A Study of Dependency Features for Chinese Sentiment Classification

Pu Zhang1,2,* Zhongshi He1, Lina Tao3
1 College of Computer Science, Chongqing University, Chongqing, China
2 College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China
3 Chongqing Communications Research and Design Institute Co., Ltd., China Merchants, Chongqing, China
*Corresponding author Email: zhangpu1977@yahoo.com

Abstract—Syntactic dependency features, which encode long-range dependency relations and word order information, have been employed in sentiment classification. However, much of the research has been done in English, and researches conducted on exploring how features based on syntactic dependency relations can be utilized in Chinese sentiment classification are very rare. In this study, we present an empirical study of syntactic dependency features for Chinese sentiment classification. First, we consider two types of feature sets (word unigrams and word-dependency relations), three commonly-used feature weighting schemes (term presence, term frequency, and TF-IDF), and two well-known learning methods (Naive Bayes and SVM) to evaluate the performance of different classifiers. Then, we use ensemble technique to combine different types of features and classification algorithms. Specifically, two types of ensemble methods, namely average combination method and meta-learning combination method, are evaluated for two ensemble strategies. Through a wide range of comparative experiments conducted on two widely-used datasets in Chinese sentiment classification, finally, some in-depth discussion is presented and conclusions are drawn about the effectiveness of dependency features for Chinese sentiment classification.

Index Terms—sentiment analysis, sentiment classification, dependency features, ensemble learning

I. INTRODUCTION

Sentiment analysis, also called opinion mining, is the field of study that analyzes people’s opinions, sentiments, evaluations, attitudes, and emotions from written language [1], and it has received increasing attention from academics and practitioners in recent years. Among various sentiment analysis tasks, an important sub-task is sentiment classification, which aims to classify an opinionated piece of text as expressing an overall positive or negative polarity.

A very popular technique for sentiment classification is supervised machine learning approach. The performance of the approach is heavily dependent on the choice of algorithms and the representation of documents. Since the pioneering work of Pang et al. [2], various supervised classification algorithms including Naive Bayes (NB), Maximum Entropy (ME), Winnow, KNN, ANN and support vector machines (SVM) have been employed as machine learning methods [3-6]. Apart from the choice of algorithms, the representation of documents also plays a critical role in supervised machine learning approaches for sentiment classification. Early work in [2] showed that using word unigrams as features and encoding each word feature in the set by its presence or absence in the document performs quite well in classification. In this case, a document is represented as a binary vector.

In subsequent research works, various kinds of features such as word Ngrams [4,8], character Ngrams [4,7,9-10], POS based features [11-12], substring-group features and sentiment word features [10] have been exploited.

With the attempt to capture word relation information behind the text, dependency-based features were also utilized in sentiment classification [13-16]. However, these works only focused on English documents, and the research results of sentiment classification of English are often unable to be directly applied to Chinese ones owing to unique way of emotional expression in Chinese and a larger variety of syntactic dependency and a higher degree of ambiguity in sentences than English [17]. To our best knowledge, the studies on using dependency-based features for Chinese sentiment classification are very rare and no extensive evaluation has been carried out to systematically analyze the impact of syntactic dependency features in Chinese sentiment classification. Furthermore, while the above-mentioned works focused on extending the document representation with dependency-based features, no systematic study has been carried out on feature weighting to explore whether different feature weighting schemes can result in different classification accuracy.

Therefore, the primary goal of this study is making an intensive study of the use of syntactic dependency features for Chinese sentiment classification, and our attempt is to seek answers based on empirical evidence to the following questions:

1) Are dependency features suitable for Chinese sentiment classification?

2) By jointly using word unigrams features and dependency-based features, can the performance of a sentiment classification model benefit from the addition...
of dependency-based features over a feature space that only includes traditional word unigrams?

3) By combining different types of features (word unigrams and dependence-based features) with classification algorithms, can the performance of a sentiment classification model benefit from the ensemble technique?

4) In topical text classification, feature weighting scheme plays an important role, does this also hold for Chinese sentiment classification?

