Chinese Short Text Categorization Based on Semi-Supervised Learning

Jie MA\(^1\)*, Zhong-Yang XIONG\(^1\), Yu-Fang ZHANG\(^1\),
Liu-Qian WANG\(^1\) and Jiang XIE\(^1\)

\(^1\)College of Computer Science, Chongqing University, Chongqing 400044, China

Email: jiema@cqu.edu.cn. Corresponding author

Keywords: Chinese text categorization, AdaBoost, The improved decision tree, Semi-supervised learning.

Abstract. Most of the text on the Internet is unlabelled with the rapid development of the Internet, and it is difficult for us to classify the unlabelled text accurately under the condition of insufficient labelled samples. Semi-supervised learning is a method, which combines the labelled samples with the unlabelled samples, can solve the problem in a better way. AdaBoost is one of the most representative algorithm of boosting algorithms, and this paper used the improved decision tree to be weak classifiers of the AdaBoost. Based on this, this paper devised a boosting algorithm which was based on semi-supervised learning and the improved decision tree. The algorithm is devoted to solving the problem of the Chinese short text categorization under the condition of insufficient labelled samples. Experiments show that the algorithm can effectively improve the performance of the Chinese short text categorization on balanced and imbalanced data sets.

Introduction

Short text has gradually become a new form of text, which is widely used in social software, network evaluation and so on. Therefore, we can categorize those large amounts of short text in order to mine useful information. Text categorization has become one of the major ways to process and organize text data, and lots of state-of-the-art supervised learning algorithms have been proposed and widely used in text categorization [1]. However, supervised learning usually needs to obtain a large number of labeled training samples for learning. Meanwhile, labeling samples requires a lot of labors, materials, financial resources, and time. On the contrary, it is very easy for people to obtain the unlabeled data, and researchers have done many experiments on the basis of using unlabeled data, and have demonstrated that it can improve learning performance to some extent [4]. Consequently, SSL (semi-supervised learning), a method which combines labeled data and unlabeled data, has attracted the attention of numerous researchers [6].

Boosting algorithm, the most representative of which is AdaBoost algorithm. It was proposed by Freund and Schapire in 1995, and has been widely and effectively applied to classification problems [11]. When researchers use AdaBoost algorithm to do classification research, they generally use the decision tree as the weak hypothesis. Each feature is regarded as a decision tree, and the judgment condition is only whether a document contains feature \(w\) [12], and the AdaBoost algorithm returns “+1” or “-1” to tackleinery categorization problems. In this case, it is possible to divide the negative samples including the feature \(w\) into the positive class, and divide the positive samples excluding the feature \(w\) into the negative class.

In order to reduce the possibility of the above, this paper puts forward a Chinese short text categorization algorithm which is based on SSL called Semi-AdaBoost. MH\(^{IDT}\) (Semi-Supervised AdaBoost MH Based on Improved Decision Tree). The algorithm is an inductive method, which learns from the labeled samples and unlabeled samples to get weak classifiers in order to predict the labels of the testing samples. The algorithm still uses decision tree as weak hypothesis, but when it selects weak hypothesis, it not only considers whether the document \(d\) contains feature \(w\) but also the similarity (cosine similarity between the document of the training set and the document consisting of features) to select the weak hypothesis. A large number of experiments show that the algorithm has a good classification effect on Chinese text classification.
The remaining of this paper is organized as follows. Section 2 discusses some related work. Section 3 describes text preprocessing and Semi-AdaBoost. MHIDT algorithm. Next, Section 4 discusses and analyzes the proposed algorithm and experimental results. Finally, section 5 draws the conclusion.

