Aspect-Based Sentiment Analysis Using Tree Kernel Based Relation Extraction

Thien Hai Nguyen and Kiyoaki Shirai

School of Information Science,
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan
{nhthien,kshirai}@jaist.ac.jp

Abstract. We present an application of kernel methods for extracting relation between an aspect of an entity and an opinion word from text. Two tree kernels based on the constituent tree and dependency tree were applied for aspect-opinion relation extraction. In addition, we developed a new kernel by combining these two tree kernels. We also proposed a new model for sentiment analysis on aspects. Our model can identify polarity of a given aspect based on the aspect-opinion relation extraction. It outperformed the model without relation extraction by 5.8% on accuracy and 4.6% on F-measure.

Keywords: Sentiment Analysis, Relation Extraction, Tree Kernel, Support Vector Machine.

1 Introduction

Relation extraction is a task of finding relations between pairs of entities in texts. Many approaches have been proposed to learn the relations from texts. Among these approaches, kernel methods have been used increasingly for the relation extraction [4,15,17,18]. The main benefit of kernel methods is that they can exploit a huge amount of features without an explicit feature representation. In the relation extraction task, many kinds of relations, from general to specific ones, are considered. This paper focuses on aspect-opinion relation, which is a relation between an aspect of an entity (eg. a price of a PC) and an opinion word or phrase that expresses evaluation on that aspect. It is still an open question if the kernel methods also work well for aspect-opinion relation extraction.

On the other hand, sentiment analysis is considered as an important task in an academic as well as commercial point of view. Many researches attempted to identify polarity of a sentence or paragraph regardless of the entities such as restaurants and their aspects such as food or service. While, this research focuses on aspect-based sentiment analysis, which is a task to identify the sentiment expressed towards aspects of entities.

The goal of our research is to develop a model to predict the sentiment categories (positive, neutral or negative) of the given aspect in the sentence. Intuitively, the opinion words related to the given aspect will have more influence

A. Gelbukh (Ed.): CICLing 2015, Part II, LNCS 9042, pp. 114–125, 2015.
DOI: 10.1007/978-3-319-18117-2_9

on the sentiment of that aspect. Our method firstly identifies the aspect-opinion relations in the sentence by tree kernel method. Then, it calculates the sentiment score for each aspect in the sentence by using these extracted relations.

Our contributions are summarized as follows:

1. We applied two existing tree kernels for aspect-opinion relation extraction.
2. We proposed a new tree kernel based on the combination of two tree kernels for aspect-opinion relation extraction.
3. We proposed a new method for aspect-based sentiment analysis enhanced by the automatically identified aspect-opinion relations.

The rest of this paper is organized as follows. Section 2 introduces some previous approaches on relation extraction and aspect-based sentiment analysis. Section 3 discusses methods for aspect-opinion relation extraction. Section 4 examines how to apply aspect-opinion relation extraction for aspect-based sentiment analysis. Finally, Section 5 concludes our research.

2 Previous Work

2.1 Relation Extraction

Some previous work used the dependency tree kernels for general relation extraction [4,15,18]. In these researches, they tried to extract all of the predefined relations in a given sentence. The predefined relations are *person-affiliation*, *organization-location* and so on. Nguyen et al. used tree kernel based on the constituent, dependency and sequential structures for relation extraction [13]. They focused on seven relation types such as *person-affiliation* in the ACE corpus, which was well-known as a dataset for general relation extraction. However, aspect-opinion relation was not considered in these researches. For the aspect-based sentiment analysis, it is very important to know whether there is a relation between an aspect and opinion word. To the best of our knowledge, there is a lack of researches trying to use tree kernel for aspect-opinion relation extraction.

Wu et al. proposed a phrase dependency parsing for extracting relations between product features and expression of opinions [17]. Their tree kernel is based on a phrase dependency tree converted from an ordinary dependency tree. However, they did not apply this model for calculating a sentiment score for a given aspect.

Bunescu and Mooney extracted the shortest path between two entities in a dependency tree to extract the relation between them [2]. The dependency kernel was calculated based on this shortest path. They suggested that the shortest path encodes sufficient information for relation extraction.