In this study, we first investigate two machine learning methods (NB and SVM) on datasets using word unigrams and dependency relations features with three feature weighting schemes. Then, we apply two types of ensemble methods (average combination method and meta learning combination method) with two ensemble strategies (ensemble of feature sets and ensemble of both feature sets and classification algorithms) to integrate feature sets with three different feature weighting schemes including term presence (TP), term frequency (TF), and term frequency – inverse document frequency (TF-IDF) respectively. A number of extensive experiments are conducted on two widely-used datasets in Chinese sentiment classification, and we make in-depth discussion and answer the above four questions.

The rest of this paper is organized as follows. Related work and the machine learning methods are described in Section 2 and 3 respectively. In Section 4, we present the ensemble framework for sentiment classification. Sections 5 and 6 describe the experimental setup and results respectively. Finally, we draw conclusions and outline directions for future work in Section 7.

II. RELATED WORK

As a special case of text classification for opinionated texts, in recent years, sentiment classification has become increasingly important due to more and more opinionated information appearing on the Internet. Pang et al. [2] firstly applied NB, ME, and SVM to classify movie reviews into positive and negative classes. The experimental results demonstrated that using word unigrams (a bag of words) as features in classification performed well. Cui et al. [3] studied multiple classification algorithms for sentiment classification on large-scale data set. Liu et al. [18] explored various lexical features and different classification strategies for opinion analysis on blog data. In subsequent work [19], they compared different linguistic features for both blog and review sentiment classification. Arora et al. [20] used an efficient frequent subgraph mining algorithm to extract subgraph features for sentiment classification.

For Chinese sentiment classification, various methods such as lexicon based method [21] and ontology based method [22] have been explored in recent years. Besides that, there have been many works based on the supervised machine learning techniques. Li et al.[4] compared four machine learning methods (NB, SVM, MaxEnt, and ANN) using different feature representations including Word-Based Unigram (WBU), Bigram (WBB), Chinese Character-Based Bigram (CBB), and Trigram (CBT) with different feature weighting schemes on a review corpus which is made up of 16,000 reviews. Tan et al. [5] used word unigram as feature with TF-IDF weighting scheme and investigated five supervised machine learning methods (NB, SVM, KNN, centroid classifier, and winnow classifier) and four feature selection methods (MLIG, CHI and DF) on a Chinese sentiment corpus. Zhai et al. [10] used the SVM method and exploited more complex features including substrings, substring-groups, and key-substring-groups on two Chinese review datasets in different domains.

As a traditional text representation method in sentiment classification, bag-of-words (BOW) model is quite efficient and simple. However, word order is disrupted and syntactic structures are broken, and a great deal of information from original text is discarded [15]. Therefore, some works have tried more sophisticated syntactic features such as dependency relations for the task of sentiment classification. Joshi et al. [13] used a transformation of dependency relation triples as additional features to unigrams and yielded a better performance than using purely lexicalized dependency relations. Xia et al. [15] conducted experiments on both the movie reviews dataset used in [2] and the E-product dataset used in [13] and found that individual word dependency relations features (WR-DP) are inferior to unigrams. Furthermore, they noted that the performance of ensemble model integrating different types of features is significantly better than joint features. In subsequent work, they also took advantage of ensemble frameworks for integrating different feature sets and classification algorithms to boost the overall performance of classification model on the five datasets [16]. By using mined frequent dependency subtree patterns as features for SVM, Matsumoto et al. [23] attained significant improvement in the performance of sentiment classification on movie reviews dataset. Dave et al. [24] used adjective-noun dependency relationships as additional features to word unigrams and found that it is ineffective to improve the performance. Ng et al. [14] observed that the addition of dependency relationships does not improve performance over a feature space that includes unigrams, bigrams and trigrams.

However, all the above-mentioned works which used dependency relationships as features only focused on English, and the studies on using dependency-based features for Chinese sentiment classifications are very rare.