Related Work

AdaBoost algorithm, as the most representative algorithm, which was proposed by Freund and Schapire in 1995, is widely used in text classification, image recognition and other fields [11]. The text categorization algorithm which is based on boosting was proposed by RE Schapire and Y Singer [12], which laid a foundation for later researchers to study text classification using boosting algorithm, but the selection of its weak hypothesis is determined only by the presences or absences of features, and it mainly aims at English text classification. Junli Chen et al. proposed multi-label classification algorithm based on boosting, which can effectively solve the problem of Chinese text classification [14], but the algorithm does not select the best weak hypothesis in the iteration process of each round. Therefore, the final strong classifier combined by weak hypothesis has some defects. Xinhao Wang et al. proposed an algorithm to improve the effect of Chinese text classification by outlier learning, the core algorithm is mainly using traditional AdaBoost algorithm to solve the problem of Chinese text classification, so in the case of insufficient labeled data, the classification results will show the algorithm does not work well [15]. Zhan Wang et al. proposed the Chinese text classification algorithm based on boosting which can effectively solve the problem of Chinese text classification [17], but the algorithm is a supervised learning method. As is known to all, supervised learning method just uses the labeled data for training. Al-Salemi B et al. proposed an algorithm that can not only improve the effect of text categorization but also reduce the time that spent in classifying [13]. Also, the algorithm mainly aims at English text categorization, and it is a supervised learning method. Therefore, as the number of texts grows, the classification effect will greatly reduce in the case of insufficient labeled samples of training sets. In recent years, the classification algorithm based on semi-supervised learning has attracted the attention of numerous researchers. Liu C L et al. put forward a text categorization algorithm based on a semi-supervised learning and Universum[16], and universum has some effect on the classification of English texts, but I have proved that the effect of Universum on Chinese text classification is not ideal by experiments. Paul M K et al. put forward an algorithm that uses the Gaussian Mixture to improve the effect of the classification of imbalanced data sets, but in the process of each round of iteration, the selection of weak hypothesis is not optimal, so the algorithm can be further improved [19]. In conclusion, the study of the algorithm of Chinese text classification still has greater space to be improved.

Semi-AdaBoost. MHIDT Algorithm

Notation

This section describes the notations used throughout in this paper. Let \( \text{sim} \) represents the similarity between documents. The number of classes or categories is \( c \), where the class labels are \( 1, \ldots, c \). Let \( \mathcal{X} \) represents the feature space, and let \( \mathcal{Y} \) be the label set. The input data includes a few labeled samples \( \mathcal{L} \), a large number of unlabeled samples \( \mathcal{U} \), and a collection of documents \( \mathcal{X} \), which consist of selected features that are highly relevant to the category. At last, \( I \) is an indicator function.

Improved Decision Tree

AdaBoost and its variants are the most representative algorithms of the boosting, and they are widely used in various fields. In general, the weak classifier of the AdaBoost algorithm can be of any type, such as neural networks, decision trees, simple empirical rule and so on [16]. AdaBoost algorithm in the binary classification returns “+1” or “-1” to do classification. The AdaBoost.MH
algorithm was proposed by Schapire R E and Singer Y, which is a multiclass and multilabel classification algorithm [20]. The decision tree is used to be weak classifier in the algorithm namely that it can divide the entire dataset into two chunks by the fact that whether the feature appears or not in the document. Let \( c_j = h(x, l) \) represent the output of a hypothesis for label \( l \) and data \( x_i \in X_j \) where \( j \in \{0,1\} \). The weak hypothesis is shown in Eq. 1

\[
h_i(x_l, l) = \begin{cases} c_{10}, & \text{if } w \not\in x_i \\ c_{11}, & \text{if } w \in x_i \\ \end{cases}
\]

(1)

It can be seen from the equation (4), the selection of the weak hypothesis is only based on the fact that whether the feature appears or not. Obviously, under the above conditions, the weak hypothesis can divide the document containing the feature and belonging to the negative class into the positive class, and divide the documents which do not contain the feature and belong to the positive class into the negative class falsely. However, most researchers have focused on other aspects of AdaBoost improvement, and they still use the above method to select the weak hypothesis [14]-[17].

