appropriate for the dataset to make a binary decision. From this experiment, we can find it that a simple J48 classifier can provide better classification result for spam mail filtering. In the near future, we plan to incorporate other techniques like different ways of feature selection, classification using ontology. Also, classified result could be used in Semantic Web by creating a modularized ontology based on classified result. There are many different mining and classification algorithms, and parameter settings in each algorithm. Experimental results in this paper are based on the default settings. Extensive experiments with different

settings are applicable in WEKA. Moreover, different algorithms which are not included in WEKA can be tested. Also, experiments with various feature selection techniques should be compared.

Furthermore, we plan to create an adaptive ontology as a spam filter based on classification result. Then, this ontology will be evolved and customized based on user's report when a user requests spam report. By creating a spam filter in the form of ontology, a filter will be user customized, scalable, and modularized, so it can be embedded to many other systems. This ontology also may be used to block porn web site or filter out spam emails on the Semantic Web.

ACKNOWLEDGEMENT

This research has been funded in part by the Integrated Media Systems Center, a National Science Foundation Engineering Research Center, Cooperative Agreement No. EEC-9529152.

REFERENCES

- [1] I. Androutsopoulos, G. Paliouras, V. Karkaletsis, G. Sakkis, C. Spyropoulos, and P. Stamatiopoulos, "Learning to Filter Spam E-Mail: A Comparison of a Naive Bayesian and a Memory-Based Approach," CoRR cs.CL/0009009, 2000.
- [2] W. Cohen, "Learning rules that classify e-mail," In Proc. of the AAAI Spring Symposium on Machine Learning in Information Access, 1996.
- [3] B. Cui, A. Mondal, J. Shen, G. Cong, and K. Tan, "On Effective E-mail Classification via Neural Networks," In Proc. of DEXA, 2005, pp. 85-94.
- [4] E. Crawford, I. Koprinska, and J. Patrick, "Phrases and Feature Selection in E-Mail Classification," In symposium of ADCS, 2004, pp. 59-62.
- [5] Y. Diao, H. Lu, and D. Wu, "A comparative study of classification based personal e-mail filtering," In Proc. of fourth PAKDD, 2000.
- [6] T. Fawcett, "in vivo spam filtering: A challenge problem for data mining," In Proc. of ninth KDD Explorations vol.5 no.2, 2003.
- [7] K. Gee, "Using latent semantic indexing to filter spam," In Proc. of eighteenth ACM Symposium on Applied Computing, Data Mining Track, 2003.
- [8] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen, "Combating Web Spam with TrustRank," In VLDB, 2004, pp. 576-587.
- [9] T. Joachims, "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization," In ICML, 1997, pp. 143-151.
- [10] T. Joachims, "Structured Output Prediction with Support Vector Machines," SSPR/SPR, 2006, pp. 1-7
- [11] S. Kiritchenko, S. Matwin, and S. Abu-Hakima, "Email Classification with Temporal Features," Intelligent Information Systems 2004, pp. 523-533.
- [12] S. Martin, B. Nelson, A. Sewani, K. Chen, and A. Joseph, "Analyzing Behavioral Features for Email Classification," CEAS, 2005.
- [13] T. Meyer, and B. Whateley, "SpamBayes: Effective open-source, Bayesian based, email classification system," In Proc. of first Conference of Email and Anti-Spam, 2004.
- [14] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A Bayesian Approach to Filtering Junk E-Mail," In Proc. of the AAAI Workshop on Learning for Text Categorization, 1998.
- [15] S. Shankar and G. Karypis, "Weight adjustment schemes for a centroid based classifier," Computer Science Technical Report TR00-035, 2000.
- [16] I. Stuart, S. Cha, and C. Tappert, "A Neural Network Classifier for Junk E-Mail," in Document Analysis Systems, 2004, pp. 442-450.
- [17] Y. Yang, "An Evaluation of Statistical Approaches to Text Categorization," Journal of Information Retrieval, Vol 1, No. 1/2, 1999, pp. 67-88.
- [18] Y. Yang and J. Pedersen, "A Comparative Study on Feature Selection in Text Categorization," In ICML, 1997, pp. 412-420.
- [19] S. Youn and D. McLeod, "Ontology Development Tools for Ontology-Based Knowledge Management," In Encyclopedia of E-Commerce, E-Government and Mobile Commerce. Idea Group Inc, 2006.