Effective Multi-Label Active Learning for Text Classification

Bishan Yang[†], Jian-Tao Sun[‡], Tengjiao Wang[†], Zheng Chen[‡]
[†]Key Laboratory of High Confidence Software Technologies (Peking University),
Ministry of Education, China
[†]School of Electronics Engineering and Computer Science, Peking University, Beijing, 100871 China
[‡]Microsoft Research Asia, No. 49, Zhichun Road, Beijing, 100190 China
{bishan_yang, tjwang}@pku.edu.cn, {jtsun, zhengc@microsoft.com}

ABSTRACT

Labeling text data is quite time-consuming but essential for automatic text classification. Especially, manually creating multiple labels for each document may become impractical when a very large amount of data is needed for training multi-label text classifiers. To minimize the human-labeling efforts, we propose a novel multi-label active learning approach which can reduce the required labeled data without sacrificing the classification accuracy. Traditional active learning algorithms can only handle single-label problems, that is, each data is restricted to have one label. Our approach takes into account the multi-label information, and aims to label data which can optimize the expected loss reduction. Specifically, the model loss is approximated by the size of version space, and we optimize the reduction rate of the size of version space with Support Vector Machines (SVM). Furthermore, we design an effective method to predict possible labels for each unlabeled data point, and approximate the expected loss by summing up losses on all labels according to the most confident result of label prediction. Experiments on seven real-world data sets (all are publicly available) demonstrate that our approach can obtain promising classification result with much fewer labeled data than state-of-the-art methods.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval; I.5.2 [**Design Methodology**]: Classifier Design and Evaluation

General Terms

Algorithms, Performance, Experimentation

Keywords

Active Learning, Text Classification, Multi-label Classification, Support Vector Machines

KDD'09, June 28–July 1, 2009, Paris, France.
Copyright 2009 ACM 978-1-60558-495-9/09/06 ...$5.00.

1. INTRODUCTION

As text data becomes a major information source in our daily life, many research efforts have been conducted in text classification to better organize text data, in applications like document filtering, email classification, Web search, etc. In particular, multi-label text classification problems have received considerable attention, since many text classification tasks are multi-labeled, i.e., each document can belong to more than one category. Take news classification as an example, one news article talking about the effect of Olympic games on tourism industry might belong to the following topic categories: *sports, economy* and *travel.*

In the literature, supervised learning algorithms are widely used in text classification. It requires a sufficient amount of labeled data for training a high quality model. However, labeling is usually a time-consuming and expensive process done by domain experts. Active learning is an approach to reduce the labeling cost. The active learner iteratively selects a sample of data to be labeled based on some selection strategies suggesting that the data most deserves to be labeled. Thus it can achieve comparable performance with supervised learners while using much less labeled data. Active learning is particularly important for the multi-label text classification task. The reason is that, in the single-label case, a human judge can stop labeling an instance once its category is identified. But in the multi-label case, human judges need to decide all possible categories for each instance. Thus the effort of assigning labels for multi-label data is much larger than for the single-label data.

Despite the value and significance of this problem, there is very limited research on multi-label active learning. Most of the active learning research focuses on single-labeled classification problem [9, 20, 13, 21]. The sample selection strategy strictly follows the assumption that each instance has only one label. Its weakness in multi-label classification can be explained by the following example. Suppose there are three categories c_1, c_2, c_3 in the multi-label classification task. The popular one-versus-all technique [3] is used and the classification probabilities on all possible classes are given. Assume the probabilities on instance x_1 are [c_1:0.8, c_2:0.5, c_3:0.1] and on x_2 is [c_1:0.7, c_2:0.1,c_3:0.1]. x_1 actually has two labels c_1 and c_2, and x_2 has one label c_1. It can be found that correctly predicting labels for x_1 is harder than x_2. However, if we assume each instance only has one label and take the most uncertainty strategy, x_2 would be considered to be harder to classify, since the probability score on the predicted label c_2 is 0.7, which is lower than that of x_2 0.8. Thus considering multi-label information in the sample selection strategy is very important.

In this paper, we propose a novel multi-label active learning approach for text classification. The sample selection strategy aims to label data which can help maximize the reduction rate of the expected model loss. To measure the loss reduction, we use Support Vector Machines (SVM) in terms of version space [20] due to the effectiveness of SVM active learning on text classification. In the original work, the loss is modeled for single-label case, and here we extend it to multi-label case. We also propose an effective method to predict labels for multi-label data. The expected loss is approximated with the loss associated with the most confident result of label prediction. We will show that a proper label prediction method is critical in measuring loss for multi-label data.

We empirically evaluate the effectiveness of the proposed approach using a number of real-world data sets that are publicly available. The results demonstrate that our method is superior to the state-of-the-art active learning algorithms for multi-label text classification, and can significantly reduce the demand of labeled data while maintaining promising classification results.

The remainder of this paper is organized as follows. Section 2 discusses related work. Section 3 presents the definition of multi-label text classification problem. Section 4 introduces our SVM-based active learner, including the loss optimization framework and the sample selection strategy. Section 5 shows experimental results of our algorithm on seven real-world data sets compared with other baseline methods. Section 6 presents conclusions and future work.

2. RELATED WORK

Active learning on text classification has been well researched. Based on the adopted sample selection strategy, they can be grouped into three types: 1) Uncertainty sampling [9, 13]. The active learner iteratively labels the unlabeled data on which the current hypothesis is most uncertain. 2) Expected-error reduction [2, 17, 21]. The strategy aims to label data to minimize the expected error on the unlabeled data. Usually it requires expensive computational effort on estimating the expected error, since each of the unlabeled data needs to be evaluated. 3) Committee-based active learner. It has the similar idea with uncertainty sampling strategy. The active learner selects data to be labeled that have largest disagreement among several committee members(classifiers) from the version space. The work of query by committee [18] is the first algorithm of this kind. In [20], the idea is extended to Support Vector Machine active learning, and it models the reduction of version space size with SVM.