As everyone knows, the selection of weak classifier plays a vital role in the final classification in the ensemble learning. Inspired by the above, this paper proposes an algorithm based on the improved decision tree to enhance the performance of the classification.

In this paper, the whole features selected from each class are considered as a document \( x_j \), where \( x_j \in X, \) \( j \in \{1,...,c\} \), \( x_j = \{f_1, f_2, ..., f_k\} \), \( f_k \) is the feature which highly correlates with category \( c \). The document \( x_i \) in the training sets is represented as a vector \( \vec{w} = \langle w_1, ..., w_k \rangle \), and when \( w_j = 1 \), the feature \( f_k \) appears in the document \( x_i \), and when \( w_k = 0 \), the feature \( f_k \) does not appear in the document \( x_i \). The document made up of \( k \) features is naturally represented as a vector \( \vec{x}_j \) that consists of \( k \) 1. In order to reduce the possibility that divides the documents which contain the feature \( f_k \) and belong to the negative class into the positive class, or the documents that do not contain the feature \( f_k \) and belong to the positive class into the negative class, the cosine similarity between document \( x_j \) and \( x_i \) is calculated by Eq. 2 in this paper.

\[
sim(x_j, w) = \cos(x_j, w) = \frac{\vec{x}_j \cdot \vec{w}}{\| \vec{x}_j \| \cdot \| \vec{w} \|}
\]

(2)

When the document \( x_i \) does not contain the feature \( f_k \) and \( \sim(x_j, w) > thr_f \), \( c_{10} \) is selected as the weak hypothesis. When the document \( x_i \) contains the feature \( f_k \) and \( \sim(x_j, w) > thr_f \), \( c_{11} \) is selected as the weak hypothesis. Otherwise, \( c_{00} \) is selected as the weak hypothesis. The selection of weak hypothesis is just like Eq. 3

\[
h_i(x_l, l) = \begin{cases} c_{10}, & \text{if } (f_k \not\in x_i) \&\& (\sim(x_j, w) > thr_f) \\ c_{11}, & \text{if } (f_k \in x_i) \&\& (\sim(x_j, w) > thr_f) \\ c_{00}, & \text{else,} \end{cases}
\]

(3)

where \( thr_f \) and \( thr_l \) are certain thresholds.

AdaBoost\(^{IDT}\)

Through the above discussion, this paper firstly applies the improved decision tree to the AdaBoost algorithm to solve the binary classification problem. AdaBoost\(^{IDT}\) (AdaBoost Based on the Improved Decision Tree) algorithm uses the improved decision tree as a weak hypothesis to construct the weak classifier, and the final classifier returns “+1” or “-1” to do classification. Algorithm 1 shows the algorithm using the improved decision tree.
**Algorithm 1** AdaBoost\(^{\text{IDT}}\) Algorithm

**Input:** Given labeled samples \(\{(x_i, y_i), \ldots, (x_k, y_k)\}\) where \(x_i \in X\) for \(1 \leq i \leq k\) and \(y_i \in \{+1,-1\}\).

**Output:** the binary classification result.

1: Consider the labeled samples as training set.
2: Initialize \(D(i) \leftarrow \frac{1}{k}, i \leftarrow 1, \ldots, k\).
3: for \(t \leftarrow 0\) to \(T\) do
4: Train weak learner using the improved decision tree in Eq. 6 and distribution \(D_t\).
5: Get weak hypothesis \(h_t : X \rightarrow \).
6: Choose \(\alpha_t \in \).
7: Update \(D_{t+1}(i) \leftarrow \frac{D(i) \exp(-\alpha_t Y_h(x_i))}{Z_t}\), where \(Z_t\) is a normalization factor and \(D_{t+1}(i)\) is also a distribution.
8: endfor
9: Get the final classifier: \(H(x) \leftarrow \text{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right)\).