Besides features, the weight of each feature in feature vector is also the key component of the representation of a document. In sentiment classification, term presence has been widely used as feature weighting method [2,15-16,25] and has become the most frequently used feature weighting scheme. Other feature weighting schemes were also used to calculate feature weights and achieved good performance in sentiment classification. For example, term frequency was used as feature weighting scheme in the works of [6, 12]. Standard TF-IDF and variants of it were used in the works of [4-5,10,26-27]. However, despite the fact that the use of these feature weighting
schemes is commonplace, there has been little research into the effects of different feature weighting schemes in sentiment classification.

III. MACHINE LEARNING METHODS

A. Naive Bayes

As a probabilistic generative model, NB treats each document as a bag of words and assumes the words are mutually independent. It classifies each test document using Bayes rule by calculating the posterior probability that the document belongs to different classes and assigns the document to the class with the highest posterior probability. Assume a test document \( d \) is represented by \([w_1, \ldots, w_m]\), where \( w_k \) is the kth word appearing in the document, and \( C \) denotes the class label set of documents, which is represented by \([c_1, \ldots, c_l]\), where \( c_k \) denotes the kth class label appearing in the documents. By using Bayes rule and conditional independent assumption, Naive Bayes decision can be described as the following:

\[
\text{argmax}_{j=1,\ldots,c} \prod_{l=1}^{m} P(c_j|w_l) \]

There are two commonly used models for Naive Bayes, namely multinomial model and multi-variate Bernoulli model [28]. In the multinomial model, a document is represented by the set of word occurrences. And in the multi-variate Bernoulli model, a document is represented as a vector of binary attributes indicating the presence or absence of the word. Based on the multinomial model, Rennie et al. [29] proposed transformed weight-normalized complement Naive Bayes (TWCNB) model with some of the modifications including TF-IDF conversion and document length normalization. Specifically, the multi-variate Bernoulli model, the multinomial model, and the TWCNB model use TP,TF and TF-IDF feature weighting schemes respectively.

B. Support Vector Machines

As a discriminative model, SVM is based on the structural risk minimization principle from the computational learning theory. It seeks the maximal margin decision boundary to separate the data points into positive and negative examples. As stated in [30], SVM can be categorized into linear SVM and nonlinear SVM.

Given the training data set as \{\( (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \)\}, where \( x_i = (x_{i1}, x_{i2}, \ldots, x_{ir}) \) is a r-dimensional input vector in a real-valued space, \( y_i = \{1,-1\}\), the optimization problem of finding maximized margin is as following:

\[
\begin{align*}
\text{Minimize:} & \quad \frac{1}{2} \|W\|^2 + C \sum_{i=1}^{N} \xi_i \\
\text{Subject to:} & \quad y_i g(X_i) \geq 1 - \xi_i, \ i = 1,2,\ldots,N \\
& \quad \xi_i \geq 0, \ i = 1,2,\ldots,N 
\end{align*}
\]

Where \( W = (w_1, w_2, \ldots, w_d) \) is called as the weight vector, \( C \) is the penalty coefficient, and \( \xi_i \) denotes the slack variable. The linear SVM uses \( g(X_i) = W^T X_i + b \) as the discriminant function. To deal with nonlinearly separable data, the same formulation and solution techniques as for the linear SVM are still used for the nonlinear SVM. It uses \( g(X_i) = W^T \Phi(X_i) + b \) as the discriminant function, where \( \Phi(\cdot) \) is a nonlinear mapping which maps the data in the input space to a feature space.

IV. THE ENSEMBLE MODEL

In recent years, there has been a growing interest in using ensemble learning techniques in sentiment classification. Xu et al. [7] proposed an ensemble learning algorithm based on random feature space division method for sentiment recognition of Chinese movie reviews. Abbasi et al. [9] proposed a correlation ensemble method for affect analysis. Whitehead et al. [31] used ensemble methods including bagging, boosting, and random subspace for sentiment classification. Wang et.al [32] also conducted a comparative assessment of the performance of these three popular ensemble methods on ten public sentiment analysis datasets to verify the effectiveness of ensemble learning for sentiment analysis. Rather than an ensemble of different data re-sampling methods such as bagging and boosting, recently, the ensemble method is further enriched by considering various feature sets and learning models, and it has been applied in sentiment classification [15-16,33-34].

