2018 3rd International Conference on Information Technology and Industrial Automation (ICITIA 2018)
ISBN: 978-1-60595-607-7

Aspect-specific Sentiment Classification Method Based on High-dimensional Representation

Hanqian Wu, Lu Cheng, Jie Li, Jingjing Wang and Jue Xie

ABSTRACT

Aspect-specific sentiment classification is a fine grained sentiment classification task. The traditional coarse-grained sentiment analysis methods only identify the consumers' sentiment polarity towards the product as a whole, but ignores the important attribute information, which leads to the inability to refine consumer preferences and clarify the advantages and disadvantages of commodity attributes. To solve this problem, we use the review text and its specific aspect information to construct a multi-level, high-dimensional deep neural network model. First, the clause segmentation algorithm is used to divide the review text into several clauses; secondly, the words in the clause are encoded by the bidirectional long short term memory neural network, and the vector representation of the clauses is obtained. Finally, the vector representation of the whole review text is obtained by using the bidirectional long short term memory neural network to code all the clauses. And then the sentiment polarity of review text is obtained by the softmax layer. The experimental result shows that the proposed method can effectively improve the performance of sentiment classification.[1]

KEYWORDS

Aspect-specific; Sentiment Classification; High-dimensional Text Representation; lstm.

[1] Hanqian Wu, Jie Li, Computer Science and Engineering, Southeast University, Nanjing, China
Lu Cheng, Jue Xie, Southeast University-Monash University Joint Graduate School, Southeast University, Suzhou, China
Jingjing Wang, Computer Science and Technology, Soochow University, Suzhou, China

INTRODUCTION

In the past decade, sentiment analysis has attracted increasing attention in the field of natural language processing and data mining for its wide range of applications and inherent challenges [1]. Aspect-level sentiment classification is a fine-grained sentiment analysis research task whose purpose is to identify the sentiment polarity expressed in the review text.

Early researches on sentiment analysis [2] are mainly divided into two categories. One category is to use a traditional supervised learning algorithm to construct a series of features (such as: bag of words, sentiment dictionary) to train the classifier (such as: support vector machine, naive Bayes) [3]. For example, Jiang et al. [4] proposed an SVM-based sentiment analysis method to solve the problem related to the target entity. The other category pays attention to the semantic orientation [5]. In recent years, neural networks have gradually shown their superiority in the field of sentiment analysis, such as recurrent neural networks. Wang et al. [6] pointed out that the emotional polarity depends on both the content and the attributes that appear in the sentence. Therefore, they proposed a long short term memory neural network based on attention mechanism to capture the hidden relationship between specific attributes and emotional polarity in aspect-level sentiment analysis tasks.

Due to the unsatisfactory classification accuracy in existing methods of sentiment classification, this paper proposes a method of aspect-specific sentiment classification based on high-dimensional representation.

MODEL EXPLANATION

Our model divides the clauses of the review text from three different dimensions of words, clauses and sentences.

Clause Segmentation

The basic idea of the clause segmentation algorithm is to use punctuation and conjunctions (collectively referred to as separators). For example, a review text is "The food was great and tasty, especially the hot dogs are top notch, but the sitting space was too small, I don't like being cramp in a corner." To perform the segmentation, the phrase "great and tasty" in the review text can be incorrectly divided into two clauses by the separator "and". Therefore, the minnum parameter is defined to limit the least number of words in a clause. In addition, the maxnum parameter is set to ensure that each sentence is divided into clauses with the same number.

High-Dimensional Text Representation Model

The high-dimensional text representation model mainly learns the relationship between the review text and the specific attribute from three different dimensions of words, clauses and sentences. Figure 1 shows the overall framework of the high-dimensional text representation model.