However, most of the previous research targets single-label classification problems. The sample selection strategy evaluates each unlabeled data by assuming it has only one label. For instance, the uncertainty sampling strategy will focus on measuring the confidence of the most likely class, and the error reduction strategy will estimate the expected error for just one class. Thus these strategies can not be directly applied in multi-label text classification.

There is very limited research on multi-label active learning. The research work of [8] is the one most related to our paper with respect to the studied problem. It decomposes the multi-label classification problem to several binary ones using one-versus-all approach, and selects data examples to minimize the smallest SVM margin among all binary classification problems. The approach does not consider the multi-label information, and treats the data as the same as multi-class data. In [11], an SVM active learning method was proposed for multi-label image classification. It selects unlabeled data which has the maximum mean loss value over the predicted classes. The multi-label classification problem is also viewed as several binary classification tasks. A threshold of loss value is estimated for each binary classifier, and then used to decide the predicted classes for unlabeled data. According to our experiments, this threshold cutting method is poor on the text classification data sets we used. Most of the time, they can not output any predicted labels. Recently, [15, 16] developed a two-dimensional active learning algorithm for image classification, which selects sample-label pairs to minimize the Bayesian classification error bound. It is reasonable to label picture-category pairs since judging a picture's label is very efficient. However, this method is not realistic for text classification task. Because it will introduce much additional cost if a document is read several times. Obviously, the cost of reading a document and judging its label is much bigger than that of a picture.

3. PROBLEM DEFINITION

Multi-label text classification is the task of automatically classifying text documents into a subset of predefined classes. Denote training examples as $\mathbf{x_1}, ..., \mathbf{x_n}$ and the k classes as $1, ..., k$. We represent the label set of $\mathbf{x_i}$ by a binary vector $\mathbf{y_i} = [y_i^1, ..., y_i^k], y_i^j \in \{-1, +1\}$, where $y_i^j = 1$ if $\mathbf{x_i}$ belongs to class j, otherwise $y_i^j = -1$. Denote the set of all possible class combinations as $\mathcal{Y} = \{-1, 1\}^k$. The multi-label classifier can be expressed as a decision function $f : \mathbf{X} \to \mathcal{Y}$.

In our active learning study, we consider SVM as the basic multi-label classifier, since SVM has met with significant success on text classification tasks [6, 22]. Usually, multi-label SVM adopts the one-versus-all approach, which trains a separate binary classifier for each possible class against the rest of classes, and combines the output of all the binary classifiers to determine the final labels of the given data. In binary classification, SVM tries to find the hyperplane that can separate the training data by a maximal margin. A *binary classifier* is of the form $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x}$. Denote f_i as the classifier with target class i. Given a test instance $\mathbf{x}'$, if $f_i(\mathbf{x}') > 0$, then $\mathbf{x}'$ belongs to class i, otherwise, the labels of $\mathbf{x}'$ will not include class i.

In this paper, we adopt the pool-based active learning approach which is the most popular paradigm of active learning in the literature. Assume we are given a pool of partially labeled data. Denote the data with labels by D_l, which is typically small in size, and the remaining data without labels by D_u. At the beginning, a classifier is trained using the initial labeled set D_l. Based on this classifier, the learner selects a sample from D_u and queries for its true labels. Then the newly labeled data is incorporated into D_l. The training and labeling process runs iteratively after a certain number of iterations or when the classifier reaches a sufficient accuracy.

The key issue of active learning is how to select the most informative data examples to be labeled, which is also called sample selection strategy. So, the research problem studied in this work can be described as follows: in order to train a reliable multi-label text classifier, implement a sample selection strategy which can reduce the human labeling cost as much as possible.

4. SVM-BASED ACTIVE LEARNING FOR MULTI-LABEL TEXT CLASSIFICATION

In this section, we will first introduce the optimization framework for multi-label active learning. Next we will describe our sample selection strategy with multi-label SVM.

4.1 Optimal Active Learning Framework

The optimization goal of our multi-label active learner is to label data which can contribute the largest reduction of the expected loss.

Let $P(\mathbf{x})$ be the input distribution. Denote the multi-label prediction function given training set D_l as f_{D_l}. The predicted label set $\mathbf{x}$ is $f_{D_l}(\mathbf{x})$. Suppose the true label set of $\mathbf{x}$ is $\mathbf{y}$, then the estimated loss on $\mathbf{x}$ can be written as $L(f_{D_l}(\mathbf{x}), \mathbf{y})$ (we will simplify it as $L(f_{D_l})$ in the following part), and the expected loss of the learner can be expressed as follows:

$$\widehat{\sigma_{D_l}} = \int_{\mathbf{x}} (\sum_{\mathbf{y}\in\mathcal{Y}} L(f_{D_l}) P(\mathbf{y}|\mathbf{x})) P(\mathbf{x}) d\mathbf{x} \tag{1}$$

As it is rather difficult to estimate $P(\mathbf{x})$ directly, a practical way to estimate $\widehat{\sigma_{D_l}}$ is to measure it over all the examples in D_u, as D_u is usually very large in size. Therefore we have

$$\widehat{\sigma_{D_l}} = \frac{1}{|D_u|} \sum_{\mathbf{x}\in D_u} \sum_{\mathbf{y}\in\mathcal{Y}} L(f_{D_l}) P(\mathbf{y}|\mathbf{x}) \tag{2}$$

The active learner will evaluate each possible set of unlabeled data D_s to find the optimal set and query for its labels. Then the newly labeled data will be incorporated to the training set. Let $D_l' = D_l + D_s$, and the expected loss for the classifier trained on D_l' as $\widehat{\sigma_{D_l'}}$. The optimization problem is to find the optimal query set D_s^*, which once added, will generate the largest reduction on expected loss.