In this study, we also adopt ensemble of feature sets and classification algorithms. In the ensemble framework, different participants can be generated by different contributing classifiers on component feature sets. And two types of feature sets are employed by us for sentiment classification, namely word unigrams and word-dependency parsing pairs. The process of feature extraction is as follows. Each review document is split by punctuation mark into sentences, then, the features can be obtained from sentences. Since Chinese does not segment words by spaces in sentence, each sentence is needed to be segmented into words and word unigram features can be obtained. Taking the sentence ‘外形确实不错 (Appearance is really good)’ as an example, ‘外形’ (‘appearance’), ‘确实’ (‘really’) and ‘不错’ (‘good’) are considered as word unigram features. Similarly, dependency features can be obtained from a given sentence by using parser. As a structured representation, the word dependency parsing pairs for a given sentence are essentially a set of triples, each of which expresses the dependency relation between words, and the triple is composed of a grammatical relation and the pair of words from the sentence. For example, the dependency triples of the sentence ‘外形确实不错 (Appearance is really good)’ are demonstrated in Fig.1.

Figure 1. A demonstration of dependency parsing tree
In the Fig.1, each dependency parsing pair has the form of \(<w_i, w_k, rel>\), where rel, is the dependency relation between words \(w_i\) and \(w_k\). For instance, dependency parsing pairs for the example sentence include \(<\text{外形}, 不错, SBV>\), \(<\text{确实}, 不错, ADV>\) and \(<\text{1, 不错}, HED>\), where ‘SBV’ indicates that ‘外形’ (‘appearance’) is a subject modifier of the target ‘不错’ (‘good’), ‘ADV’ indicates that ‘确实’ (‘really’) is an adverbial modifier of the target ‘不错’ (‘good’), and ‘HED’ indicates that ‘不错’ (‘good’) is the head word in the sentence. In this study, we straightforwardly use dependency parsing pairs as dependency features.

For our ensemble task, the output values generated by base classifiers are taken as inputs of the ensemble methods to form an integrated output. And we apply the following ensemble methods.

1) Average
It is the most intuitive combination method and the new semantic orientation value of a document is the average of the output values from constituent classifiers.

2) Meta-Learning
The meta-learning method [35] has been used in sentiment classification [15-16]. The key idea behind it is to train a meta-classifier with the outputs of the base classifiers as input attributes. Development data is usually needed for meta-learning to generate the meta-training data. In this study, instead of using extra development data, we perform stacking [36] with 5-fold cross-validation to generate the meta-training data. The stacking is a two-phase framework that is concerned with combining multiple classifiers. In the first phase, a set of base-level classifiers are generated. And in the second phase, a meta-level classifier that combines the outputs of the base-level classifiers is learned.

Taking a dataset as an example, in each loop of the 5-fold cross validation, we consider the probabilistic outputs of the test fold as test samples for meta-learning. To generate a training set for meta-learning classifier, we apply an inner 4-fold leave-one-out procedure to the training data. In each of the four fold, samples are trained on the remaining three folds to obtain the probabilistic outputs which are treated as training samples for meta-learning classifier.

It should be noted that the outputs of different classifiers should be transformed to a uniform measure when evaluating the degree of decision confidence [37]. Therefore, for NB model, we use the posterior probability obtained from the classifier as the output. And the posterior probability can be obtained by the following formula:

\[
p(y|d) = \frac{\exp(o_j)}{\sum_{k=1}^{C} \exp(o_k)}
\]  

(3)

Where \(o_j = \log(p(y=j)p(d|y=j))\), it denotes the output belonging to class \(j\). \(C\) is the number of class label set, \(d\) denotes a document and \(y\) is the class label of \(d\).