Kobayashi et al. combined contextual and statistical clues for extracting aspect-evaluation and aspect-of relations [7]. Since the contextual information is domain-specific, their model cannot be easily used in other domains.

2.2 Aspect-Based Sentiment Analysis

Aspect-based sentiment analysis has been found to play a significant role in many applications such as opinion mining on product reviews or restaurant reviews. The popular approach is to define a sentiment score of a given aspect by the weighted sum of opinion scores of all words in the sentence, where the weight is defined by the distance from the aspect [10,14]. Because this approach is simple and popular, it will be a baseline model in our experiment in Section 4. To the best of our knowledge, there is no research trying to apply aspect-opinion relation extraction for calculating the sentiment score of the given aspect in the sentence.

Other researches have attempted to use unsupervised topic modeling methods for aspect-based sentiment analysis. To identify the sentiment category of the aspect, topic models which can simultaneously exploit aspect and sentiment have been proposed, such as ASUM [5], JST [9] and FACTS model [8]. However, it is not obvious to map latent (inferred) aspects/sentiments to aspects/sentiments in the text.

3 Aspect-Opinion Relation Extraction

For a given sentence where an aspect phrase and opinion phrase have been already identified, we will determine whether there is a relationship between the aspect and opinion phrase. To achieve this goal, four supervised machine learning methods will be presented in the following subsections. One is Support Vector Machine (SVM) with a linear kernel and the others are SVM with tree kernels.

3.1 SVM-B: A Baseline Model

SVM has long been recognized as a method that can efficiently handle high dimensional data and has been shown to perform well on many applications such as text classification [6,12]. A set of features used for training SVM is shown in Table 1. Because this model was also used in previous work [7,17] for relation extraction, we chose it as a baseline model to compare with other methods.

Table 1. Features used in SVM-B

Feature	Values
Position of opinion word in sentence	{start, end, other}
Position of aspect word in sentence	{start, end, other}
The distance between opinion and aspect	{1, 2, 3, 4, other}
Whether opinion and aspect have direct dependency relation	{True, False}
Whether opinion precedes aspect	{True, False}
Part of Speech (POS) of opinion	Penn Treebank Tagset
POS of aspect	Penn Treebank Tagset

3.2 CTK: Constituent Tree Based Tree Kernel

Tree kernel for the constituent tree has been used successfully in many applications. Various tree kernels have been proposed such as subtree kernel [16] and subset tree kernel [3]. We applied the subtree kernel for this research. Figure 1 shows an example of a constituent tree for the sentence "It has excellent picture quality and color."

Given a constituent tree of a sentence, we represented each $r(e_1, e_2)$, aspect-opinion relation between the aspect entity e_1 and opinion entity e_2, as a subtree T rooted as the lowest common parent of e_1 and e_2. Notice that the aspect and opinion entity can be phrases in general. The subtree T must contain all of the words in these phrases. For example, the relation between the aspect "picture quality" and opinion "excellent" in Figure 1 is represented by the subtree rooted at "NP" node [1], which is the lowest common parent of "picture", "quality" and "excellent" node. The main idea of this tree kernel is to compute the number of the common substructures between two tree T_1 and T_2 which represent two relation instances. The kernel between two trees T_1 and T_2 is defined as in Equation (1).

$$K(T_1, T_2) = \sum_{n_1 \in N_1} \sum_{n_2 \in N_2} C(n_1, n_2) \tag{1}$$

N_1 and N_2 are the set of the nodes in T_1 and T_2. $C(n_1, n_2)$ is the number of common subtrees of two trees rooted at node n_1 and n_2. It is calculated as follows:

1. If n_1 and n_2 are pre-terminals with the same POS tag: $C(n_1, n_2) = \lambda$
2. If the production rules at n_1 and n_2 are different: $C(n_1, n_2) = 0$
3. If the production rules at n_1 and n_2 are the same:

$$C(n_1, n_2) = \lambda \prod_{j=1}^{nc(n_1)} (1 + C(ch(n_1, j), ch(n_2, j)))$$

where $nc(n_1)$ is the number of the children of n_1 in the tree. $ch(n_i, j)$ is the j^{th} child-node of n_i. Since the production rules at n_1 and n_2 are the same, $nc(n_1) = nc(n_2)$. We set $\lambda = 0.5$ in our experiment.

