Spam Filtering with Active Feature Identification

Masayuki Okabe
Toyohashi University of Technology
Tenpaku 1-1, Toyohashi, Aichi, Japan
okabe@imc.tut.ac.jp

Seiji Yamada
National Institute of Infomatics
Chiyoda, Tokyo, Japan
seiji@nii.ac.jp

***Abstract*—This paper proposes a spam filtering method that utilizes active learning and feature identification. Identification of effective features are very important procedure in spam filtering because spam mail includes so much meaningless words that are slightly different from each other. Those words bring down much calculation cost and performance reduction in filtering process. Thus identifying effective and ineffective features is promising approach in spam filtering. However traditional feature selection methods calculate the score of features based on some amount of labeled training data. This assumption does not hold in the situation of spam filtering. Spam filtering process starts with non or few labeled data, and gradually increases labeled data using user feedback. We propose a method to identify effective features through this active learning process in spam filtering based on naive Bayes approach. Experimental results show that our method outperforms traditional method with no feature identification.**

I. Introduction

There is a report that 96.5% of e-mails flowing in the Internet are spams [1]. Spam filtering is a disgusting task for the most of users who use e-mail in daily life. They not only annoys users but also make traffic jams in the Internet. Most of recent e-mail clients adopt spam filtering function. However it often fails to remove spams at it's starting point. To filter e-mails correctly, enough feedback have to be given from the client's user. In the early stage of the client's usage, the user have to notify mis-filtered e-mails to the client every time new e-mails arrives. Without enough training, we may miss e-mails that have critical information for us.

The process of giving feedback to e-mail clients by labeling 'ham' or 'spam' shown in Fig.1 is called active learning in the filed of machine learning [2]. This learning technique tries to reduce the cost of data labeling, i.e. reducing the number of data to label by finding effective data to make filter. This approach is to reduce the number of data that does not contribute to make spam filters.

Several active learning techniques have been proposed so far [3], [4], [5]. The most famous one is uncertainty sampling [6] that select a data for the candidate to label, which have the most uncertain value of a decision function. Decision function is defined by the filtering method used in a spam filtering system, such as naive Bayes or support vector machines. Uncertain sampling is a type of active learning technique that tries to reduce the version space. On the other hand, there is an active learning technique that tries to reduce estimated error that is calculated by the distribution of estimated conditional class probability [7]. The results of the experiments in which error reduction approach is tested on naive Bayes show good performance compared with uncertainty sampling.

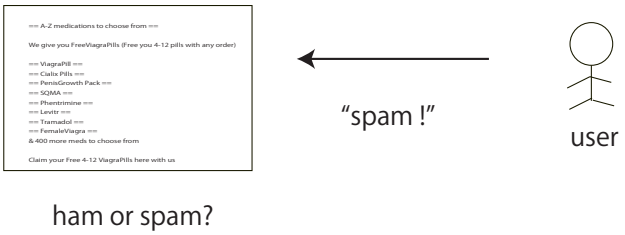

Fig. 1. Spam Filtering Process

The performance of filtering is influenced by not only the number of training data, but also the feature set used in the filtering process. There are several techniques to improve the performance of filtering. Feature selection is known to be a good approach [8]. In classification learning like filtering, feature selection methods that uses class information are promising. However a certain amount of labeled data is necessary for those methods work effectively. This is a serious problem when applying feature selection in the process of active learning. We propose a method to solve this problem. Our method assumes the class of each unlabeled data using conditional class probability estimated during active learning process. This procedure can makes a pseudo set of labeled data enough to calculate a feature's score of importance. Once we can get enough amount of (pseudo) labeled data, we have another problem that is how to select features to use for making classifiers. We will propose a method to decide the number of necessary features by finding the point of the largest gap between the sorted score list of features. These proposed methods are used to identify a useful set of features to make good classifier.