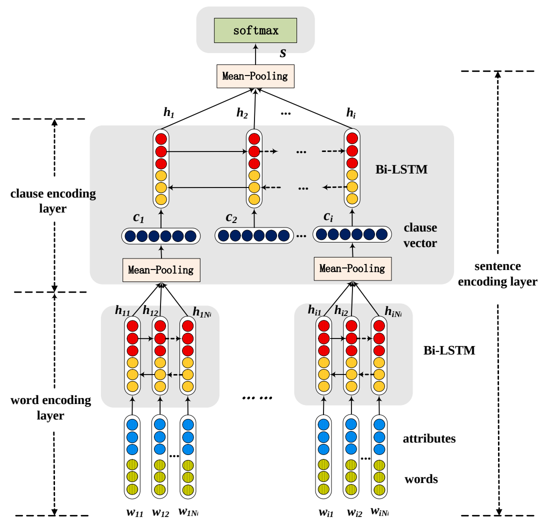

Figure 1. The overall architecture based on high-dimensional text representation.

WORD ENCODING LAYER

Suppose the review text contains a total of C clauses, where c_i is used to represent thei-th clause. Each clause contains N_i words, I_{ij} means the words appearing in the j-thposition ofthei-thclause, where$j \in [1, N_i]$. The words in the clause $c_i$are represented by $w_{ij} \in \boldsymbol{R}$d, where$j \in [1, N_i]$, $w_{ij} = E_w \cdot I_{ij}$are stored in a word embedding matrix, where $E_w \in \boldsymbol{R}$d×|V|, d represents dimension of the word vector, Vrepresents the vocabulary. Besides, the aspect category consists of two parts: the entity and the attribute. Specifically, assuming that the length of the entity string $e_1 = \{z_1, \ldots, z_{L_1}\}$ is L_1 . $x_n \in \boldsymbol{R}$ d' represents the d' dimensional vector representation of then-th word in the entity string. Correspondingly, the feature string is$e_2 = \{z_1, \ldots, z_{L_2}\}$. Usually the word vector representation has a linear structure, so we sum up the entity word vector and the feature word vector to get the final representation of the aspect word vector:

$$e_{aspect} = e_1 + e_2 = \frac{1}{L_1}\sum_{n=1}^{L_1} x_n + \frac{1}{L_2}\sum_{n=1}^{L_2} z_n \tag{1}$$

Then, the word vector is added to obtain the aspect expansion representation:

$$\widehat{w}_{ij} = w_{ij} \oplus e_{aspect} \tag{2}$$

In the above formula, $\widehat{w}_{ij} \in \mathbf{R}^{d+d'}$, where the dimension of $\widehat{w}_{ij}$ is $(d + d')$, $i \in [1, C]$, $j \in [1, N_i]$, $\oplus$represents the vector splicing operator, C indicates the number of clauses, $N_i$and indicates the number of words contained in the clausec_i.

Next, using the word vector $\widehat{w}_{ij}$ obtained above as input, the forward and backward words are encoded by Bi-directional Long Short Term Memory Neural Network (Bi-LSTM). The forward LSTM is expressed as $\vec{f}$, the neural network reads the words in the c_i clause from front $I_{i,1}$ to back I_{i,N_i}, the corresponding backward LSTM is represented as $\overleftarrow{f}$, which reads words from backend I_{i,N_i} to front $I_{i,1}$.

$$\vec{h}_{ij} = \overrightarrow{LSTM}\left(\widehat{w}_{ij}\right);\ \ \mathrm{i}\in[1, \mathrm{C}],\ \ \mathrm{j}\in[1, \mathrm{Ni}] \tag{3}$$

$$\overleftarrow{h}_{ij} = \overleftarrow{LSTM}\left(\widehat{w}_{ij}\right);\ \ \mathrm{i}\in[1, \mathrm{C}],\ \ \mathrm{j}\in[\mathrm{Ni}, 1] \tag{4}$$