$$\begin{aligned} D_s^* &= \arg\max_{D_s} (\widehat{\sigma_{D_l}} - \widehat{\sigma_{D_l'}}) \\ &= \arg\max_{D_s} (\sum_{\mathbf{x}\in D_u} \sum_{\mathbf{y}\in\mathcal{Y}} (L(f_{D_l}) - L(f_{D_l'})) P(\mathbf{y}|\mathbf{x})) \end{aligned} \tag{3}$$

As in [1], we assume that any x in $D_u - D_s$ has equal impact on the learner trained from D_l and D_l'. Then we will have

$$D_s^* = \arg\max_{D_s} (\sum_{\mathbf{x}\in D_s} \sum_{\mathbf{y}\in\mathcal{Y}} (L(f_{D_l}) - L(f_{D_l'})) P(\mathbf{y}|\mathbf{x})) \tag{4}$$

4.2 Sample Selection Strategy with Multi-label SVM

According to Equation 4, the optimization problem can be divided into two parts: how to measure the loss reduction of the multi-label classifier and how to provide a good probability estimation for the conditional probability $p(\mathbf{y}|\mathbf{x})$. We will address these two issues respectively in the following subsections.

4.2.1 Estimate Loss Reduction

As discussed in Section 3, we use SVM for the base binary classifier, and decompose the multi-label problem to 1-vs-all subproblems in active learning. By decomposing the classifier into several binary ones, the overall loss of the classifier can be measured by gathering the loss of all binary classifiers.

$$L(f) = \sum_{i=1}^{k} l(f_i), \tag{5}$$

where $l(f_i)$ is the loss on binary classifier f_i. So the problem becomes how to estimate the model loss of each binary classifier. As suggested by S. Tong et al. [20], we measure the model loss by the size of version space of a binary SVM. According to [20], the version space of SVM can be defined as follows:

$$V = \{\mathbf{w} \in W \mid \|\mathbf{w}\| = 1, y_i(\mathbf{w}\cdot\mathbf{x_i}) > 0, i = 1, ..., n\} \tag{6}$$

where W denotes the parameter space. The size of a version space is defined as the surface area of the hypersphere $\|\mathbf{w}\| = 1$ in W.

Based on the work in [20], we can use SVM margin as the measure of the version space size. When a new labeled example is added, we can approximate the new version space size by computing the SVM margin of the updated classifier. However, it is too expensive in computation when each data in the unlabeled pool is evaluated. To make it more practical, we apply the heuristics idea in [19] to simplify the approximation by mapping the SVM margin of the current classifier to the size of the new version space.

In multi-label settings, denote $V^i_{D_l}$ as the size of version space of the binary classifier $f^i_{D_l}$ associated with target class i and learnt from labeled data D_l. After adding point $(\mathbf{x}, y^i)$, where $y^i \in \{-1, +1\}$ is the true label for data $\mathbf{x}$ on class i, the reduction of model loss on the binary classifier $f^i_{D_l}$, can be approximated by:

$$\frac{l(f^i_{D_l+(\mathbf{x},y^i)})}{l(f^i_{D_l})} \approx \frac{V^i_{D_l+(\mathbf{x},y^i)}}{V^i_{D_l}} \approx \frac{1 + y^i f^i_{D_l}(\mathbf{x})}{2} \tag{7}$$

Then the loss reduction part in Equation 4 can be re-written by:

$$\begin{aligned} L(f_{D_l}) - L(f_{D_l'}) &= \sum_{i=1}^{k} (l(f^i_{D_l}) - l(f^i_{D_l'})) \\ &= \sum_{i=1}^{k} (l(f^i_{D_l}) \cdot (1 - \frac{l(f^i_{D_l'})}{l(f^i_{D_l})})) \\ &\propto \sum_{i=1}^{k} (\frac{1 - y^i f^i_{D_l}(\mathbf{x})}{2}) \end{aligned} \tag{8}$$

Note that $l(f^i_{D_l})$ has nothing to do with the selected unlabeled example $\mathbf{x}$, so we can focus on optimizing the reduction rate.

Intuitively, the idea of the above estimation can be explained as follows. Consider an unlabeled data example $\mathbf{x}$, if $\mathbf{x}$ can be correctly predicted by the current classifier f, then the smaller the value of $|f(\mathbf{x})|$ is, the more uncertain the classifier is on $\mathbf{x}$, and $\mathbf{x}$ deserves more to be labeled. This is consistent with the result of the above measure, since $\mathbf{x}$ will contribute more in reducing the size of the version space. On the other hand, if the classifier provides wrong prediction result for $\mathbf{x}$, then the larger $|f(\mathbf{x})|$ is, the more mistake the classifier will make, and in another view, adding $\mathbf{x}$ will greatly help reduce the size of the version space.

4.2.2 Label Prediction

Now we come to the issue of estimating the conditional probability $p(\mathbf{y}|\mathbf{x})$, $\mathbf{y} \in \mathcal{Y}$. Note that for k labels, there are 2^k possible label combinations. It is intractable for active learner to provide estimation on all these possibilities. Particularly, it will become harder when the training data is quite limited, which is common in active learning. To simplify the estimation, we approximate the expected loss function with the loss function on the most possible label combination. It implies that the loss can be expected to have large reduction since the most confident labeling will be most likely to be correct. Thus the problem becomes how to produce better label prediction on the unlabeled data. We propose a novel prediction approach to address this problem. Instead of directly estimating the possible labels for each data, we first try to decide the possible label number each data may have, and then determine the final labels based on the probability on each label obtained by the corresponding binary classifier.