In the following sections, we first describe naive Bayes classifier and a sample error reduction approach of active learning based on the classifier in Section 2. Then we propose a feature identification method through the naive Bayes active learning in Section 3. In Section 4, we show experimental results in which our method and other two ones are compared on the data set of TREC Spam Filtering track. We also discuss about the results of some variational tests of our feature identification in Section 5. Finally we conclude in Section 6.

II. Spam Filtering with Naive Bayes Classifier

A. Naive Bayes Classifier

Naive Bayes is a basic classification technique that is simple and easy to implement but shows relatively good performance [9]. Although there are other classifiers that show better performance such as Support Vector Machines, many spam filter implementation is based on naive Bayes approach because of its simplicity.

The classifier calculates the probability for each mail to belong to 'ham' or 'spam' based on the Bayes theorem. Suppose we want to classify e-mails X into a class in C. Here $X = (x_1, x_2, \ldots, x_i)$ and $C = `ham', `spam'$.

The conditional probability $P(C|X)$ that a data belongs to a class is written by $P(C|X) = \frac{P(C)P(X|C)}{P(X)}$ in which Bayes theorem is applied.

Naive Bayes classifier adopts a generative model that assumes data X are produced by the selection of a class $c \in C$ and $x \in X$ given c. Moreover, it assumes the independence of features of a data x. According to the generative approach, the conditional probability $P(c_j|x_i, \theta)$ that a data x_i belongs to a class c_j is written as follows.

$$P(c_j|x_i, \theta) = \frac{P(c_j|\theta)\Pi_{k=1}^{n_i} P(w_{i,k}|c_j, \theta)}{\sum_{j=1}^{|C|} P(c_j|\theta)\Pi_{k=1}^{n_i} P(w_{i,k}|c_j, \theta)} \quad (1)$$

Here $w_{i,k}$ is the kth word that appears in x_i, $\theta = (\theta_c, \theta_{w|c})$ is a set parameters that includes the class prior and the probability of generating word w class c. Using maximum *a posteriori* estimation, each parameter is calculated as follows.

$$\hat{\theta}_{c_j} = \frac{1 + \sum_{i=1}^{|X|} \delta(x_i, c_j)}{|C| + |X|} \quad (2)$$

$$\hat{\theta}_{w_k|c_j} = \frac{1 + \sum_{i=1}^{|X|} N(w_k, x_i)\delta(x_i, c_j)}{|V| + \sum_{k=1}^{|V|}\sum_{i=1}^{|X|} N(w_k, x_i)\delta(x_i, c_j)} \quad (3)$$

where V is all kinds of words that are used to estimate parameters. $N(w_k, x_i)$ is word frequency of a word w_k appears in x_i. $\delta(x_i, c_j)$ is a function that returns 1 if the class of x_i equals to c_j and 0 otherwise.

The above estimation includes smoothing parameters that removes zero counts when there is no training data. This simple smoothing method is called the Laplace smoothing.

B. Naive Bayes with Active Learning

Until now, we have explained classification technique with an assumption that we have enough training data (i.e. enough mails categorized either 'ham' or 'spam'). However such assumption is unrealistic. When we start to use a mail client, there is no labeled mails. We notify spams to the client one by one as described in the introduction of this paper.

This process is called active learning where the objective is to reduce the cost of labeling procedure by selecting a data that is expected to improve the classification performance most. Among several techniques of active learning as described in the section of introduction, we adopt a sampling technique of error reduction approach proposed by Roy[Roy01]. This approach estimates the expected error E using a loss function L that calculates between true distribution of the class probability conditioned on x $P(C|X)$ and estimated one using a loss function.