**Semi-AdaBoost. MH\(^{\text{IDT}}\)**

Owing to the AdaBoost. MH\(^{\text{IDT}}\) is a multilabel and multiclass algorithm, therefore the weight distribution of the training set is \(D_{i,l}\) rather than \(D(i)\). Meanwhile, the labels of each sample are a collection rather than a categorical value. Therefore, Eq. 4 is used to denote the label information

\[
Y[l] = \begin{cases} +1, & \text{if } l \in Y \\ -1, & \text{if } l \notin Y. \end{cases}
\] (4)

By discussing in the 3.2 section, the selection of weak hypothesis has been determined. Schapire R E and Singer Y proved that the training error bound of the final classifier of AdaBoost algorithm is \(\prod Z_i\) [20], where \(t\) is the number of iterations and \(Z_i\) is a normalization factor. The theorem indicates that we can select the appropriate weak hypothesis in each round iteration that can minimize \(Z_i\) in order to make the training error drop fastest. That inspires us to select the weak hypothesis in each iteration that can minimize the \(Z_i\) to make the training error drop fastest. Given the current distribution \(D_t\) and feature \(w\), for each possible label \(l\), for \(j \in \{0,1\}\), and for \(b \in \{0,1\}\), the weight of each sample can be calculated by Eq. 5

\[
W_{j,b} = \sum_{i=1}^{m} D_t(i,l) I\{x_i \in x_j \land (Y[l] = b)\}. \tag{5}\]

It has proved that for a given feature, the value of weak hypothesis can be calculated by Eq. 6[12]

\[
c_{j,b} = \frac{1}{2} \ln(\frac{W_{j,b}}{W_{j,-b}}), \tag{6}\]

and when \(\alpha_s = 1\), \(Z_i\) is the smallest. It can be calculated by Eq. 7

\[
Z_i = 2 \sum_{j \in \{0,1\}} \sum_{b \in \{0,1\}} \sqrt{W_{j,b} W_{j,-b}}. \tag{7}\]

Algorithm 2 shows the Semi-AdaBoost. MH\(^{\text{IDT}}\) algorithm, and the inputs include labeled samples, unlabeled samples, and number of categories \(c\). The line 4 of algorithm 2 selects the weak
hypothesis, which is determined by the improved decision tree defined in Eq. 3. After determining the weak hypothesis, Eq. 6 shows that how to calculate the value of the weak hypothesis. In order to make the final strong classifier have good classification performance, this paper selects the weak hypothesis that can minimize \( Z_t \) in each round of iterations, which can minimize the training error.

```
Algorithm 2 Semi-AdaBoost.MH^{DT}

Input: Given the number of categories \( c \), labeled samples \( \mathcal{L} = \{(x_i,y),..., (x_i,y)\} \), unlabeled sample \( \mathcal{U} = \{(x_{k+1},y_{k+1}),..., (x_{k+n},y_{k+n})\} \), where \( x_i \in \mathcal{X} \) and \( y_i \subseteq \mathcal{Y} \) for \( 1 \leq i \leq k+n \), and the test data set.

Output: the classification result.

1: repeat
2: Initialize \( D(i,l) \leftarrow 1/|\mathcal{L}| \), where \( i \leftarrow 1,...,|\mathcal{L}| \) and \( l \leftarrow 1,...,c \).
3: for \( t \leftarrow 1 \) to \( T \) do
4: Pass distribution \( D_t \) to weak learner using the improved decision tree in Eq. 6 in order to get a weak hypothesis \( h_t: \mathcal{X} \times \mathcal{Y} \rightarrow \) with the smallest \( Z_t \) defined in Eq. 10.
5: Choose \( \alpha_t \in \mathbb{R} \).
6: Update \( D_{t+1}(i,l) \leftarrow D_t(i,l) \exp(-\alpha_t Y|hl(x_i,l)) \), where \( Z_t \) is a normalization factor, and \( D_{t+1}(i,l) \) is also a distribution. \( h_t(x_i,l) \) is defined in Eq. 9 and \( i \leftarrow 1,...,|\mathcal{L}| \).
7: end for.
8: for \( i \leftarrow k+1 \) to \( k+n \) do
9: \( x_i, \text{label} \leftarrow \arg \max \sum_{l=1}^{c} \alpha_t h_t(x_i,l) \).
10: \( x_i, \text{conf} \leftarrow \max \sum_{l=1}^{c} \alpha_t h_t(x_i,l) \).
11: end for
12: for \( l \leftarrow 1 \) to \( c \) do
13: \( \text{averageConf}[l] \leftarrow \frac{\sum_{i=k+1}^{k+n} x_i, \text{conf} \times 1(x_i, \text{label} = l)}{\sum_{i=k+1}^{k+n} 1(x_i, \text{label} = l)} \).
14: for \( i \leftarrow k+1 \) to \( k+n \) do
15: if \( (x_i, \text{label} = l) \) and \( (x_i, \text{conf} \geq \text{averageConf}[l]) \) then
16: \( \mathcal{L} \leftarrow \mathcal{L} \cup x_i \)
17: \( \mathcal{U} \leftarrow \mathcal{U} - x_i \)
18: end if
19: end for
20: end for
21: Until Convergence
22: Get the final classifier: \( f(x_i,l) \leftarrow \sum_{l=1}^{c} \alpha_t h_t(x_i,l) \).
```