Then, the forward hidden layer state $\vec{h}_{ij}$ and the backward hidden layer state $\overleftarrow{h}_{ij}$ are spliced to obtain the final hidden layer state representation:

$$h_{ij} = \vec{h}_{ij} \oplus \overleftarrow{h}_{ij} \tag{5}$$

Finally, through the Mean-Pooling layer, the hidden layer state h_{ij} of each word I_{ij} is averaged to obtain the final representation of the clause:

$$c_i = \frac{1}{t}\sum_{t=1}^{T} h_{ij} \tag{6}$$

CLAUSE ENCODING LAYER

For the clause vector c_i obtained in the previous section, the Bi-LSTM is used:

$$\vec{h}_i = \overrightarrow{LSTM}(c_i);\ \mathrm{i}\in[1, \mathrm{C}] \tag{7}$$

$$\overleftarrow{h}_i = \overleftarrow{LSTM}(c_i);\ \mathrm{i}\in[1, \mathrm{C}] \tag{8}$$

The final hidden layer state representation of each clause c_i is obtained by splicing the forward hidden layer state $\vec{h}_i$ and the backward hidden layer state$\overleftarrow{h}_i$:

$$h_i = \vec{h}_i \oplus \overleftarrow{h}_i \tag{9}$$

Finally, the hidden layer state h_i is averaged through the Mean-Pooling layer:

$$s = \frac{1}{t}\sum_{t=1}^{T} h_i \tag{10}$$

SOFTMAX LAYER

The final representation s is entered into the $softmax$ classifier, then the category probability distribution for a specific aspect of the review text is obtained:

$$o = W_l \cdot s + b_l \tag{11}$$

In the above formula, $o \in \boldsymbol{R}$K is the output, W_l is the weight matrix, and b_l is the offset. The probability of a given sentence belongs to each category $k \in [1, K]$ is:

$$p_\theta = \frac{\exp(o_k)}{\sum_{t=1}^{K} \exp(o_t)} \tag{12}$$

In the above formula, θ represents all the parameters, and the category label of the highest probability calculated is used as the final category label of the review text.

Model Training

In this paper, we apply the Cross-Entropy Loss Function. Given training data x_t, a_t, y_t, where x_t represents the t-th sample to be predicted, a_t represents the attribute appearing in the sample, y_t indicates the true category label of the sample x_t with a particular attribute a_t. We consider our model as a black box function$\phi(x, a)$. The training goal is to minimize the loss function:

$$J(\theta) = -\sum_{t=1}^{M}\sum_{k=1}^{K} y_t^k \cdot log\phi(x_t, a_t) + \frac{l}{2}||\theta||_2^2 \tag{13}$$

In the above formula, M represents the number of training samples, K indicates the number of category labels, $l$indicates the L_2 regularization of the bias parameters.

The model parameters are optimized using the Adagrad optimization function [7], and the matrix and vector parameters are randomly initialized in $[-\sqrt{6/(r+c')}, \sqrt{6/(r+c')}]$, where r and c' are the number of rows and columns in the matrix respectively [8]. In order to avoid over-fitting during training, the Dropout strategy was adopted in the Bi-LSTM layer.

TABLE I. DISTRIBUTION OF RESTAURANT AND LAPTOP DATA SETS.

Dataset		Positive	Negative	Neutral	Total
Restaurant	Training	1198	403	53	1654
	Testing	454	346	45	845
Laptop	Training	1103	765	106	1974
	Testing	541	329	79	949

EXPERIMENTAL STUDY

Experiment Setup

DATA SETTINGS

In this paper, our datasets are Laptop and Restaurant in Task12 of SemeEval-2015 semantic evaluation. Each dataset consists of multiple user comments, each of which contains a list of attributes and the emotional polarity of each attribute, where the emotional polarity includes positive, neutral, and negative. Table I shows the distribution of data in the Laptop and Restaurant areas. In addition, we randomly selected 10% of the data from the training set as the development data set for adjusting the algorithm parameters, and selected Glove 2 as the pre-trained word vector.