Algorithm 1 Procedure for active learning on naive Bayes

```
P : a pool of unlabeled data.
E(x_i, c) : expected loss when adding a data (x_i, c)
E(x_i) : total expected loss when adding a data x_i
for i = 1 to |P| do
    E(x_i) = 0
    for c ∈ 'ham', 'spam' do
        E_(x_i,c) = 0
        for k = 1 to |P| do
            next if i == k
            E(x, c) += −P̂*(c|x) log P̂*(c|x)
        end for
        E(x_i) += P̂(c|x_i) * E(x_i, c)
    end for
end for
return argmax E(x_i)
```

$$E = \int_x L(P(c|x), \hat{P}(c|x))P(x) \quad (4)$$

As a loss function, we use the log loss function describe below.

$$L = \sum_{c \in C} -P(c|x) \log \hat{P}(c|x) \quad (5)$$

Since true distribution is unknown, we replace $P(c|x)$ to $\hat{P}(c|x)$. As a result, an expected loss when adding a data is calculated by the following formula.

$$E = \frac{1}{|P|} \sum_{x \in P} \sum_{c \in C} -\hat{P}^*(c|x) \log \hat{P}^*(c|x) \quad (6)$$

where $\hat{P}^*(c|x)$ is conditional class probability when adding a data (x, c) that is a data selected by each active learning process. Algorithm 1 shows precise algorithm of active learning. This algorithm repeats to consider each unlabeled data in the pool as a feedback candidate. For each data, it considers each possible label c for x, and add each pair (x, c) to training data set to make temporary new training data by which a temporal new classifier is created. By using this temporal classifier, resulting expected loss are estimated. Each estimated losses $E(x_i, c)$ are aggregated by multiplying its present conditional class probability $P(c|x_i)$.

Because the true distribution is unknown, the value is actually calculated by using estimated distribution for true one. According to this replacement, loss function actually calculates the sharpness of class distribution, large difference is more preferable.

III. Feature Identification through Active Learning

A. Feature selection by Information Gain

Feature selection is a technique of improving classification performance by removing unnecessary features. Although there are several measures to select effective features in text classification, we use information gain that is one of most effective technique. Information gain (IG) is calculated by the following formula.

$$\begin{aligned} IG(w) &= -\sum_{c\in C} P(c)\log P(c) \\ &\quad +P(w)\sum_{c\in C} P(c|w)\log P(c|w) \\ &\quad +P(\bar{w})\sum_{c\in C} P(c|\bar{w})\log P(c|\bar{w}) \end{aligned} \tag{7}$$

Where $P(c)$ is the probability that data with class c is selected among all training data, $P(c|w)$ is the probability that data with class c is selected among all data in which a word w appears, and $P(c|\bar{w})$ is the probability that data with class c is selected among all data in which a word w does not appear.

When using IG in active learning, there are mainly two problems. The first one is the lack of training data. Since there are few training data in the early stage of active learning, the confidence of the value of IG is low. The second is how to decide the number of features to use. Although this problem is not limited to the case of active learning, it is more serious because the confidence of IG is low. We explain the solution of these problems.

B. Identification of a feature set though active learning process

We summarize again each problem when applying feature selection in active learning and show the solution for each problem.

The first problem is how to prepare enough labeled data. The normal feature selection technique needs some amount of labeled training data. However there are small number of labeled data in the early stage of active learning. We propose a solution for the problem by assuming the class of unlabeled data that have some certain confidence of their class labels. we assume the class label of unlabeled data by the probability $P(c|x)$ with large difference between two classes, i.e. 'ham' and 'spam' class. Moreover even if there are enough unlabeled data with high probability of conditional class confidence, it is difficult to discriminate the important features if the number of data in each class are biased to one class. According to the above consideration, we adopt two threshold to prepare the source data for selecting important features to classify.

1) T_{conf} : threshold for the conditional probability of class confidence of each unlabeled data.
2) T_{bal} : threshold for the balance of the number of training data in classes.