Through the above analysis and discussion, weak hypothesis \( c_j \) can provide additional information for each category when classifying, namely that the presences or absences of features and the similarity have effect on each category \( l \). Consequently, compute the value of weak hypothesis for each category \( l \), so the value of \( c_j \) is a vector of \( c_j \), where \( 1 \leq i \leq c \), and the highest among the vector values is the most likely category. Each weak hypothesis will provide a prediction, and the final strong classifier which linearly combines the whole weak hypothesis will
provide a comprehensive prediction. In the lines from 8 to 11 of the algorithm 2, \textit{label} and \textit{conf} represent the classification results and the confidence of the classification results respectively.

After using the final classifier to make a comprehensive prediction for unlabeled samples, each unlabeled sample has a confidence of a possible classification result. This paper uses semi-supervised learning to add unlabeled samples that meet a given condition to training set in order to achieve good classification results. Therefore, in lines from 12 to 20 this paper sets a threshold, and when the confidence level is greater than the threshold, the unlabeled samples are added to the category \( l \) so as to enrich the training set, and remove the sample from unlabeled data set.

Finally, when the algorithm converges to a certain condition, the final classifier is obtained.

**Experiment**

**Data Sets and Evaluation Indicator**

In this paper, the experimental data sets consisting of Chinese news is provided by Sogou laboratory (http://www.sogou.com/labs/resource/list_pingce.php), including finance, IT, health, sports, tourism, education, employment, culture and military news data, and the number of documents in each category is 1990, totaling 17910 documents. In Chinese text categorization, they are widely used in text categorization to evaluate system performance. NLPIR is used in word segmentation and CHI is used to do feature selection in this paper[18]. The source code of this experiment (including data preprocessing and classification algorithm) has been shared on github.

Precision, recall, and \( F_1 \) score are often used in the field of information retrieval and statistical learning to evaluate system performance. Eq. 8 defines precision, recall, and \( F_1 \) score, where \( TP \) indicates the number of true positive samples, \( TN \) the number of true negative samples, \( FP \) the number of false positive samples, and \( FN \) the number of false negative samples [22]. Eq. 9 defines the macro mean of the \( F_1 \) score, where \( c \) represents the number of categories, and \( F_i \) represents the \( F_1 \) score of the \( i \)th class.

\[
\text{Precision} = \frac{TP}{TP + FP}
\]
\[
\text{Recall} = \frac{TP}{TP + FN}
\]
\[
F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{8}
\]
\[
\text{Macro-average } F_i = \frac{\sum F_i}{c} \tag{9}
\]

**Performance Impact with the Improved Decision Tree and SSL Given Different Amount of Labeled Samples**

**Experiments on Balanced Data Sets**

In order to prove the proposed algorithm is helpful to improve classification effect, binary classification is carried out to prove the above conclusion. Firstly, two groups of experiments were conducted using AdaBoost, AdaBoost\textsuperscript{IDT} and Semi-AdaBoost\textsuperscript{IDT} algorithms on balanced data sets, finance and health, education and employment respectively. Secondly, the same algorithm is also tested on the same data set on the imbalanced data sets.

