Modeling Multi-Targets Sentiment Classification via Graph Convolutional Networks and Auxiliary Relation

Ao Feng[1], Zhengjie Gao[1, *], Xinyu Song[1], Ke Ke[2], Tianhao Xu[1] and Xuelei Zhang[1]

Abstract: Existing solutions do not work well when multi-targets coexist in a sentence. The reason is that the existing solution is usually to separate multiple targets and process them separately. If the original sentence has N target, the original sentence will be repeated for N times, and only one target will be processed each time. To some extent, this approach degenerates the fine-grained sentiment classification task into the sentence-level sentiment classification task, and the research method of processing the target separately ignores the internal relation and interaction between the targets. Based on the above considerations, we proposes to use Graph Convolutional Network (GCN) to model and process multi-targets appearing in sentences at the same time based on the positional relationship, and then to construct a graph of the sentiment relationship between targets based on the difference of the sentiment polarity between target words. In addition to the standard target-dependent sentiment classification task, an auxiliary node relation classification task is constructed. Experiments demonstrate that our model achieves new comparable performance on the benchmark datasets: SemEval-2014 Task 4, i.e., reviews for restaurants and laptops. Furthermore, the method of dividing the target words into isolated individuals has disadvantages, and the multi-task learning model is beneficial to enhance the feature extraction ability and expression ability of the model.

1 Introduction

With the development of the social economy, people's lives are increasingly dependent on the mining of large amounts of data. Sentiment Analysis is a general term for tasks such as sentiment subject recognition and sentiment polarity classification. In the sentiment classification task, the classification task of text-level and sentence-level giving a holistic evaluation of the review text is relatively more researched, while the fine-grained sentiment classification task is relatively less studied. In the fine-grained

---

[1] Department of Computer Science, Chengdu University of Information Technology, Chengdu, 610225, China.

[2] Central Washington University, Des Moines, WA 98198, USA.

* Corresponding Author: Zhengjie Gao. Email: gaozhengj@foxmail.com.

sentiment classification task, there are two definitions, one is to be more detailed in the division of emotional granularity, for example, from the general positive, negative and neutral triages, divided into anger, disgust, fear, happiness, like, sadness, surprise, such a seven-category task [Rathnayaka, Abeysinghe, Samarajeewa et al. (2019)]; another definition refers to the corresponding sentiment classification task for specific subjects in the comment text, called Target-Dependent Sentiment Classification (TDSC) or Aspect-Based Sentiment Classification (ABSC). The difference between the two is that objects will appear in the original text in the TDSC evaluation while the objects may appear after the original abstraction in the ABSC evaluation. Both are clearly defined in SemEval-2014 task 4 [Pontiki, Galanis, Pavlopoulos et al. (2014)]. In this paper, we focus on the Target-Dependent Sentiment Classification [Jiang, Yu, Zhou et al. (2011); Dong, Wei, Tan et al. (2014); Vo and Zhang (2015); Tang, Qin, Feng et al. (2015); Song, Wang, Jiang et al. (2019)]. As a special case of aspect-level sentiment classification, the targets in the target-dependent sentiment classification task is bound to appear in the text and the polarity of sentiments towards them needs to be identified separately. For example, given a sentence "While this is a pretty *place* in that overly cute French way, the ***food*** was insultingly horrible.", the sentiment polarity for "place" is positive, for "food" is negative. Another example, given a sentence "Not a large *place*, but it's cute and cozy.", the sentiment for "place" is conflict, as both negative (Not a large) and positive (cute and cozy) sentiments are expressed towards the same target.

In the previous method, the researchers usually split the multiple targets in a sentence into multiple instances for processing. This way ignores the correlation and influence between the targets, so in this paper we propose a new model called TSR-GCN, which uses Graph Convolutional Networks to model multi-targets in a sentence simultaneously based on the positional relationship, and we introduce an auxiliary relation classification task to further explore the sentiment polarity relation between targets (nodes of the graph). The experimental results show that our model can still achieve comparable performances with the current best results when the composition is relatively rough, indicating that the approach worth is further exploration and research.

2 Related work

In this section, we will review related works on Target-Dependent Sentiment Classification (TDSC) and Graph Convolutional Networks (GCN).