Suppose there are k labels. Using the one-versus-all approach, we can have k binary classifiers. Given data $\mathbf{x}$, denote $p(y^i = 1|\mathbf{x})$ as the probability of $\mathbf{x}$ belonging to class i. We can obtain k classification probabilities on $\mathbf{x}$ produced by the k binary classifiers. Sort these k probabilities in decreasing order. If $\mathbf{x}$ actually has m labels, the first m probabilities are expected to be large while the other $k - m$ probabilities are expected to be small. Based on this assumption, we want to predict the number of labels for each data based on the probabilities output by the binary classifiers.

Specifically, we predict the number of labels by tackling a multi-class classification problem. Logistic regression (LR) algorithm is used to train a predictive model. For k labels, there are k possible number of labelss. So we have k classes in the multi-class classification problem. Before LR is used, we transform the decision output on the training data to classification probabilities. Here, we use the sigmoid function [12] to transform the SVM output to probability values. For a data example $\mathbf{x}$, we have

$$p(y^i = 1|\mathbf{x}) = \frac{1}{1 + exp(Af_i(\mathbf{x}) + B)}$$

where f_i is the binary SVM classifier associated with class i, A and B are scalar values estimated according to maximum likelihood criteria.

The process of predicting number of labels can be described as follows:

1. Use the SVM classifier to assign classification probabilities for all data examples.

2. For each instance $\mathbf{x}$, normalize the classification probabilities $p(y^1 = 1|\mathbf{x}), ..., p(y^k = 1|\mathbf{x})$ to make $\sum_{i=1}^{k} p(y^i = 1|\mathbf{x}) = 1$. Sort them in decreasing order and obtain $q_1(\mathbf{x}), ..., q_k(\mathbf{x})$.

3. Train logistic regression classifier. For each training data $\mathbf{x}$, present $[1 : q_1(\mathbf{x}), 2 : q_2(\mathbf{x}), ..., k : q_k(\mathbf{x})]$ as the training features for LR model. The number of labels of $\mathbf{x}$ is used as the new category to train a multi-class classifier.

4. For each data in the unlabeled pool, apply the LR classifier to predict the probabilities of having different number of labels, and output the label with the largest probability to be the predicted number of labels for the data.

Suppose the most possible number of labels for data $\mathbf{x}$ is m, and i_1, ..., i_m are the m classes with the largest probabilities produced by the binary SVM classifiers. Then the predicted label vector $\widehat{\mathbf{y}}$ can be represented by the binary vector $[\widehat{y}^{i_1} = 1, .., \widehat{y}^{i_j} = 1, \widehat{y}^{i_{j+1}} = -1, ..., \widehat{y}^{i_k} = -1]$. We call this approach $LR - based$ label prediction.

By incorporating the predicted label vector into the expected loss estimation, we obtain our data selection strategy, Maximum loss reduction with Maximal Confidence(MMC). It can be written as

$$D_s^* = \arg\max_{D_s}\Big(\sum_{\mathbf{x} \in D_s} \sum_{i=1}^{k} \Big(\frac{1 - \widehat{y}^i f_i(\mathbf{x})}{2}\Big)\Big), \tag{9}$$

Based on the above discussion, the proposed active learning algorithm is described in Algorithm 1.

Algorithm 1 Multi-label Active Learning

```
Input: Labeled set $D_l$
Unlabeled set $D_u$
Number of iterations $T$
Number of selected examples per iteration $S$

for $t = 1$ to $T$ do
    Train a multi-label SVM classifier $f$ based on training data $D_l$
    for each instance $\mathbf{x}$ in $D_u$ do
        Predict its label vector $\widehat{\mathbf{y}}$ using the LR-based prediction method described in Section 4.2.2.
        Calculate the expected loss reduction with the most confident label vector $\widehat{\mathbf{y}}$, $score(\mathbf{x}) = \sum_{i=1}^{k}(\frac{1-\widehat{y}^i f_i(\mathbf{x})}{2})$
    Sort $score(\mathbf{x})$ in decreasing order for all $\mathbf{x}$ in $D_u$
    Select a set of $S$ examples $D_s^*$ with the largest scores, and update the training set $D_l \leftarrow D_l + D_s^*$
Train the multi-label learner $\ell$ with $D_l$
```

5. EXPERIMENTS

In this section, we will evaluate our proposed multi-label active learning approach for multi-label text classification task on seven real-world data sets, comparing with the state-of-the-art active learning approaches.

5.1 Data Sets and Experiment Settings

The first data set we used is RCV1-V2 [10] text data set, which has been widely used as a benchmark data set to evaluate text classification algorithms. It contains Reuters newswire stories which are organized by three different category sets: Topics, Industries, and Regions. We considered the Topics category set in our experiments. Each document is assigned with at least one related topic category. The data used in our work can be downloaded from the web[1]. It is a subset of RCV1-V2 data used in [4] and contains 3,000 documents falling into 101 categories.

The other 6 data sets are web pages collected through the hyperlinks from Yahoo!'s top directory (www.yahoo.com).

[1] http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/multilabel.html#rcv1v2 (topics; subsets)

Each data set is associated with one of Yahoo!'s top categories, and each page is labeled with one or more second level sub-categories. The 6 data sets used in our experiments are: Arts&Humanities, Business&Economy, Computers&Internet, Education, Entertainment, and Health. They can be downloaded from the web[2]. They are also used in [14, 7] to evaluate multi-label text classification algorithms.

The details of all the 7 data sets are given in Table 1[7]. "#Inst" is the number of instances in each data set. "#Feat" is the feature dimension of each data set. In our experiments, we use words to represent each document. "#Label" is the number of labels. The "Label size percentage" gives the percentage of instances with different number of labels.