The experimental results on balanced data sets are shown in Table 1. The percentage of labeled samples is the proportion of labeled samples in the total data sets and the rest are used as a test set to test the classification effect of the algorithm. When testing the classification effect of Semi-AdaBoost\textsuperscript{IDT} algorithm, a certain number of unlabeled samples are added to the training set. From
the results of the two groupofexperiments, we can see that the AdaBoost$^{IDT}$ algorithm and the Semi-AdaBoost$^{IDT}$ algorithm can effectively improve the classification results when the percentage of labeled samples is small, and especially in the experimental results of education and employment, the AdaBoost$^{IDT}$ algorithm and the Semi-AdaBoost$^{IDT}$ algorithm are more outstanding. This is because in the condition of insufficient labeled samples the categorization mode trained by AdaBoost algorithm has some defects that only uses the presence or absence of the feature to select weak hypothesis, which affects the performance of weak classifiers in a certain extent. However, when the improved decision tree is used in AdaBoost$^{IDT}$ algorithm and Semi-AdaBoost$^{IDT}$ algorithm, the classification effect of the weak classifiers is improved, so that the final classifier is better than the AdaBoost algorithm. Meanwhile, it can be seen that the Semi-AdaBoost$^{IDT}$ algorithm is superior to the other two algorithms in the case of insufficient labeled samples in Table 2, because SSL adds the unlabeled samples meeting certain condition to training set to increase the number of training set to construct a more robust categorization model, which can improve the classification effect. With the percentage of labeled samples increasing, the advantages of Semi-AdaBoost$^{IDT}$ and AdaBoost$^{IDT}$ algorithms are gradually reducing, because the classification algorithm can use a large number of training samples to learn a good classification model. Consequently, when the classification effects of AdaBoost algorithm, Semi-AdaBoost$^{IDT}$ algorithm and AdaBoost$^{IDT}$ algorithm tend to be better with the increasing percentage of labeled samples, the narrowing among the three is decreasing.

Table 1. Experimental results on balanced data set.

<table>
<thead>
<tr>
<th>percentage of labeled samples</th>
<th>Finance vs Health</th>
<th>Education vs Employment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AdaBoost</td>
<td>AdaBoost$^{IDT}$</td>
</tr>
<tr>
<td>0.05</td>
<td>0.83479</td>
<td>0.86019</td>
</tr>
<tr>
<td>0.1</td>
<td>0.86072</td>
<td>0.88423</td>
</tr>
<tr>
<td>0.2</td>
<td>0.87812</td>
<td>0.89965</td>
</tr>
<tr>
<td>0.3</td>
<td>0.89332</td>
<td>0.91427</td>
</tr>
<tr>
<td>0.4</td>
<td>0.89831</td>
<td>0.91983</td>
</tr>
<tr>
<td>0.5</td>
<td>0.90956</td>
<td>0.92236</td>
</tr>
<tr>
<td>0.6</td>
<td>0.90886</td>
<td>0.92768</td>
</tr>
<tr>
<td>0.7</td>
<td>0.91375</td>
<td>0.93665</td>
</tr>
<tr>
<td>0.8</td>
<td>0.91074</td>
<td>0.94863</td>
</tr>
</tbody>
</table>