Finally, since the value of $K(T_1, T_2)$ will depend greatly on the size of the trees T_1 and T_2, we normalize the kernel as in Equation (2).

$$K'(T_1, T_2) = \frac{K(T_1, T_2)}{\sqrt{K(T_1, T_1) K(T_2, T_2)}} \tag{2}$$

3.3 DTK: Dependency Tree Based Tree Kernel

A dependency tree kernel has been proposed by Culotta and Sorensen for general relation extraction [4]. This paper applies it for aspect-opinion relation extraction. Given a dependency tree of a sentence, we represent each relation $r(e_1, e_2)$

[1] It is denoted by the circle in Figure 1.

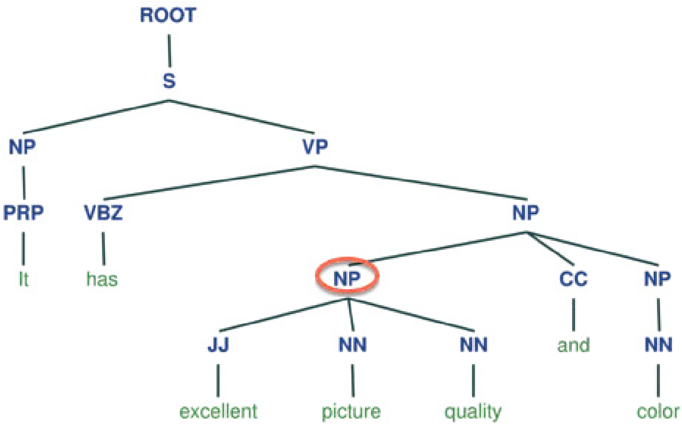

Fig. 1. An Example of Constituent Parsing Tree

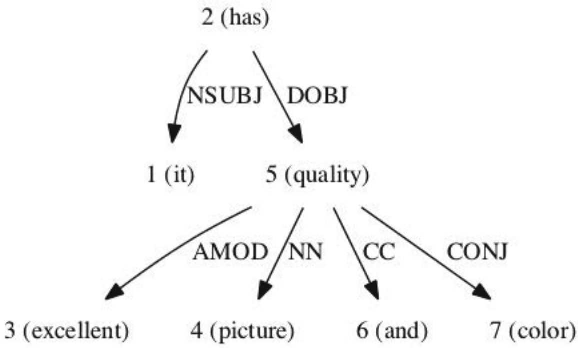

Fig. 2. An Example of Dependency Tree

as a subtree T rooted as the lowest common parent of the aspect e_1 and opinion e_2. For example, the relation between the aspect "picture quality" and opinion "excellent" in Figure 2 is the subtree rooted at "quality" node, which is the lowest common parent of "picture", "quality" and "excellent" node.

A subtree T of a relation instance can be represented as a set of nodes $\{n_0, \cdots, n_t\}$. Each node n_i is augmented with a set of features $f(n_i)$=$\{v_1, \cdots, v_d\}$. They are subdivided into two subsets $f_m(n_i)$ (features used for matching function) and $f_s(n_i)$ (for similarity function). A matching function $m(n_i, n_j) \in \{0, 1\}$ in Equation (3) checks if $f_m(n_i)$ and $f_m(n_j)$ are the same. A similarity function $s(n_i, n_j)$ in $(0, \infty]$ in Equation (4) evaluates the similarity between $f_s(n_i)$ and $f_s(n_j)$.

$$m(n_i, n_j) = \begin{cases} 1 \text{ if } f_m(n_i) = f_m(n_j) \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

$$s(n_i, n_j) = \sum_{v_q \in f_s(n_i)} \sum_{v_r \in f_s(n_j)} C(v_q, v_r) \tag{4}$$

In Equation (4), $C(v_q, v_r)$ is a compatibility function between two feature values as:

$$C(v_q, v_r) = \begin{cases} 1 \text{ if } v_q = v_r \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