2.1 Conventional neural networks

Traditional approaches mainly focus on designing a set of features to train a classifier (e.g., SVM) for target-dependent sentiment classification [Jiang, Yu, Zhou et al. (2011); Wagner, Arora, Cortes et al. (2014); Kiritchenko, Zhu, Cherry et al. (2014)]. The traditional method of sentiment analysis needs to rely on the complex feature engineering, needs to spend a lot of manpower and resources, and the method is poor universal in the cross-domain. Multiple sentiment lexicons are built for this purpose [Neviarouskaya, Prendinger and Ishizuka (2009); Qiu, Bing, Bu et al. (2009); Taboada, Brooke, Tofiloski et al. (2011)].

With the development of deep learning, neural network models are of growing interests

for this Natural Language Processing (NLP) task because of neural networks' capacity of learning representation from data without feature engineering [Dong, Wei, Tan et al. (2014); Tang, Qin, Feng et al. (2015); Tang, Qin and Liu (2016); Wang, Huang, Zhao et al. (2016); Ma, Li, Zhang et al. (2017); Chen, Sun, Bing et al. (2017); Huang and Carley (2018); Zhang, Wang, Li et al. (2018); Song, Wang, Jiang et al. (2019); Sun, Huang and Qiu (2019)]. The mainstream neural networks methods are based on long short-term memory networks [Hochreiter and Schmidhuber (1997)], memory networks [Sukhbaatar, Weston, Fergus et al. (2015)] and attention mechanism [Bahdanau, Cho and Bengio (2014)]. Recursive neural networks [Dong, Wei, Tan et al. (2014)], and gated neural networks [Zhang, Zhang and Vo (2016); Xue and Li (2018)]. Convolutional neural networks [Huang and Carley (2018)] are used relatively rarely in this field.

More recently, the pre-trained language models, such as ULMFiT [Howard and Ruder (2018)], OpenAI GPT [Radford, Narasimhan, Salimans et al. (2018)], ELMo [Peters, Neumann, Iyyer et al. (2018)] and BERT [Devlin, Chang, Lee et al. (2018)], have shown great power in the semantic expression of text. In particular, BERT achieved excellent results in sentence-level sentiment classification. Song et al. [Song, Wang, Jiang et al. (2019)] proposed an Attentional Encoder Network (AEN) without cyclic recursive structure, and used Attentional Encoder method to model between context and target words. According to the given target, Sun et al. [Sun, Huang and Qiu (2019)] transformed the target-dependent sentiment classification problem into a sentence-pair classification task by constructing an auxiliary question. Xu et al. [Xu, Liu, Shu et al. (2019)] believe that customer reviews can be transformed into a large-scale source of knowledge that can then be used to answer users' questions. A new task that is named Review Reading comprehensions (RRC), Xu et al. [Xu, Liu, Shu et al. (2019)] through exploration, fine-tuning BERT model to further improve the expressive force of RRC task, and then will be based on specific target sentiment classification problem into a special Machine Reading Comprehension (Machine Reading comprehensions, MRC) problems, including all the issues related to sentiment tendency of the given target. Hazarika et al. [Hazarika, Poria, Vij et al. (2018)] processed multiple targets with LSTM network in the first stage, and then used LSTM to aggregate each group of features in the second stage, indicating that the target words in the previous text would affect the target words in the following text. Ma et al. [Ma, Zeng, Peng et al. (2019)] introduced positional attention. In the first stage, the model processed the target words one by one, and in the second stage, the model integrated multiple target words in the whole sentence simultaneously.

2.2 Graph convolutional networks

As shown in the Fig. 1, the left picture is a 2D Convolutional Neural Network (CNN). Its input is a matrix of 4 rows and 4 columns. The convolution operation of the entire input is realized by gradually moving the convolution kernel. The input on the right is a graph network, whose structure and connections are irregular and can't implement convolution operation like CNN. How to perform the convolution operation on the graph structure data, Defferrard et al. [Defferrard, Bresson, and Vandergheynst (2016)] present a formulation of CNN in the context of spectral graph theory, which provides the necessary mathematical background and efficient numerical schemes to design fast localized convolutional filters on graphs. Defferrard et al. [Defferrard, Bresson and Vandergheynst