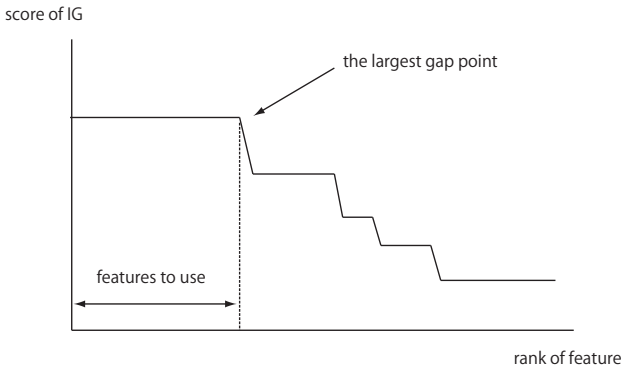

Fig. 2. Illustration of the largest gap point

Both thresholds are lower limits that must be satisfied. T_{conf} controls the number of source data for feature selection. T_{bal} is a parameter to prevent from making an unbalanced source data set where data with only one class occupies the set. This is the solution for the first problem.

The second problem is how to define the number of useful features. According to the scoring function (IG), we can discriminate the usefulness of each feature. However we also have to decide the number of feature to use in classification. To solve this problem, we propose to define the number of features by finding the largest gap point. Gap points can be found by firstly sorting the score of $IG(w)s$, and secondly check the difference of the score between rank k and rank $k+1$. If the largest gap point is K, we select features until Kth feature in the sorted list of IG as shown in Fig.4. Since the probabilities used in IG are calculated based on the number of data in which each word appears, a group of words tend to have the same value like shown as horizontal lines in Fig.4. Thus this procedure selects such groups of words by finding the largest gap.

Algorithm 2 summarizes the procedure of active learning accompanied with feature identification.

Algorithm 2 Procedure for active learning with feature identification

```
V : vocabulary set
w_i : ith word in V
Conduct algorithm1
Calculate conditional probability P(c|x) for each x
Select x that has the probability more than T_conf
Make a pseudo training data set D_pse that consist of true
    training data and data selected above procedure.
for i = 1 to |V| do
    calculate IG(w_i) using D_pse
end for
Sort V in descending order
Find the largest gap point and create new vocabulary set
    V*
return V*
```