On all data sets, the documents are transformed to vectors with TF-IDF format, and each vector has unit modulus with L-2 length normalization. One-versus-all classification is conducted for each category and the multi-label classification problem is treated as several binary classification problems, where the documents from the category of interest are labeled as positive one (i.e. $y = 1$), and the rest of the documents are labeled as negative one (i.e. $y = -1$). SVM^{Light} package [6] is downloaded and used to train the binary classifier. Linear kernel is used due to its good performance in text classification task [5]. The penalty parameter C is set to 1.0 by default.

In our active learning experiments on each data set, we first randomly selected a small set of documents to form the initial labeled set, and left the remaining documents as the unlabeled pool. An active learning method was applied to select a given number of examples from the unlabeled pool in each iteration, and then add them to the labeled set with their labels. We performed several active learning iterations on each data set until the learner achieves sufficient accuracy. In every iteration, once the selected data being incorporated, the active learner retrained a new classifier on the expanded labeled set and then its performance was evaluated on the remaining data examples. We used Micro-Average F1 score as the evaluation measure, since it is a standard evaluation used in most previous text classification research. As defined in [22], micro-F1 score in multi-label case is given as follows

$$\frac{2\sum_{j=1}^{k}\sum_{i=1}^{n}\widehat{y}_i^j y_i^j}{\sum_{j=1}^{k}\sum_{i=1}^{n}\widehat{y}_i^j + \sum_{j=1}^{k}\sum_{i=1}^{n} y_i^j}$$

where n is the number of test data, $\mathbf{y_i}$ is the true label vector of the i-th data instance, $y_i^j = 1$ if the instance belongs to category j; otherwise $y_i^j = -1$. $\widehat{\mathbf{y}}_\mathbf{i}$ is the predicted label vector. We computed the average of micro-F1 scores for each active learning iteration based on 10 randomized experiments.

In our experiments, we will evaluate and compare four active learning methods:

- MMC. The sample selection strategy proposed in this paper.

- Random. The strategy is to randomly select data examples from the unlabeled pool.

- BinMin. This is a sample selection strategy proposed in [8], which is most related to our research work with respect to the problem studied. In this work, one-versus-all approach is used for multi-label classification, and SVM is used as the basic classifier. The optimal unlabeled example is selected according to

$$\arg\min_{x} \min_{i=1,\ldots,k} |f_i(x)|$$

where f_i is the binary classifier on the binary problem associated with class i. That is, it selects unlabeled examples with respect to the most uncertain label. As stated in Section 2, this method does not take advantages of the multi-label information. Therefore the selected example is not optimal for multi-label classification.

- Mean Max Loss(MML).This strategy is to select unlabeled data which has the maximum mean loss value over all the predicted labels [11]. For each predicted label j, the loss is measured as

$$\sum_{i=1}^{k} \max[(1 - m_{ij} f_i(x)), 0]$$

where $m_{ij} = 1$ if $i = j$, else $m_{ij} = -1$, and f_i is the binary SVM classifier on class i. The algorithm used a threshold cutting method to decide the predicted labels. However, according to our experiments on the text data sets, this method is usually unable to pick out predicted labels correctly. Thus we replace the label prediction part with our LR-based prediction method in Section 4.2.2, and focus on evaluating the effectiveness of the loss optimization.

5.2 Results and Discussions

In this section, we will present and discuss the experiment results on the RCV1-V2 data set as well as the 6 Yahoo! data sets.

Experimental Results with RCV1-V2 data set.

In the first experiment, we would like to verify whether our method of label prediction (presented in Section 4.2.2) is effective when only a small amount of training data is available, as this is very typical in active learning. Two popular prediction methods for multi-label classification are implemented for comparison purposes. In previous studies, the SCut method is widely used and proved very effective for predicting labels in multi-label classification tasks [10]. In [10], a binary classifier is first trained for each label. A threshold score is tuned for each binary classification task and then used to decide if an unseen data example belongs to the corresponding class or not. The second prediction method is simply setting the threshold score to be zero for each binary problem. If the classification score is positive, then the data belongs to this class, and vice versa. This simple method has its theoretical foundation, as when SVM is used, zero score corresponds with the classification hyperplane induced from statistical learning theory.

In order to verify the effectiveness of the LR-based method in predicting labels, we varied the number of training data from 100 to 1,000 (with 100 as step size). The corresponding micro-F1 curves for predicting labels are plot in Figure 1. We can observe that, as the number of training data varies, the LR-based method achieves substantially better

[2] http://www.kecl.ntt.co.jp/as/members/ueda/yahoo.tar.gz

Table 1: Statistics on RCV1-V2 and Yahoo! Data Sets

Data sets	#Inst	#Feat	#Label	Label size percentage(%)				
				1	2	3	4	≥ 5
RCV1-V2	3,000	47,236	101	12.3	29.5	35.7	10.8	11.7
Arts&Humanities	3,711	23,146	26	55.6	30.5	9.7	2.8	1.4
Business&Economy	5,709	21,924	30	57.6	28.8	11.1	1.7	0.8
Computers&Internet	6,269	34,096	33	69.8	18.2	7.8	3.0	1.1
Education	6,029	27,534	33	66.9	23.4	7.3	1.9	0.6
Entertainment	6,355	32,001	21	72.3	21.1	4.5	1.0	1.1
Health	4,556	30,605	32	53.2	34.0	9.5	2.4	0.9

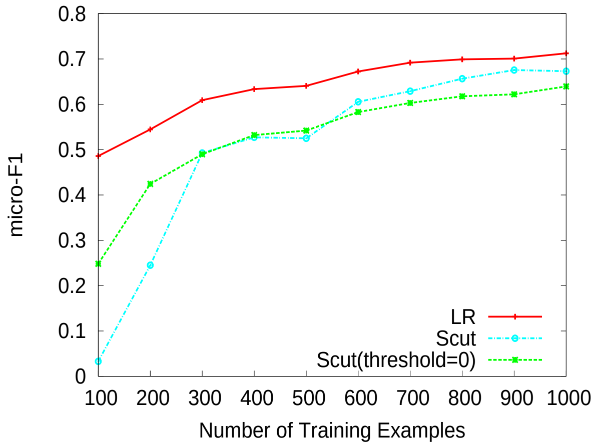


Figure 1: Comparison between label prediction methods on RCV1-V2 data set (no active learning)