As the output of the SVM classifier for a document is a real number score, to convert the score into the polarity probability, we use the following formula which is presented in [38].

\[
\begin{align*}
Pr_{pos}(d) &= \begin{cases}
1 & s \geq \xi \\
0.5 + \frac{s - \xi}{2\xi} & \xi < s \leq -\xi \\
0 & s \leq -\xi
\end{cases} \\
Pr_{neg}(d) &= 1 - Pr_{pos}(d)
\end{align*}
\]

(4)

In the formula, \(s\) denotes the output of the polarity classifier for a document \(d\), it is a real-number score. And \(\xi\) is a threshold value, we use an empirical threshold \(\xi = 2\).

V. EXPERIMENTS SETUP

A. Datasets and Evaluation Metrics
We conduct experiments on the two Chinese datasets. The first is ChnSentiCorp-2000 dataset [39] from hotel domain. The other dataset is from IT product domain and it is introduced in [40], for convenience, we name it as “Mobile” dataset. A brief summary of the two datasets is shown in Table 1.

| TABLE I | THE SUMMARY OF THE DATASETS |
|---|---|---|
| Mobile | Labeled positive reviews | 1159 |
| | Labeled negative reviews | 1158 |
| ChnSentiCorp-2000 | Labeled positive reviews | 1000 |
| | Labeled negative reviews | 1000 |

We use accuracy metric to measure the overall performance. And the metric is calculated by using the following formula:

\[
\text{accuracy} = \frac{\text{number of correctly identified reviews}}{\text{number of labelled reviews}}
\]

(5)

B. Pre-processing
Unlike English, Chinese does not segment words by spaces in sentence. Therefore, to get the word unigram features and dependency relations, pre-processing steps such as word tokenization and dependency parsing should be taken. We use the ICTCLAS toolkit [41] for word tokenization. And LTP- an integrated Chinese processing platform described in [42] is chosen as the tool to extract dependency parsing features.

C. Implementation
For the two datasets, each dataset is evenly divided into 5 folds and all the following experimental results are obtained with a 5-fold cross validation, where each test fold contains all the reviews of one of the fold, and the reviews of the remaining 4 folds are used for training. The performance results reported in all of the following tables is in terms of the average classification accuracy.

We experiment with two standard classification algorithms: NB and SVM. For NB classification model, three types of naive bayes classifiers provided in WEKA [43], namely NaiveBayes, NaiveBayesMultinomial and ComplementNaiveBayes are employed. Specifically, these three classifiers are implementations of the multivariate Bernoulli model, the multinomial model, and the TWCNB model respectively. The SVMLight toolkit [44] is chosen as the SVM classifier. The tool of SVMLight is
chosen with the linear kernel and default parameter values. NB and SVM are also used as base level classification models of the ensemble model.

VI. EXPERIMENT RESULTS AND ANALYSIS

In this section, we first show the results of individual classifiers, and then report the results of two ensemble models which adopt average ensemble method and meta-learning ensemble method respectively with two different ensemble strategies, namely ensemble of feature sets and ensemble of both feature sets and classification algorithms.

A. Results of Individual Classifiers

We evaluate performances of two learning methods on the two datasets using different feature representations combined with three different feature weighting schemes. The feature representation schemes include Word-Based Unigram (WU), dependency parsing pair (DEP), and joint feature (WU+DEP). The feature weighting schemes include TP, TF, and TF-IDF. The results are presented in Table 2-3 respectively. In order to show the comparative performances respect to TP and TF, we also give the average accuracies of the two datasets in Table 4.

(1) Comparison of different features

On the ChnSentiCorp-2000 dataset, as shown in Table 4, we can see that the average accuracies on the two datasets in Table 4.