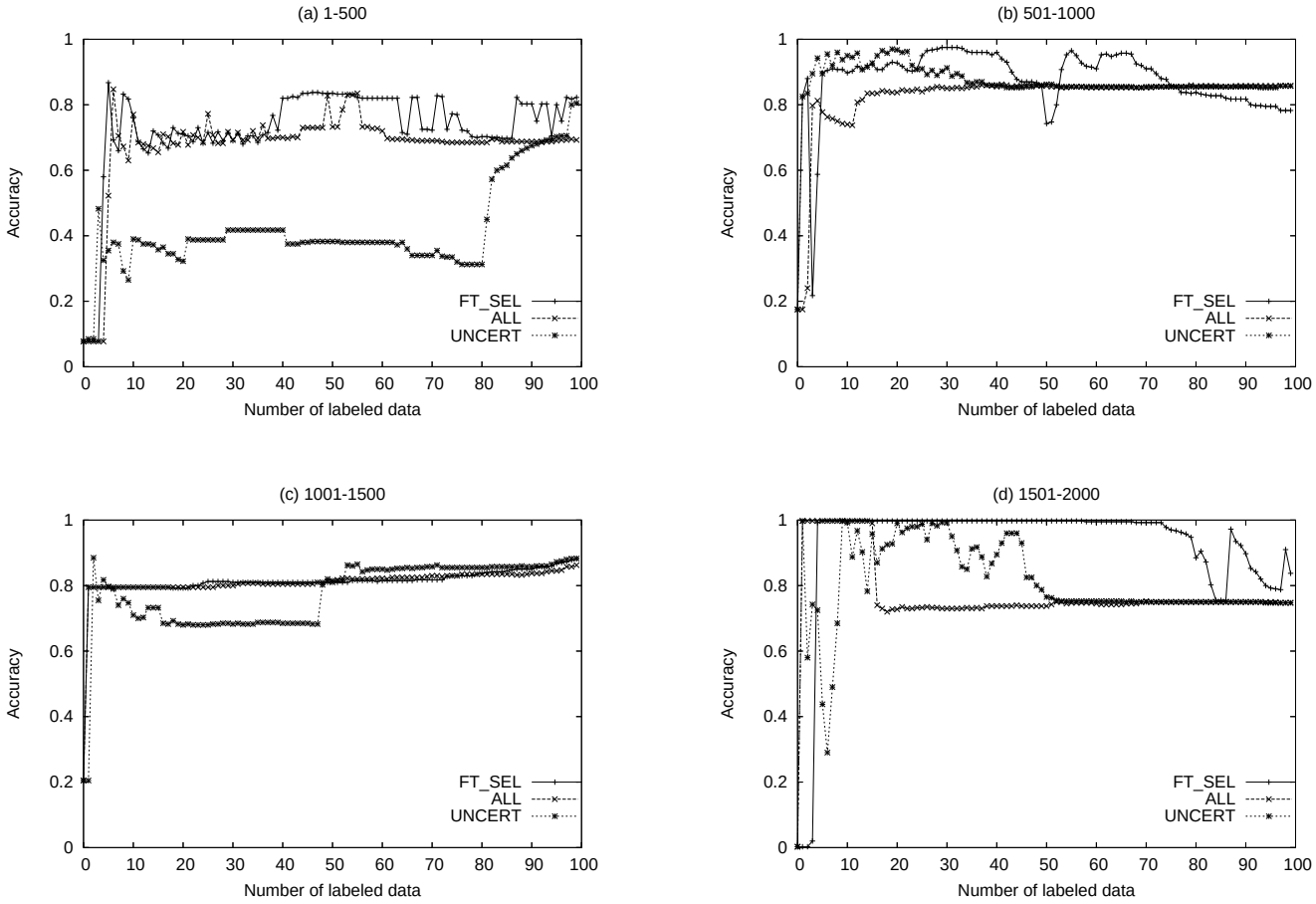

Fig. 3. Results of experiments

IV. EXPERIMENTS

This section shows the experiments of spam filtering to compare our proposed method with two others. In the experiments, we used an e-mail collection that is used in the TREC Spam Filtering Track [10]. This data set contains about 90,000 e-mails that are sorted by date. Since this data is too large to simulate active learning, we select 4 data sets. They are labeled **(a) 1-500**, **(b) 501-100**, **(c) 1001-1500**, **(d) 1501-2000**, respectively. Each data set consists of 500 data in which first 100 data are used for the pool of unlabeled training, the rest 400 are used for test. The process of active learning starts from no labeled training data, and then add labeled data one by one until total 100 data are added. E-mails are preprocessed by removing only symbols. Each data are represented as a bag of words.

We compared our method with other two ones.

1) **FT_SEL** : This is our proposed method. Parameters T_{conf} and T_{bal} are set to 0.9 and 0.1, respectively.
2) **ALL** : naive Bayes based active learning with sample error reduction
3) **UNCERT** : naive Bayes based active learning with uncertainty sampling

The first one is naive Bayes based active learning with sample error reduction approach that is the basis of our proposed method. Our method merge feature selection procedure to this method. This base method uses all features. The second one is also naive Bayes based active learning with uncertainty sampling. This technique selects a data that is the most difficult to classify, namely the difference between class probability.

Fig.3 shows the results on 4 data sets. On all data sets, our method shows better or comparative performance to other two methods. We summarize the characteristics of each result below.