(2016)] employed Graph Convolutional Networks (GCN) in text classification tasks and outperformed the traditional CNN models. Kipf et al. [Kipf and Welling (2016)] propose a more general GCN model, and experiments on citation networks and knowledge graph data sets have achieved excellent results. By stacking GCN layers, the hidden state of neighbor nodes at the current time is used as part of input to generate the hidden state of center nodes at the next time until the change of hidden state of each node is very small, and the information flow of the whole graph tends to be stable. So far, each node has aggregated its neighbor's information. GNN is widely used in various fields, such as relation extraction [Zhang, Qi and Manning (2018); Zhu, Lin, Liu et al. (2019)], aspect-based sentiment classification [Zhang, Li and Song (2019); Zhaoa, Houb and Wua (2019); Huang and Carley (2019); Hou, Huang, Wang et al. (2019)], text classification [Huang, Ma, Li et al. (2019); Yao, Mao and Luo (2019)] etc.
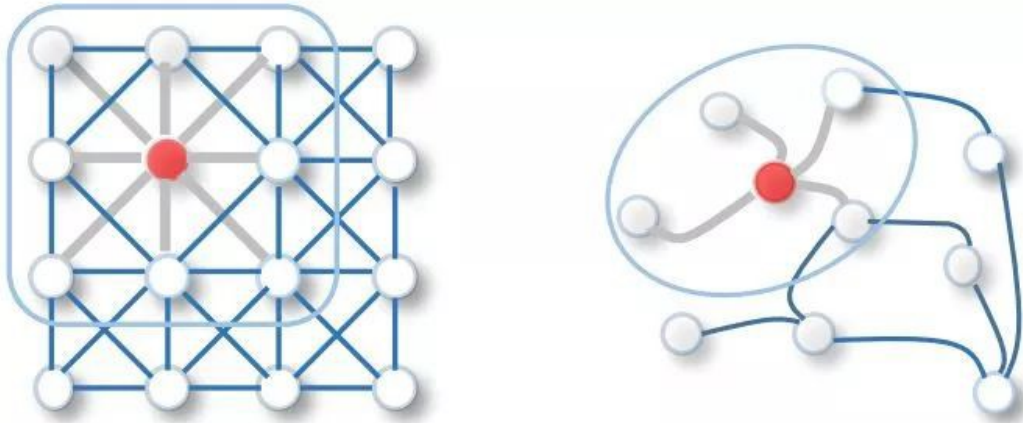

Figure 1: Graph convolutional networks (GCN)

Zhaoa et al. [Zhaoa, Houb and Wua (2019)] applied Graph Convolutional Network in the field of fine-grained sentiment classification earlier. They took the target word as the node of the graph and proposed two methods of composition, one is to connect the nodes based on the right and left adjacent positions, and the other is to connect the nodes in pairs globally. Both of the two methods of composition contain the self-loop of the node (that is, the node itself is connected with an edge).

3 Our approach

3.1 Problem definition and notations

A target-dependent sentiment classification task usually predicts the sentiment polarity of a tuple (s, t) which consists of a sentence and a target. The difference between multi-targets sentiment classification and target-dependent sentiment classification in the general sense is that the former will be processed at the same time when there are multiple targets in a sentence, while the latter will be processed separately. The sentence

$$s = [w_1, \ldots, w_i, \ldots, w_n] \tag{1}$$

consists of n words, and the number of targets in each sentence is at least 1 and less than

n. As shown below, the target t_k contains m_k words and any target word are a subsequence of the sentence s, the intersection between any two target words is empty.

$$t_k = \left[w_{i_k}, w_{i_k+1}, \ldots, w_{i_k+m_k-1}\right], k \in [1,p], 1 \leq p < n \tag{2}$$

$$t_k \subseteq s \ and \ t_k \neq s \tag{3}$$

$$t_a \cap t_b = \emptyset, a \in [1,p], b \in [1,p], a \neq b \tag{4}$$

The goal of this task is to determine the sentiment polarity y_k of sentence s towards the target t_k, where