Table II.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Classifier</th>
<th>Feature weighting</th>
<th>TP</th>
<th>TF</th>
<th>TF-IDF</th>
</tr>
</thead>
<tbody>
<tr>
<td>WU</td>
<td>NB</td>
<td>83.25</td>
<td>91.66</td>
<td>91.79</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>86.66</td>
<td>86.18</td>
<td>93.98</td>
<td></td>
</tr>
<tr>
<td>WU+Dep</td>
<td>NB</td>
<td>87.35</td>
<td>90.37</td>
<td>90.28</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>91.70</td>
<td>91.49</td>
<td>94.43</td>
<td></td>
</tr>
</tbody>
</table>

Table III.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Classifier</th>
<th>Feature weighting</th>
<th>TP</th>
<th>TF</th>
<th>TF-IDF</th>
</tr>
</thead>
<tbody>
<tr>
<td>WU</td>
<td>NB</td>
<td>76.20</td>
<td>88.15</td>
<td>88.60</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>88.45</td>
<td>86.85</td>
<td>90.35</td>
<td></td>
</tr>
<tr>
<td>WU+Dep</td>
<td>NB</td>
<td>87.85</td>
<td>86.65</td>
<td>92.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>86.95</td>
<td>85.39</td>
<td>92.20</td>
<td></td>
</tr>
</tbody>
</table>

Table IV.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Classifier</th>
<th>Feature weighting</th>
<th>TP</th>
<th>TF</th>
<th>TF-IDF</th>
</tr>
</thead>
<tbody>
<tr>
<td>WU</td>
<td>NB</td>
<td>82.38</td>
<td>91.21</td>
<td>91.67</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>87.78</td>
<td>87.31</td>
<td>93.02</td>
<td></td>
</tr>
<tr>
<td>WU+Dep</td>
<td>NB</td>
<td>90.68</td>
<td>89.17</td>
<td>92.39</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SVM</td>
<td>88.62</td>
<td>88.08</td>
<td>92.98</td>
<td></td>
</tr>
</tbody>
</table>

3, under NB or SVM model with three different feature weighting methods, DEP is always superior to WU. For example, the accuracy of NB using DEP feature with TF weights is 90.75%, while the accuracy of NB using WU with TF is 88.15%.

On the Mobile dataset, as we can see from Table 2, for NB and SVM, when using TP as feature weighting, WU consistently outperforms DEP, while using the other two feature weighting schemes, this is not always the case.

Another observation from Table 2 and 3 is that when using TP as feature weighting scheme, for NB classification model, joint features (WU+DEP) provide a significant improvement over any individual feature set. For example, there is 11.65% absolute improvement in accuracy over the case using WU features on the ChnSentiCorp-2000 dataset (87.85% vs. 76.20%) in Table 3. In most other cases, compared with the best results achieved for individual feature sets, however, WU+DEP leads to no significant improvements, sometimes it even yields worse results. For example, on the ChnSentiCorp-2000 dataset, we can see from Table 3 that with TF feature weighting method, the performance of NB using DEP is 90.75%, while the performance of NB using WU+DEP is 86.65%, with a decrease of 4.1% in accuracy.

So we can see that adding DEP as extra feature does not always provide benefit over a simple bag-of-words based feature space when using TP or TF feature weighting. On the other side, when using TF-IDF feature weighting scheme, as indicated by the last column of Table 4, improvement in performance can be obtained for NB and SVM by adding DEP as extra feature in addition to WU (92.98% vs. 91.67% & 89.44% and 93.90% vs. 93.02% & 92.39%).

(2) Comparison of different feature weighting schemes

In each row of the Table 2-4, we can see that among the three feature weighting schemes, TF-IDF consistently outperforms the other two schemes once the classifier and the feature sets are selected. So, the results demonstrate that TF-IDF turns out to be the most effective among all the three feature weighting schemes. Meanwhile, with performances respect to TP and TF feature weights, we can see that in Table 2-4, for the SVM model, no matter what form of feature sets are used, TP is always better than TF. While for the NB model, when using DEP or WU feature sets, TF is always better than TP.