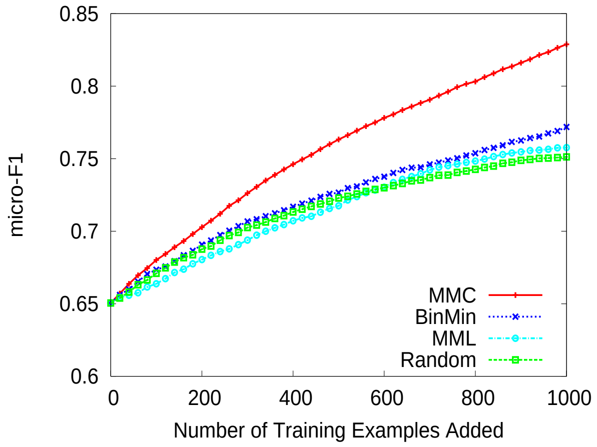


Figure 2: Micro-F1 score on RCV1-V2 data set

Table 2: Micro-F1 score at different iterations on RCV1-V2 data set(%)

K	MMC	BinMin	MML	Random
100	68.02	67.35	66.38	67.10
200	70.28	69.08	68.05	68.77
300	72.62	70.68	69.39	70.26
400	74.62	71.69	70.72	71.33
500	76.33	72.66	71.75	72.29
600	77.81	73.76	73.04	72.99
700	79.07	74.61	74.23	73.69
800	80.32	75.37	74.84	74.27
900	81.62	76.25	75.47	74.89
1000	82.88	77.19	75.77	75.12

performance than both baseline methods. When less training data is available, the advantage of LR is more obvious. This demonstrates that it is more effective in predicting labels, thus suitable for label prediction in multi-label active learning framework. We also find that the Scut method is not stable. When the training data number is small (e.g., <300), the tuned threshold score is even worse than the default zero score. When more training data is available (e.g., >600), the tuned score is better than the default zero score but not so good as LR method.

In the following we will report the active learning experiment results. We randomly selected 500 examples as the initial labeled data. Active learning was iteratively performed for 50 iterations, selecting 20 examples from the unlabeled pool each time. Figure 2 and Table 2 show the experimental results of micro-F1 scores averaging over 10 random trials. The proposed MMC strategy outperforms other baseline methods by a large margin. Surprisingly, we can see that MML performs even worse than Random at the beginning, and worse than MMC and BinMin for all cases. Since MML adopts the same label prediction approach as MMC, the observation above indicates that the loss optimization approach used in MML is not effective. Instead, our approach optimizes the loss reduction rate over all labels based on the most confident label vector, and it can successfully pick out useful data examples to be labeled. We can also find that BinMin strategy is only slightly better than Random, while our proposed method outperforms all baseline methods much more significantly. This is because the BinMin strategy does not take advantage of the multi-label information, but equally deals with the loss of the correctly predicted label and that of the wrongly predicted label, while our approach effectively estimates possible labels for each instance and incorporates the multi-label information to optimize the expected loss reduction.

Table 2 shows the performance results with the number of training samples added. We can find that as the number of selected data increases, the improvement becomes more and more significant. For example, when 1,000 examples are added, the micro-F1 score of our method achieves 82.88%, while that of BinMin, MML and Random are 77.19%, 75.77% and 75.12% respectively. We can find that MMC achieves the similar performance with BinMin by using about 600 se-

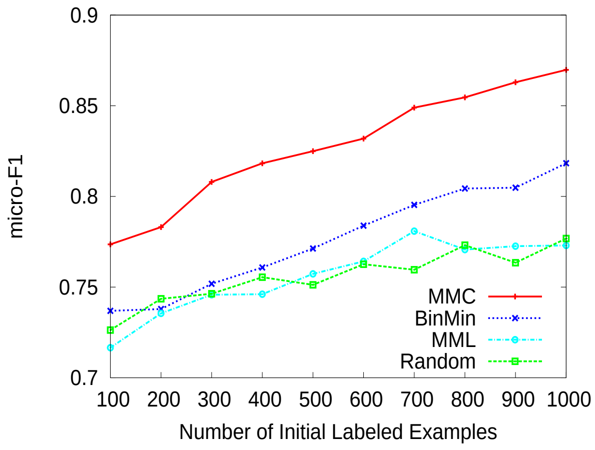


Figure 3: Micro-F1 score on RCV1-V2 data set after adding 1000 examples

lected examples, while BinMin needs to select 1,000 examples. It indicates that MMC can save about 40% labeling effort compared with BinMin.

In order to investigate if our MMC algorithm is sensitive to the size of initial labeled data set, we varied the number of initial training data from 100 to 1,000, with 100 as step size. For each fixed initial labeled set, we applied active learning and selected 20 examples at each iteration. Then we compared the performance of the final classifier after 50 active learning iterations. Figure 3 presents the micro-F1 scores of final classifiers with the size of initial training data set. We can see that our proposed MMC algorithm consistently outperforms all other methods when the initial training data set varies in size. The consistent improvement indicates that our MMC strategy is robust with different size of the initial labeled data set.

We also varied the sampling size per run and investigated its impact on the performance of the active learner. In this experiment, we started with 500 training examples and stopped after 1,000 examples are added. The sampling size S was set to 1, 20, 50, 100 and 200. The results of MMC with various sampling size are depicted in Figure 4. We can see that generally the performance improves as the sampling size decreases. A possible explanation is that having more chances to query labels enables the learner make better evaluation on unlabeled examples, and choose more informative examples to be labeled.