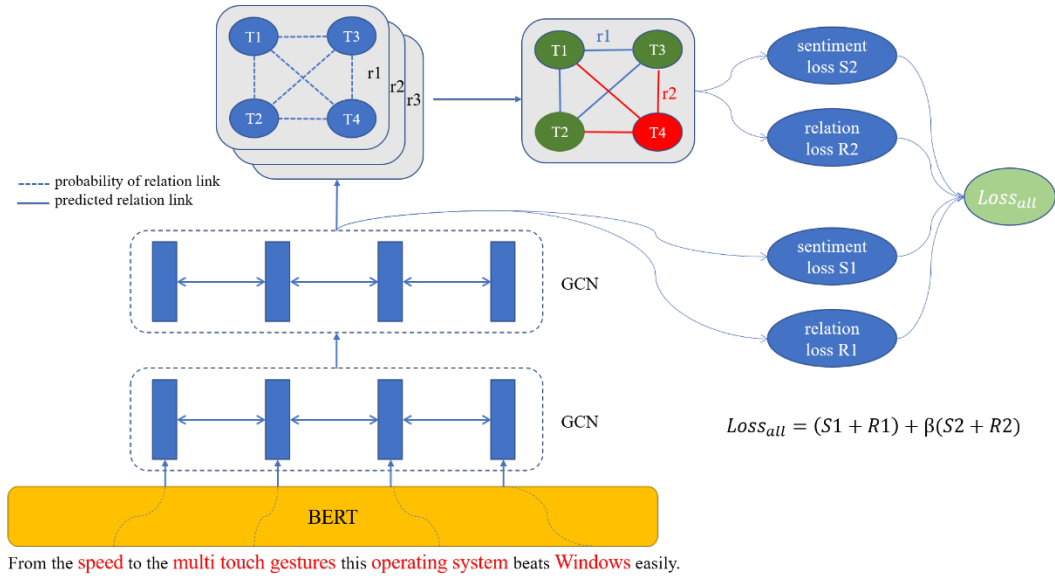$$y_k \in \{positive, negative, neutral, conflict\} \tag{5}$$

Figure 2: The model TSR-GCN for multi-targets sentiment classification via graph convolutional networks and auxiliary relation

3.2 Target representation

As shown in Fig. 2, TSR-GCN (Target Sentiment and Relation-Graph Convolutional Networks) is the model proposed by us. In the model, we use the pre-trained BERT model to get the word vector and fine-tune the $BERT_{base}$ model in the process of training the model TSR-GCN. In the BERT model, after WordPiece [Wu, Schuster and Chen (2016)] segmentation, the target word will be divided into multiple sub-word units. The output of the corresponding position in the last transformer encoding layer will be taken as the feature representation of the target, and max-pooling will be used to extract the significant feature as the vector representation of the entire target word.

3.3 Graph convolutional networks

We construct a graph to capture the sentiment dependencies between multi-targets in one sentence, where each node is regarded as a target and each edge is treated as the

sentiment dependency relation. As shown in Fig. 3. A graph is a set of N nodes connected via a set of edges. If two nodes are connected by an edge, it means that the two nodes are neighboring to each other. Formally, given a node v, we use $N(v)$ to denote all neighbors of v. The adjacency matrix of a graph A encodes graph topology, where each element A_{ij} represents an edge from node i to node j. If the value is 1, means that there is an edge between node i and node j. If the value is 0, there is no edge to join between node i and node j.

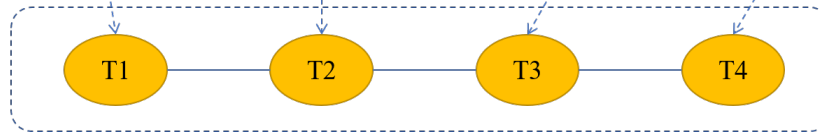From the speed to the multi touch gestures this operating system beats Windows easily.

Figure 3: The way of construct graph in our method. T1, T2, T3, and T4 denote four targets in a sentence

$$A_{ij} = \begin{cases} 1, & if\ i = j \\ 1, & if\ i+1 == j\ or\ i == j+1 \\ 0, & otherwise \end{cases} \tag{6}$$

A GCN layer propagates the node features h^l at layer l, using a function $f(A)$ of the adjacency matrix and has an output given by

$$h^{l+1} = \sigma(f(A) \cdot h^l \cdot W^l + b^l) \tag{7}$$

where W is the weight matrix and b is the bias, W and b are learned weights parameters. σ is a nonlinear activation function, where ReLU is used by us. f is called propagation rule. There are usually three rules from Li et al. [Li, Tarlow, Brockschmidt et al. (2015); Kipf and Welling (2016); Hamilton, Ying and Leskovec (2017)] as follows:

$$f_1 = A \tag{8}$$

$$f_2 = D^{-1}A \tag{9}$$

$$f_3 = D^{-1/2}AD^{-1/2} \tag{10}$$

where D is a degree matrix that defined as follows:

$$D_{ij} = \begin{cases} \sum_{i=1}^{p} A_{ij}, & if \ i = j \\ 0, & otherwise \end{cases} \tag{11}$$