(3) Comparison of different classifiers

With TP feature weighting scheme, on individual feature sets (WU or DEP), we can see from Table 2-3 that SVM always exceeds better performance than NB. This is in accordance with the results reported by Pang et al. [2]. But it turns out this is not always the case when using TF or TF-IDF feature weighting schemes. For example, when using DEP feature sets with TF weights, from Table 4, we can see that the average accuracies on the two datasets of NB always outperform SVM. However, in most other cases, the performance of SVM is superior to NB.
B. Results of Ensemble Models

The results of ensemble models on the two datasets are reported in Table 5-6. And there are two ensemble methods, namely average combination method and meta-learning combination method, both of which use two ensemble strategies (ensemble of different feature sets and ensemble of both feature sets and classifiers). For convenience, we name the former strategy as Strategy-1, and the latter strategy as Strategy-2. In Table 7, we also give out the average results of the ensemble models on two datasets.

In Table 5-7, we use denotations for convenience. For example, ‘Ave-(SVM@WU)&(SVM@DEP)’ denotes an ensemble model which uses the average combination method with Strategy-1 as the ensemble strategy, and the strategy integrates two kinds of feature sets(WU and DEP) with the individual SVM classifier. While ‘Meta-(SVM@WU)&(SVM@DEP)’ denotes an ensemble model which uses the meta-learning combination method with Strategy-1. Similarly, ‘Ave-(SVM@WU)&(NB@DEP)’ denotes an ensemble model which uses the average combination method with Strategy-2 as the ensemble strategy, and the strategy integrates two kinds of feature sets(WU and DEP) with different classifiers (SVM and NB), where SVM uses WU as feature sets and NB uses DEP as feature sets. ‘Meta-(NB@DEP)&(SVM@WU)’ denotes an ensemble model which uses the meta-learning combination method with Strategy-2. The other denotations are similar, and so on.

For comparison purposes, we also report the results of the classifiers which use joint feature sets (WU+DEP) in Table 5-7, where ‘NB@(WU+DEP)’ denotes the NB model which uses WU+DEP as feature representation. Similarly, ‘SVM@(WU+DEP)’ denotes the SVM classification model on WU+DEP feature sets.

(1) Comparison with joint features

From Table 7, we can see that when using TF feature weighting scheme, all of the ensemble models can consistently outperform the individual classification model which uses joint features, and the performance improvements range from 2.07% (90.15% vs. 88.08%) to 5.08% (92.58% vs. 87.50%). For the other two feature weighting schemes, when the highest performance of the individual classification model and that of the ensemble models are selected for comparison, we can see the performance of the ensemble model gains improvement by 4.36%(93.00% vs. 88.64%) for TP weighting scheme.
while with little drop (93.77% vs. 93.90%) for TF-IDF weighting scheme. So we can conclude that when using TF-IDF weights, in most cases, the performance do not benefit from using multiple classifiers combination.

(2) Comparison with ensemble models
Among the different ensemble models, the model which uses average combination method is the most attractive. As shown in Table 7, on the average, regarding to the three feature weighting methods: TP, TF, and TF-IDF, the highest performance is achieved by ‘Ave-(SVM@WU)&(SVM@DEP)’, ‘Ave-(NB@WU)&(NB@DEP)’, and ‘Ave-(SVM@WU)&(SVM@DEP)’ respectively. Furthermore, among all the ensemble models, in general, ‘Ave-(SVM@WU)&(SVM@DEP)’ yields the best results. From the above results, we can conclude that the average combination method is preferred since its overall performance is better than the meta-learning combination method, furthermore, the computational cost of the average combination method is less than the meta-learning combination method.

When considering the ensemble strategies, it can be seen from Table 5-7, the best performances are always achieved by the Strategy-1 across both ensemble methods and datasets. It is thus concluded that the Strategy-1 is more effective than the Strategy-2.

(3) Comparison with feature weighting schemes
As we can see from each row of Table 7, the results of TF-IDF consistently exceed the other results. In more detail, the same conclusion holds for each row of Table 5-6 with only one exception in Table 5, that is ‘Ave-(NB@WU)&(NB@DEP)’.

The reason for this phenomenon lies in the fact that in Table 2-4, we can see that TF-IDF always achieves the best performance once the individual classifier and the feature sets are selected. Naturally, when the multiple classifiers using TF-IDF weights are combined into an ensemble model, the results also attain the highest performance. This is in accordance with the conclusions drawn from Table 2-4.