Experimental Results with Yahoo! data sets.

The following experiments are conducted with the 6 Yahoo! data sets. On each data set, we randomly selected 500 data instances as the initial training data, and set the sampling size in each active learning run to 20. The learning process was repeated for 50 rounds. The active learning results were averaged over 10 random trials. Fig 5 presents the performance of all active learners with the number of training data added. We can observe that our proposed method MMC consistently outperforms other baseline methods on all six data sets. The most noticeable case is the Computers&Internet data set, where the BinMin method is unable to improve the micro-F1 measurement than Random and MML presents similar performance with Random. However, MMC achieves substantially better performance. It can be observed that MMC only requires labeling 200 examples to achieve the similar performance with MML and Random which require labeling about 7,00 and 1,000 examples respectively. We can also see that MML has worse or similar performance compared with Random on five data sets. This casts doubt on the effectiveness of the optimization framework which MML takes to maximize mean loss over the predicted labels. The poor performance of BinMin underscores the importance of considering multi-label information when evaluating unlabeled examples. The promising results of MMC confirm that the proposed method can provide proper evaluation on the unlabeled data examples, and select the informative ones which can help enhance the learner more effectively. Table 3 summarizes the classification results measured by micro-F1 after 50 active learning iterations on the six Yahoo! data sets. It shows that the proposed MMC method significantly improves the micro-F1 measurement over all other baseline methods for all six data sets. When BinMin and MML only makes slight improvement, the improvement of MMC is much more significant.

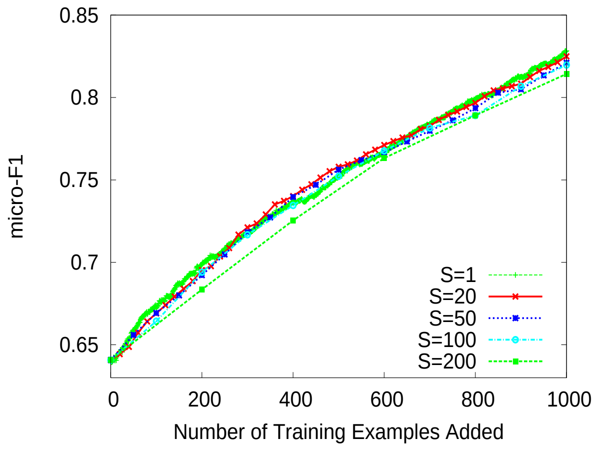


Figure 4: Micro-F1 score of MMC on RCV1-V2 data set with different sampling sizes per run

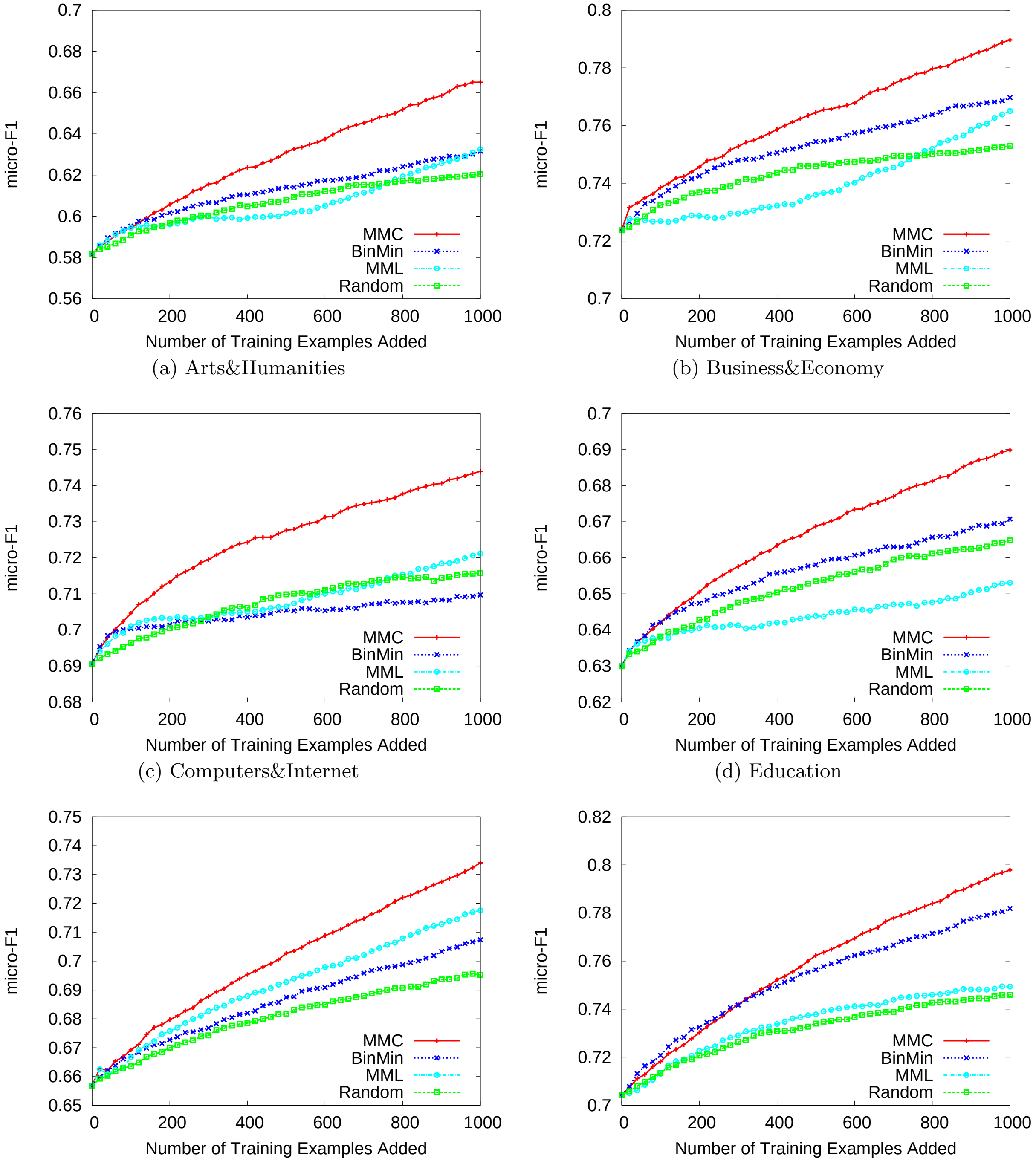
Table 3: Micro-F1 score on the Yahoo! data sets with 1,000 training samples added (%)