Experiments on Imbalanced Data Sets

In the imbalanced data set 1, the number of samples of each class in the training set is 0.65 and 0.35 respectively, 0.8 and 0.2 in the imbalanced data set 2. It can be seen that the classification effect of Semi-AdaBoost$^{IDT}$ and AdaBoost$^{IDT}$ algorithm is still better than AdaBoost algorithm in Table 2 and 3. At the same time, it can be seen that the AdaBoost$^{IDT}$ and AdaBoost algorithms are in unstable state, that is, the classification effect does not get better with the increasing number of training sets, because the number of samples of each class is no longer equal in the imbalanced data sets, which leads to some defects to categorization model. On the contrary, the Semi-AdaBoost$^{IDT}$ algorithm performs better than the other two algorithms, and it is in a relatively stable state, because SSL adds the unlabeled samples meeting certain conditions to training sets, which makes the training sets richer, and in this way classification algorithms can use enough training samples to learn a good classification model.

In a word, Semi-AdaBoost$^{IDT}$ and AdaBoost$^{IDT}$ algorithm in balanced data sets and imbalanced data sets have good classification effect on the binary classification, which indicates the proposed algorithm can improve the performance of the classification.
Table 2. Experimental results on imbalanced data set 1.

<table>
<thead>
<tr>
<th>percentage of labeled samples</th>
<th>Finance vs Health</th>
<th>Education vs Employment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AdaBoost</td>
<td>AdaBoost&lt;sup&gt;IDT&lt;/sup&gt;</td>
</tr>
<tr>
<td>0.1</td>
<td>0.86664</td>
<td>0.8805</td>
</tr>
<tr>
<td>0.2</td>
<td>0.86828</td>
<td>0.88651</td>
</tr>
<tr>
<td>0.3</td>
<td>0.87705</td>
<td>0.90283</td>
</tr>
<tr>
<td>0.4</td>
<td>0.85954</td>
<td>0.89431</td>
</tr>
<tr>
<td>0.5</td>
<td>0.85932</td>
<td>0.89764</td>
</tr>
<tr>
<td>0.6</td>
<td>0.81489</td>
<td>0.87768</td>
</tr>
</tbody>
</table>

Table 3. Experimental results on imbalanced data set 2.

<table>
<thead>
<tr>
<th>percentage of labeled samples</th>
<th>Finance vs Health</th>
<th>Education vs Employment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AdaBoost</td>
<td>AdaBoost&lt;sup&gt;IDT&lt;/sup&gt;</td>
</tr>
<tr>
<td>0.1</td>
<td>0.8551</td>
<td>0.86657</td>
</tr>
<tr>
<td>0.2</td>
<td>0.84115</td>
<td>0.84568</td>
</tr>
<tr>
<td>0.3</td>
<td>0.78937</td>
<td>0.81307</td>
</tr>
<tr>
<td>0.4</td>
<td>0.73315</td>
<td>0.74536</td>
</tr>
<tr>
<td>0.5</td>
<td>0.63519</td>
<td>0.63837</td>
</tr>
<tr>
<td>0.6</td>
<td>0.29533</td>
<td>0.34906</td>
</tr>
</tbody>
</table>

Multi-Classification Experiments with Insufficient Labeled Samples

In this paper, we carried out multiclass classification experiments under the condition of insufficient labeled samples to better evaluate the classification effect of the improved algorithm. Experiments on the algorithm of AdaBoost.MH, AdaBoost. MH<sup>IDT</sup> and Semi-AdaBoost. MH<sup>IDT</sup> are carried out respectively. In this experiment, the proportion of labeled samples in all data sets is 0.1, that is, the total number of training data set is 1791, and the total number of test data set is 16119. The result is shown in Table 4.

Table 4. Results of multi-classification experiments with insufficient labelled samples.