VII. CONCLUSION AND FUTURE WORK
In this study, we aim to explore the use of dependency features for Chinese sentiment classification. We conduct a range of comparative experiments on two widely-used Chinese datasets by considering two types of feature sets, two schemes of ensemble methods, and two ensemble strategies. Based on the experimental results, questions in the Section 1 are answered respectively as following:

1). As we can see from Table 3, on the CnsentiCorp-2000 dataset from hotel domain, DEP feature is more effective than traditional WU feature when the classifier and feature weighting scheme are selected. On the Mobile dataset from digital products domain, from Table 2, we can see that when using TF and TF-IDF schemes, WU is superior to DEP. Therefore, in general, we can see that DEP feature can still be treated as a candidate effective feature depending on the domain and the feature weighting scheme. Furthermore, when using dependency features as feature representations, to achieve better performance, if TP or TF-IDF are selected as feature weight schemes, we prefer using SVM as supervised learning method. But when using TF feature weighting scheme, we prefer using NB as learning method.

2). When using dependency features as additional supplement features to word unigrams, with TF-IDF weighting scheme, it is effective for Chinese sentiment classification. While with TF feature weighting scheme, the effects are not so obvious. Furthermore, when using TF as feature weighting scheme, the performances of the classification models on the two datasets even degrade. So, we can conclude that the effectiveness of using joint feature sets (WU+DEP) should be related to feature weighting scheme.

3). As described in the Section 6, from the last column of Table 7, we can see that when using TF-IDF feature weighting scheme, compared with the results of individual classifiers on joint feature sets, in most cases, the performances do not benefit much from using multiple classifiers combination. Whereas using TF weighting scheme, all of the ensemble models which use different ensemble methods with different ensemble strategies consistently outperform the performances of individual classifier models which use joint feature sets, as is shown in Table 7. When using TF feature weighting scheme, we can see that ‘Ave-(SVM@WU)&(SVM@DEP)’ performs well compared with the results of individual classification models which use joint features, whereas for the other ensemble models, this is not always the case. Generally, when using ensemble technique for Chinese sentiment classification, we prefer average combination method with the strategy that is an ensemble of feature sets using a single classification model.

4). From the above points, we can conclude that feature weighting scheme does indeed impact on the performance of classification model and choosing the proper feature weighting scheme can be crucial to the performance of Chinese sentiment classification. Overall, TF-IDF is the most successful feature weighting scheme for both SVM and NB.

Future work will include evaluating feature selection methods and finding out the reason of why the effectiveness of dependency features depends on domain and feature weighting scheme, it is also worth utilizing corpus in other domains (e.g. restaurant reviews) for this work.

ACKNOWLEDGMENT
This work was supported in part by the National Nature Science Foundation of China (No. 61075019) and the Fund of Nature Science of Chongqing University of Posts and Telecommunications (No.A2011-31).

REFERENCES

Cui, H., Mittal, V., and Datar, M., “Comparative Experiments on Sentiment Classification for Online Product Reviews”, In Procedings of the Twenty-First National Conference on Artificial Intelligence (AAAI-06), July 16 - 20, 2006, Boston, Massachusetts.


[41] ICTCLAS: [http://www.ictclas.org](http://www.ictclas.org)


Pu Zhang is a PhD candidate in College of Computer Science, ChongQing University, China. He received M.S. degree in Computer Science Department from Sichuan University, China. He is an Associate Professor in College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing, China. His current research focuses on nature language processing, sentiment analysis and text mining.

Zhongshi He received Ph.D. degree in Computer Science from Chongqing University in 1996. He is now a Professor and PhD supervisor of Chongqing University, Chongqing, China. His research interests include nature language processing, machine learning and pattern recognition.

Lina Tao is now a Research Associate of Chongqing Communications Research and Design Institute Co.,Ltd., Chongqing, China. Her current research interests include data mining, and scientific computing.