Data sets	MMC	BinMin	MML	Random
Arts&Humanities	66.50	63.16	63.25	62.05
Business&Economy	78.97	76.97	76.50	75.29
Computers&Internet	74.40	70.97	72.12	71.58
Education	68.99	67.07	65.31	66.48
Entertainment	73.40	70.74	71.76	69.52
Health	79.78	78.18	74.94	74.60

From the above experiments, we can observe that MMC provides promising performance on diverse data sets. This indicates that it is more effective and robust for training multi-label text classifier than the state-of-the-art active learning methods.

6. CONCLUSIONS

In this paper, we try to address the problem of multi-label active learning for text classification. The goal is to reduce



(a) Arts&Humanities

(b) Business&Economy

(c) Computers&Internet

(d) Education

(e) Entertainment

(f) Health

Figure 5: Micro-F1 score on Yahoo! data sets

the required size of labeled data in multi-label classification while maintaining favorable accuracy performance. We propose a novel multi-label active learning algorithm with Support Vector Machines (SVM). The optimization goal is to select data to be labeled which can maximize the expected reduction in model loss. Our approach provides proper approximation on the loss reduction and the expected loss in the optimization framework. Experiments on several real-world data sets show that our proposed method outperforms the state-of-the art active learning techniques on multi-label text classification by a large margin and can significantly reduce the labeling cost.

Note that our active learning approach should evaluate each of the unlabeled data at every active learning iteration. The computation would be expensive when the size of unlabeled pool is very large and the number of categories is very big. So it would be interesting to study how to evaluate only a subset of the unlabeled pool and also be able to pick out informative data to be labeled. We plan to explore this extension in the future. Also, we will apply our method on other multi-label classification tasks, e.g., image classification.

7. REFERENCES

[1] C. Campbell, N. Cristianini, and A. J. Smola. Query learning with large margin classifiers. In *Proceedings of the 7th International Conference on Machine Learning (ICML'00)*, pages 111–118, 2000.

[2] D. A. Cohn, Z. Ghahramani, and M. I. Jordan. Active learning with statistical models. In *Advances in Neural Information Processing Systems*, volume 7, pages 705–712. The MIT Press, 1995.

[3] C. Cortes and V. Vapnik. Support vector networks. In *Machine Learning*, pages 273–297, 1995.

[4] R.-E. Fan and C.-J. Lin. A study on threshold selection for multi-label classification. *Technical Report, National Taiwan University*, 2007.

[5] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. pages 137–142. Springer Verlag, 1998.

[6] T. Joachims. *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms.* Kluwer Academic Publishers, Norwell, MA, USA, 2002.

[7] H. Kazawa, T. Izumitani, H. Taira, and E. Maeda. Maximal margin labeling for multi-topic text categorization. In *Advances in Neural Information Processing Systems (NIPS'05)*, pages 649–656, 2005.

[8] K. Brinker. *On Active Learning in Multi-label Classification. "FromData and Information Analysis to Knowledge Engineering" of BookSeries "Studies in Classification, Data Analysis, and Knowledge Or-ganization"*, Springer, 2006. 1, 2.

[9] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval(SIGIR'94)*, pages 3–12, 1994.

[10] D. D. Lewis, Y. Yang, T. G. Rose, G. Dietterich, F. Li, and F. Li. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

[11] X. Li, L. Wang, and E. Sung. Multi-label svm active learning for image classification. In *International Conference on Image Processing*, pages 2207–2210, 2004.

[12] H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Journal of Machine Learning Research*, 68(3):267–276, 2007.

[13] T. Luo, K. Kramer, D. B. Goldgof, L. O. Hall, S. Samson, A. Remsen, and T. Hopkins. Active learning to recognize multiple types of plankton. *Journal of Machine Learning Research*, 6:589–613, 2005.

[14] N. Ueda and K. Saito. Single-shot detection of multiple categories of text using parametric mixture models. In *Proceedings of the 8th ACM SIGKDD international conference on Knowledge discovery and data mining(KDD'02)*, pages 626–631, 2002.

[15] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang. Two-dimensional active learning for image classification. *IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[16] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, and H.-J. Zhang. Two-dimensional multi-label active learning with an efficient online adaptation model for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 99(1), 2008.

[17] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the 8th International Conference on Machine Learning(ICML'01)*, pages 441–448, 2001.

[18] H. S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the 5th annual workshop on Computational learning theory(COLT'92)*, pages 287–294, 1992.

[19] S. Tong. *Active learning: Theory and Applications.* PhD thesis, Standford University, CA, 2001.

[20] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of Machine Learning Research*, 2:45–66, 2002.

[21] R. Yan, J. Yang, and A. Hauptmann. Automatically labeling video data using multi-class active learning. In *Proceedings of the 9th IEEE International Conference on Computer Vision(ICCV'03)*, page 516, 2003.

[22] Y. Yang. A study on thresholding strategies for text categorization. In *Proceedings of 24th International Conference on Research and Development in Information Retrieval(SIGIR'01)*, pages 137–145, 2001.