- Data 1-500: FT_SEL and ALL are comparative during the early stage (until about 40 data are added). In the following stage, FT_SEL mostly outperforms ALL. UNCERT shows bad performance in this data set.
- Data 501-1000: UNCERT slightly outperforms FT_SEL until about 20 data are added. However from the point, FT_SEL gets better, and UNCERT decreasing to the same performance of ALL.
- Data 1001-1500: In this data set, FT_SEL and ALL show almost the same performance. UNCERT shows worse performance than other two methods. Its performance

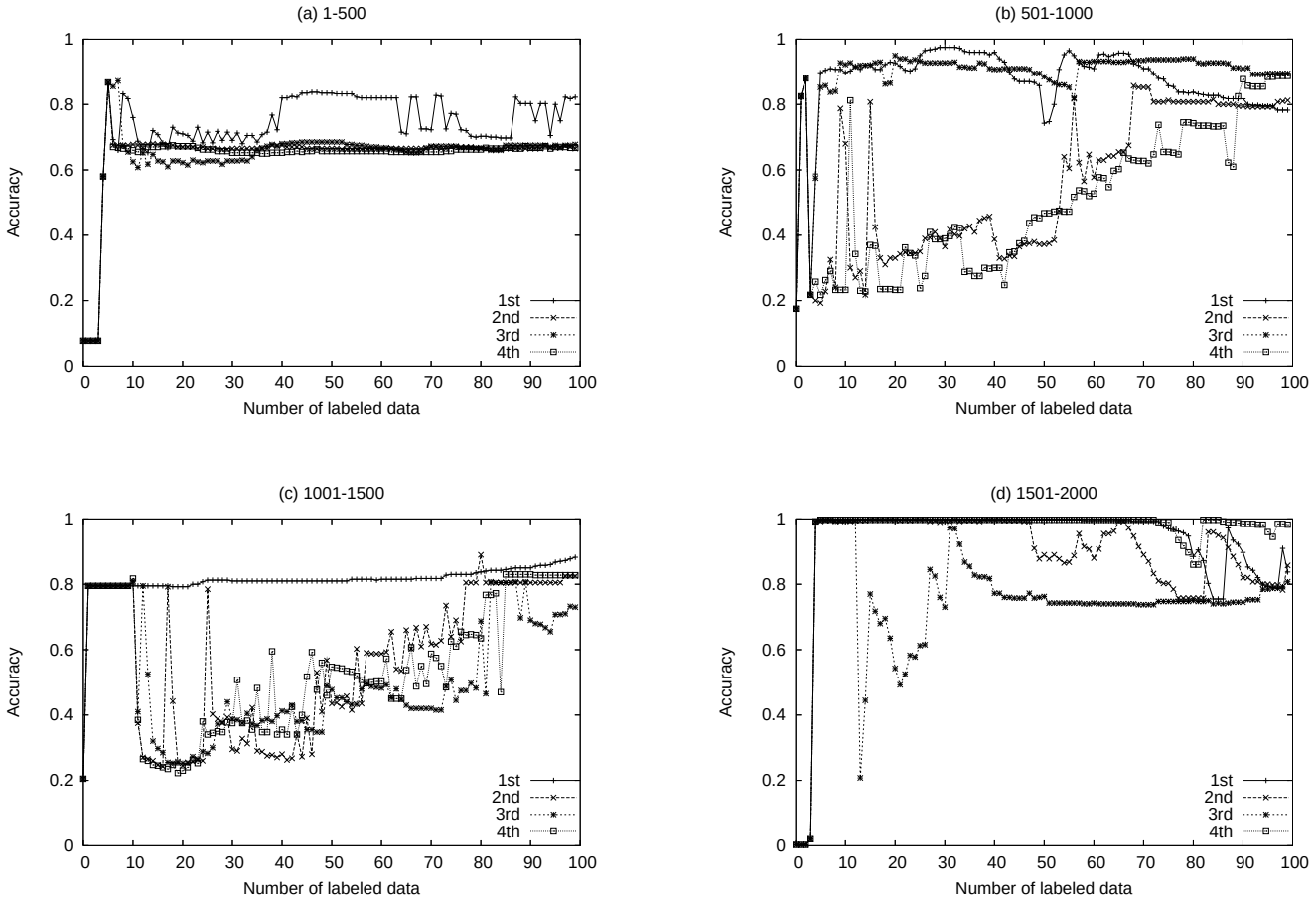

Fig. 4. Results of various gap selections

gets better around 50 data are added.