SUPER PARAMETER SETTING

In the experiment, a uniform vector U(-0.01,0.01) was used to randomly generate word vector representation of unregistered words. The dimension of word vector and Bi-LSTM is 300. Other super parameters are adjusted according to the development data set. Specifically, the initial value of the learning rate is 0.1, the regularization weight is 10-5, and the dropout rate is 0.25. In addition, in the clause segmentation algorithm, the parameter minnum is 3, and the parameter maxnum is set 4.

EVALUATION CRITERIA

In our experiment, we utilized accuracy to evaluate the model's performance.

TABLE II. COMPARISON OF RESULTS FOR DIFFERENT CLASSIFICATION METHODS.

Model	Restaurant	Laptop
Majority	0.537	0.570
LSTM	0.721	0.704
TC-LSTM	0.732	0.737
ATAE-LSTM	0.735	0.743
IAN	0.745	0.742
Hierarchical Bi-LSTM	0.763	0.763

Comparative Analysis

Table II shows the performance comparison between our proposed method and other benchmark methods, we can observe that the performance of the Majority algorithm is the worst. The classification accuracy constructed by the Majority algorithm in the Restaurant field and the Laptop field is 53.7% and 57.0%, respectively. All the other methods are based on the LSTM neural network model, and their classification performance is better than the Majority algorithm. The experimental results show that the LSTM model not only has the potential to automatically generate representations, but also can be used in attribute-level sentiment classification.

In addition, it can be seen from Table II that the classification accuracy of TC-LSTM, ATAE-LSTM and IAN are better than LSTM. This result confirms that it is helpful to consider attribute information when classifying emotions for specific attributes. Finally, we can see that our proposed Hierarchical Bi-LSTM method is superior to all the methods mentioned above, which highlights the importance of clause information.

CONCLUSIONS

This paper proposes an emotional classification method based on specific attributes of high-dimensional representation. First, we introduce the clause segmentation algorithm, which divides the comment text into several clauses. Then, from the three dimensions of words, clauses and sentences, the sentiment classification method based on the specific attributes of high-dimensional representation is introduced in detail. This method is constructed from three different dimensions of words, clauses and sentences, using the comment text and its specific attribute information. A multi-level, high-dimensional deep neural network model. The experimental results show that the proposed method achieves better classification performance and also highlights the importance of using clause information.

REFERENCES

1. Yanyan Zhao, Bing Qin, Ting Liu. Sentiment Analysis [J]. Journal of Software, 2010, 21(8): 1834-1848.
2. Veronica Perez-Rosas, Carmen Banea, and Rada Mihalcea. Learning Sentiment Lexicons in Spanish [C]//. In Proceedings of the Eighth International Conference on Language Resource and Evaluation, 2012: 3077-3081.
3. Hassan Ahmed, Vahed Qazvinian, and Dragomir R. Radev. What's with the Attitude? Identifying Sentences with Attitude in Online Discussions [C]. //In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Mit Stata Center, Massachusetts, USA, 2010: 1245-1255.
4. LongJ iang, Mo Yu, Ming Zhou, Et al. Target-Dependent Twitter Sentiment Classification[C]. //In Proceedings of the Association for Computational Linguistics: Human Language Technologies. Association for Computational Linguistics, 2011: 151-160.
5. Decheng Lou, Tianfang Yao. Semantic Polarity Analysis and Opinion Mining on Chinese Review Sentence [J]. Journal of Computer Applications, 2006, 26(11): 2622-2625.
6. Yequan Wang, Minlie Huang, Xiaoyan Zhu, Et al. Attention-Based LSTM for Aspect-Level Sentiment Classification [C].// In Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2016: 606-615.
7. John Duchi, Elad Hazan, and Yoram Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization [J]. Journal of Machine Learning Research, 2011, 12: 2121–2159.
8. Glorot Xavier, Bengio Yoshua. Understanding the Difficulty of Training Deep Feedforward Neural Networks [J]. Journal of Machine Learning Research, 2010, 9: 249–256.