For two given two subtrees T_1 and T_2 which represent for two relation instances with root nodes r_1 and r_2, the tree kernel $K(T_1, T_2)$ is defined as in Equation (6):

$$K(T_1, T_2) = \begin{cases} 0 & \text{if } m(r_1, r_2) = 0 \\ s(r_1, r_2) + K_c(r_1[c], r_2[c]) & \text{otherwise} \end{cases} \tag{6}$$

where K_c is a kernel function over children. Let $\mathbf{a}$ and $\mathbf{b}$ be sequences of children nodes' indices of node n_i and n_j, respectively. We denote the length of $\mathbf{a}$ by $l(\mathbf{a})$. K_c is defined as Equation (7). $n_i[\mathbf{a}]$ stands for the subtree consisting of children indicated by $\mathbf{a}$, while $n_i[a_h]$ is h^{th} child of n_i. In this equation, we consider the contiguous kernel enumerating children subsequences that are not interrupted by not matching nodes. In our experiment, λ is set to 0.5.

$$K_c(n_i[c], n_j[c]) = \sum_{\mathbf{a}, \mathbf{b}, l(\mathbf{a}) = l(\mathbf{b})} \lambda^{l(\mathbf{a})} K(n_i[\mathbf{a}], n_j[\mathbf{b}]) \prod_{h=1}^{l(\mathbf{a})} m(n_i[a_h], n_j[b_h]) \tag{7}$$

Finally, we also normalize the kernel as in Equation (2).

The augmented features are shown in Table 2. Note that POS, $isAspectNode$ and $isOpinionNode$ are used for matching between two nodes, while the rest is used for measuring the similarity of them.

Table 2. Features for Each Node in the Dependency Tree

	Feature	**Values**
f_m	POS	Penn Treebank POS Tagset
	$isAspectNode$	{0,1}
	$isOpinionNode$	{0,1}
f_s	NER	StanfordCoreNLP Name Entity Tagset
	$relationToParentNode$	StanfordCoreNLP Dependency Relation
	$POSofParentNode$	Penn Treebank POS Tagset
	$NERofParentNode$	StanfordCoreNLP Name Entity Tagset

3.4 CTK + DTK: Combination of Two Kernels

We proposed a new tree kernel based on the combination of two kernels CTK and DTK for aspect-opinion relation extraction. That is, we try to utilize the information from both the constituent and dependency tree. Equation (8) defines the combined kernel function.

$$K_{CTK+DTK}(T_1, T_2) = K_{CTK}(T_1, T_2) + K_{DTK}(T_1, T_2) \tag{8}$$

$K_{CTK}(T_1, T_2)$ and $K_{DTK}(T_1, T_2)$ are the CTK and DTK tree kernels described in Subsection 3.2 and 3.3, respectively. Since the summation of two kernels is valid, $K_{CTK+DTK}$ is obviously a valid kernel.

3.5 Evaluation of Tree Kernel Based Relation Extraction

Dataset: We conducted experiments with labeled dataset developed by Wu et al. [17]. We also corrected some errors such as typing errors, aspect and opinion marking errors, and removed redundant relations. There are 5 domains (DVD Player, Cell Phone, Digital Camera, Diaper, MP3 Player) in this dataset. Stanford CoreNLP [11] was used to parse constituent and dependency tree for each sentence.

Two experiments were designed. The first one is in-domain evaluation. This experiment tries to answer the question how well the models classify the data in the test set which is the same domain of the training data. We divided each domain into 80% for training and 20% for testing. The second experiment is cross-domain evaluation. This evaluates the models on the test set which is different domain to the training data. We used the sentences in "Digital Camera" and "Cell Phone" domains for training, and evaluated the models on "DVD Player", "Diaper" and "MP3 Player" domains. Accuracy, Precision, Recall and F-measure are used as the evaluation metrics. F-measure is the main metric to compare among four models SVM-B, CTK, DTK and CTK + DTK.