- Data 1501-2000: Three methods reach the best performance until less than 10 data are added in this data set. However only FT_SEL keeps the best performance until around 70 data are added.

V. Discussion

Our method fixes the threshold to the largest gap point for feature selection. This strategy seems to be an adhoc heuristic. Thus we check the adhocness of the strategy by changing the threshold to select features. We test the 2nd, 3rd and 4th largest gap points in our method. Fig.4 shows the results. On every data set, the largest gap point mostly performs the best. In data set (a), feature sets without the largest gap selection show almost the same performance. This results indicates that our feature selection procedure could remove unnecessary features. It is interesting that the result of the 3rd largest gap is comparable with the largest one in a data set (b). Since the feature set by 2nd largest selection shows worse performance, there are very important features between 2nd and 3rd largest gap points. In data set (c), the performance of feature sets without by the largest gap is terribly worse than the largest one. Feature selection may not be necessary in this data set. Finally the feature set by the 3rd largest gap shows worse performance than other ones. It is interesting that the feature set by the 4th largest gap outperforms others. Larger feature sets may work well in the stage i which there are a lot of data with true label. According to these results, feature selection by the largest gap works relatively well.

VI. Conclusions

In this paper we propose a feature identification approach with active learning. Our approach iteratively apply information gain based feature selection method through the process of active learning. Since our task is focused on spam filtering, we use naive Bayes as basic classifier in our approach. Data selection is conducted by the sample error reduction approach that estimates the errors of unlabeled training data by temporally updated conditional class probabilities. Then we pointed out the problem when applying the information gain based feature selection method through active learning process. For the first problem that the lack of labeled training data, our solution is to attach pseudo labels estimated by the temporal conditional class probabilities. The solution for the second problem of how to define the number of features to use is to set a threshold by finding the largest gap among

information gain scores. Experimental results shows that our proposed method shows good performance compared with an approach without feature selection and a approach based on uncertainty sampling. According to additional experiments, feature selection by the largest gap point work relatively works well irrespectively to data sets.

REFERENCES

[1] Sophos's report: http://www.sophos.com/pressoffice/news/articles/2008/07/dirtydozjul08.html, 2008.

[2] D. Sculley, Online Active Learning Methods for Fast Label-Efficient Spam Filtering, In *Fourth Conference on Email and Anti-Spam*, 2007.

[3] S. Tong, D. Koller, Support vector machine active learning with applications to text classification. In *Proceedings of the Seventeenth International Conference on Machine Learning* , 2001.

[4] M. Lindenbaum, S. Markovitch, D. Rusakov, Selective sampling for nearest neighbor classifiers, In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pp.366-371, 1999.

[5] A. McCallum, K. Nigam, Employing EM and pool-based active learning for text classification, In *Proceegings of the Fifteenth International Conference on Machine Learning*, pp. 359-367, 1998.

[6] D. Lewis, W. Gale, A sequential algorithm for training text classifiers, In *Proceedings of the International ACM-SIGIR Conference on Research and Development in Information Retrieval*, pp. 3-12, 1994.

[7] N. Roy and A. McCallum, Toward Optimal Active Learning through Sampling Estimation of Error Reduction, In *Proceedings of the Eighteenth International Conference on Machine Learning*, pp.441-448, 2001.

[8] Y. Yang and J. O. Pederson, Feature selection in statistical learning of text categorization, In *Proceedings of the Fourteenth International Conference on Machine Learning*, pp.412-420, 1997.

[9] A. McCallum, K. Nigam, A comparisonof event models for naive Bayes text classification, In *Proceedings of AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[10] G. Cormack and T. Lynam, TREC 2005 Spam Track Overview, In *Proceedings of the Fourteenth Text Retrieval Conference*, 2005.