<table>
<thead>
<tr>
<th>Category</th>
<th>F1 Value</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AdaBoost</td>
<td>AdaBoost&lt;sup&gt;IDT&lt;/sup&gt;</td>
<td>Semi-AdaBoost&lt;sup&gt;IDT&lt;/sup&gt;</td>
</tr>
<tr>
<td>finance</td>
<td>0.72091</td>
<td>0.76462</td>
<td>0.8575</td>
</tr>
<tr>
<td>IT</td>
<td>0.63</td>
<td>0.68</td>
<td>0.7963</td>
</tr>
<tr>
<td>health</td>
<td>0.7213</td>
<td>0.7641</td>
<td>0.8649</td>
</tr>
<tr>
<td>sports</td>
<td>0.8798</td>
<td>0.90245</td>
<td>0.9047</td>
</tr>
<tr>
<td>tourism</td>
<td>0.678</td>
<td>0.731</td>
<td>0.8321</td>
</tr>
<tr>
<td>education</td>
<td>0.783</td>
<td>0.816</td>
<td>0.8675</td>
</tr>
<tr>
<td>employment</td>
<td>0.5234</td>
<td>0.6049</td>
<td>0.7124</td>
</tr>
<tr>
<td>culture</td>
<td>0.3142</td>
<td>0.395</td>
<td>0.5293</td>
</tr>
<tr>
<td>military</td>
<td>0.8483</td>
<td>0.872</td>
<td>0.9075</td>
</tr>
</tbody>
</table>

As can be seen from Table 4, the classification effects of AdaBoost.MH<sup>IDT</sup> and Semi-AdaBoost.MH<sup>IDT</sup> algorithm are better than the AdaBoost.MH algorithm in the case of multiclass classification. In the meantime, Semi-AdaBoost.MH<sup>IDT</sup> algorithm is superior to the other two algorithms because the Semi-AdaBoost.MH<sup>IDT</sup> algorithm uses SSL, which can add unlabeled samples meeting the given condition to the training set to train a better classification model than the other two algorithms. Overall, the Semi-AdaBoost.MH<sup>IDT</sup> algorithm performs well. However, as you can see from the Table 8, the classification effect of culture category of the three algorithms is somewhat deficient. The reason for this is that, except the factor of insufficient labeled samples, another factor that affects the classification results of the three algorithms is the features of the culture category. By looking through the features of culture category, we find that there are many features which consist of a single word in the high frequency features of the selected culture
category. On the contrary, there are fewer words. This has affected the classification effect to a certain extent. Overall, the classification results of Semi-AdaBoost.MH^{IDT} and AdaBoost.MH^{IDT} algorithms are better than AdaBoost.MH algorithm. Experiments show that the improved algorithm can enhance the classification result effectively. At the same time, the SSL can also enhance the classification performance effectively under the condition of insufficient labeled samples.

Conclusion and Future Work

This paper devises an algorithm which is based on SSL and improved decision tree called Semi-AdaBoost. MH^{IDT}. First, AdaBoost^{IDT} and Semi-AdaBoost^{IDT} are used to do binary classification. Experiments show that AdaBoost^{IDT} and Semi-AdaBoost^{IDT} can enhance the text classification performance effectively in the binary classification. Secondly, multiclass experiments with AdaBoost.MH, AdaBoost. MH^{IDT}, and Semi-AdaBoost. MH^{IDT} are carried out under the condition of insufficient labeled samples. Experiments indicate that SSL can enhance the text classification performance effectively under the condition of insufficient labeled samples. Meanwhile, the AdaBoost. MH^{IDT} can also enhance the classification performance effectively in the multiclass classification experiments.

In conclusion, Semi-AdaBoost.MH^{IDT} has a good performance on Chinese text categorization. In the future, we will try to reduce the time complexity of the algorithm first. Secondly, in the experiments, we find that the $Z_i$ of each feature tends to 1 when the iterations are over 100, so it is difficult to select the best weak hypothesis. Therefore, we will devote to solving the problem.

References

[22] Li, W., Duan, L., Xu, D., & Tsang, I. W., IEEE Pattern Anal, 36, 6, 1134-1148, 2014.