In-Domain Results: Table 3 summarizes the results of each domain in four metrics for each method. SVM-B performed worst in all of the domains in F-measure. Our method CTK + DTK improves F-measure of SVM-B method by 6.2%, 8.2%, 3.7%, 1.1% and 3.3% in "DVD Player", "Cell Phone", "Digital Camera", "Diaper" and "MP3 Player", respectively. Therefore, our CTK + DTK method is better than SVM-B in in-domain evaluation. In addition, CTK + DTK method beats CTK in "Cell Phone", "Digital Camera" and "MP3 Player" domain and achieves competitive performance in "DVD Player" and "Diaper" domain. Thus, we can concluded that CTK + DTK is better than CTK method. Finally, CTK + DTK method is better than DTK in "DVD Player" and "MP3 Player" domain, comparable in "Cell Phone" and "Diaper" domain. To sum, DTK and CTK + DTK are the best methods for aspect-opinion relation extraction in in-domain evaluation.

Cross-Domain Results: The results of four methods in cross-domain are shown in Table 4. Our method CTK + DTK outperformed the baseline SVM-B in all domains in F-measure. Improvements of 5.4%, 2.5% and 3.9% of F-measure are found in "DVD Player", "Diaper" and "MP3 Player" domain, respectively. Therefore, our CTK + DTK method is better than SVM-B. In addition, CTK + DTK is better than CTK by 1.3%, 1.6% and 4.5% F-measure in each domain. Finally, compared with DTK, CTK + DTK shows 2.1% and 1.9% F-measure improvement in "DVD Player" and "Diaper" domain, and achieves competitive performance in "MP3 Player" domain. Therefore, we can conclude that CTK + DTK is the best method for extraction of aspect-opinion relations in the cross-domain evaluation.

Table 3. In-domain Results of Aspect-Opinion Relation Extraction

Domain	Metric	SVM-B	CTK	DTK	CTK + DTK
DVD Player	A	0.804	**0.902**	0.863	**0.902**
	P	**0.905**	0.898	0.878	0.898
	R	0.864	**1.00**	0.977	**1.00**
	F	0.884	**0.946**	0.925	**0.946**
Cell Phone	A	0.728	0.712	**0.837**	0.815
	P	0.817	0.704	**0.884**	0.811
	R	0.764	**0.984**	0.870	0.943
	F	0.790	0.820	**0.877**	0.872
Digital Camera	A	0.721	0.652	**0.756**	0.709
	P	**0.798**	0.648	0.741	0.690
	R	0.746	**0.980**	0.940	0.975
	F	0.771	0.780	**0.829**	0.808
Diaper	A	**0.783**	0.739	0.739	0.739
	P	**0.929**	0.739	0.739	0.739
	R	0.765	**1.00**	**1.00**	**1.00**
	F	0.839	**0.850**	**0.850**	**0.850**
MP3 Player	A	0.800	0.705	0.800	**0.813**
	P	**0.923**	0.718	0.832	0.818
	R	0.769	**0.932**	0.885	**0.932**
	F	0.839	0.811	0.857	**0.872**

Table 4. Cross-domain Results of Aspect-Opinion Relation Extraction

Domain	Metric	SVM-B	CTK	DTK	CTK + DTK
DVD Player	A	0.749	0.778	0.787	**0.808**
	P	**0.863**	0.793	0.859	0.834
	R	0.787	**0.952**	0.855	0.928
	F	0.824	0.865	0.857	**0.878**
Diaper	A	0.804	0.780	0.794	**0.812**
	P	**0.910**	0.786	0.846	0.823
	R	0.810	**0.964**	0.881	0.949
	F	0.857	0.866	0.863	**0.882**
MP3 Player	A	0.765	0.686	**0.792**	0.772
	P	**0.833**	0.683	0.805	0.760
	R	0.774	**0.954**	0.894	0.942
	F	0.802	0.796	**0.847**	0.841

4 Aspect-Based Sentiment Classification Based on Relation Extraction

As mentioned in Section 1, aspect-based sentiment analysis is a task to identify the sentiment categories for a given aspect in a sentence. In this section, we tried to integrate the relation extraction model to aspect-based sentiment analysis. Intuitively, not all opinion words in the sentence represent emotion on the given aspect. Therefore, CTK + DTK described in Subsection 3.4 will be used to identify the strong relations between aspect and opinion entities.

4.1 Aspect-Based Sentiment Analysis Without Relation Extraction: A Baseline Model

We used a popular algorithm for calculating a score of a given aspect [10,14]. Even though this algorithm is simple, it can perform well in many cases. Given a sentence, which contains a set of aspects $A = \{a_1, \cdots, a_m\}$ and a set of opinion words $OW = \{ow_1, \cdots, ow_n\}$, the sentiment score for each aspect a_i is calculated as in Equation (9). The closer between the aspect phrase and the opinion word, the higher affection of that opinion on the aspect. Therefore, the sentiment value of the aspect is defined as the summation over all opinion values divided by their distances to that aspect. The aspect is categorized as positive, negative and neutral if $sentimentValue(a_i)$ is greater than 0.25, less than -0.25 and other.

$$sentimentValue(a_i) = \sum_{j=1}^{|OW|} \frac{opinionValue(ow_j)}{distance(a_i, ow_j)} \tag{9}$$

Opinion words were identified based on SentiWordNet [1] that is a lexical resource for opinion mining. Three sentiment scores (positivity, objectivity and negativity) are assigned to each word in SentiWordNet. $opinionValue(ow)$ is defined as Equation (10).

$$opinionValue(ow) = \frac{positivityScore - negativityScore}{positivityScore + negativityScore} \tag{10}$$

4.2 Aspect-Based Sentiment Analysis with Relation Extraction

We proposed a new method for identifying the sentiment category of a given aspect based on the aspect-opinion relations. The method supposes that the opinion words having relation with the aspect will more influence the polarity of it. Identification of the aspect-opinion relations in the sentence can help to improve the prediction of sentiment categories of the given aspect. In other words, aspect-opinion relation extraction enables us to distinguish opinion words of the target aspect and other aspects.

For a given sentence, the aspect-opinion relations were extracted by using the tree kernel method CTK + DTK. Then, we put more weight on the important opinion words in the sentiment score of the aspect as shown in Equation (11).

$$sentimentValue(a_i) = \sum_{j=1}^{|OW|} weight(a_i, ow_j) \cdot \frac{opinionValue(ow_j)}{distance(a_i, ow_j)} \tag{11}$$

The weight of opinion is calculated as:

$$weight(a, ow) = \begin{cases} 2 \text{ if } r(a, ow) = 1 \\ 1 \text{ otherwise} \end{cases} \tag{12}$$

4.3 Evaluation of Aspect-Based Sentiment Analysis

Dataset: Because the data used in Subsection 3.5 is not annotated with the sentiment categories of the aspects, we used the restaurant reviews dataset in SemEval2014 Task 4 [2]. It consists of over 3000 English sentences of the restaurant reviews. For each sentence, the aspect terms and their polarity are annotated. The possible values of the polarity field are "positive", "negative", "neutral" and "conflict". Since, we do not deal with "conflict" category in our model, 84 sentences including the aspects with "conflict" polarity are removed from the dataset. CTK + DTK was trained from the sentences in "Digital Camera" and "Cell Phone" domains in Wu et al.'s dataset.

To investigate the effectiveness of integrating aspect-opinion relation extraction to aspect-based sentiment analysis, we compared the model with and without relation extraction (we call "ASA with RE" and "ASA w/o RE", respectively). Table 5 shows Accuracy, Precision, Recall and F-measure for all aspect phrases in the dataset. Precision, Recall and F-measure are the average for three polarity categories weighted by the number of true instances. Accuracy of "ASA with RE" was 0.523 [3]. It outperformed the baseline by 5.8%. Furthermore, Recall and F-measure of "ASA with RE" were greatly improved. Table 6 shows the results of the sentence-based evaluation. Exact Match Ratio (EMR) is defined as a ratio of correctly classified sentences where the polarity of all aspects in the sentence are successfully identified. Partial Match Ratio (PMR) is the average of the partial matching scores of individual sentences, that is the proportion of the number of correctly classified aspects to all aspects in the sentence. "ASA with RE" was better 3.8% EMR and 3.6% PMR than "ASA w/o RE". From these results, we can conclude that using the aspect-opinion relation extraction is useful for sentiment analysis of aspects.

Table 5. Results of Aspect-based Sentiment Identification

Metric	ASA w/o RE	ASA with RE
A	0.465	**0.523**
P	**0.532**	**0.532**
R	0.465	**0.523**
F	0.477	**0.523**

Table 6. Results of Sentence-based Sentiment Identification

Metric	ASA w/o RE	ASA with RE
EMR	0.596	**0.634**
PMR	0.666	**0.702**

[2] `http://alt.qcri.org/semeval2014/task4/`

[3] Accuracies of participating systems in SemEval 2014 Task 4 were between 0.42 and 0.81. However, these results cannot be simply compared to Table 5. Our method was evaluated on a training data of the task, while the participating systems were trained on it and evaluated on a separate test data. Furthermore, our system was not developed by supervised machine learning, unlike the top participating system.

5 Conclusion

We applied two kernels of constituent and dependency trees and proposed the new tree kernel for aspect-opinion relation extraction. The results showed that the models using tree kernels outperformed the baseline SVM-B. Furthermore, we proposed the new method for identifying the sentiment categories of the aspects in the sentences with the relation extraction module. Our method achieved better performance in almost all metrics compared to the method without relation extraction.

Our tree kernel based model for aspect-opinion relation extraction can be further improved by using semantic information from semantic trees. In addition, combining the syntactic tree and semantic tree for calculating tree kernel will be explored in our future work.

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC), vol. 10, pp. 2200–2204 (2010)
2. Bunescu, R.C., Mooney, R.J.: A shortest path dependency kernel for relation extraction. In: Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP), pp. 724–731. ACL (2005)
3. Collins, M., Duffy, N.: New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics (ACL), pp. 263–270. ACL (2002)
4. Culotta, A., Sorensen, J.S.: Dependency tree kernels for relation extraction. In: Scott, D., Daelemans, W., Walker, M.A. (eds.) Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL), pp. 423–429. ACL (2004)
5. Jo, Y., Oh, A.H.: Aspect and sentiment unification model for online review analysis. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, pp. 815–824. ACM (2011)
6. Joachims, T.: Text categorization with support vector machines. In: Nédellec, C., Rouveirol, C. (eds.) ECML 1998. LNCS, vol. 1398, pp. 137–142. Springer, Heidelberg (1998)
7. Kobayashi, N., Inui, K., Matsumoto, Y.: Extracting aspect-evaluation and aspect-of relations in opinion mining. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 1065–1074. ACL (2007)
8. Lakkaraju, H., Bhattacharyya, C., Bhattacharya, I., Merugu, S.: Exploiting coherence for the simultaneous discovery of latent facets and associated sentiments. In: Proceedings of the Eleventh SIAM International Conference on Data Mining (SDM), pp. 498–509. SIAM/Omnipress (2011)
9. Lin, C., He, Y.: Joint sentiment/topic model for sentiment analysis. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, pp. 375–384. ACM (2009)

10. Liu, B., Zhang, L.: A survey of opinion mining and sentiment analysis. In: Mining Text Data, pp. 415–463. Springer (2012)
11. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations (ACL), pp. 55–60 (2014)
12. Nguyen, T.H., Shirai, K.: Text classification of technical papers based on text segmentation. In: Gulliver, T.A., Secord, N.P. (eds.) Information Theory 1993. LNCS, vol. 7934, pp. 278–284. Springer, Heidelberg (2003)
13. Nguyen, T.T., Moschitti, A., Riccardi, G.: Convolution kernels on constituent, dependency and sequential structures for relation extraction. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1378–1387. ACL (2009)
14. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1-2), 1–135 (2008)
15. Reichartz, F., Korte, H., Paass, G.: Dependency tree kernels for relation extraction from natural language text. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009, Part II. LNCS, vol. 5782, pp. 270–285. Springer, Heidelberg (2009)
16. Smola, A.J., Vishwanathan, S.: Fast kernels for string and tree matching. In: Becker, S., Thrun, S., Obermayer, K. (eds.) Advances in Neural Information Processing Systems 15 (NIPS), pp. 585–592. MIT Press (2003)
17. Wu, Y., Zhang, Q., Huang, X., Wu, L.: Phrase dependency parsing for opinion mining. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1533–1541. ACL (2009)
18. Zelenko, D., Aone, C., Richardella, A.: Kernel methods for relation extraction. Journal of Machine Learning Research (JMLR) 3, 1083–1106 (2003)