Different processing rules can obtain different characteristics of nodes in the graph. Refer to Dehmamy et al. [Dehmamy, Barabási and Yu (2019)], we also combine three different GCN propagation modules and residual connection into our model.

$$H = \ h^1 \oplus \ldots \ldots \oplus h^l \oplus \sigma\big(f_1 \cdot h^l \cdot W^l + b^l\big) \oplus \sigma\big(f_2 \cdot h^l \cdot W^l + b^l\big)$$
$$\oplus \sigma(f_3 \cdot h^l \cdot W^l + b^l) \tag{12}$$

The outputs of the modules are concatenated and fed into a fully connected layer.

3.4 Sentiment classification and auxiliary relation

By classifying the relations among nodes, a multi-task joint learning model is constructed. In relation classification, we do not rely on the existing adjacency matrix (dependent edges) to directly predict the relation between all nodes.
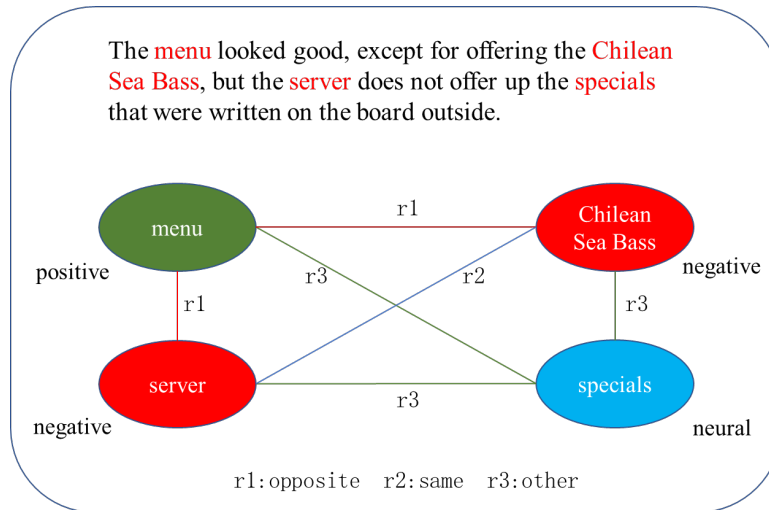

Figure 4: An example of how to construct the sentiment relation between nodes

As shown in Fig. 4, the relation r we designed can be divided into three types according to the difference in sentiment polarity between nodes, the same, the opposite, and others. For each relation r, the model can learn weight matrices W_r^1, W_r^2, W_r^3 and calculate the relation tendency score S as

$$S_{(t_a, r, t_b)} = W_r^3 \sigma(W_r^1 h_{t_a} \oplus W_r^2 h_{t_b}) \tag{13}$$

$$P_r(t_a, t_b) = softmax\big(S_{(t_a, r, t_b)}\big) \tag{14}$$

where $S_{(t_a, r, t_b)}$ represents the relation tendency score for targets pair (t_a, t_b) under relation r. We apply the softmax function to $S_{(t_a, r, t_b)}$, yielding $P_r(t_a, t_b)$, which represents the probability of each relation r for (t_a, t_b). With $P_r(t_a, t_b)$, it will be used to calculate the relation categorical loss $R1$.

Then, GCN is used in each graph, and the influence degree between different relations and nodes is taken as the comprehensive target feature. The process is as follows:

$$h_{t_a}^{l+1} = \sigma\left(\sum_{b \in T} \sum_{r \in R} P_r(t_a, t_b) \times \big(W_r^l h_{t_b}^l + b_r^l\big)\right) + h_{t_a}^l \tag{15}$$

where $P_r(t_a, t_b)$ represents the edge weight. W_r and b_r means the GCN weight under relation r. T includes all targets and R contains all relations.

As illustrated in Fig. 2, we use two types of loss in our model TSR-GCN: sentiment loss and relation loss, both of which belong to categorical loss. For sentiment loss, we use the *Positive*, *Negative*, *Neutral* and *Conflict* (The *conflict* tag appears only in the four-category task) as the ground-truth labels. Every target belongs to one of the three or four classes. The ground-truth sentiment labels for sentiment loss $S1$ and sentiment loss $S2$ are the same. We use cross-entropy as the categorical loss function during training.

For relation loss, we feed in a one-hot relation vector as the ground truth of $P_r(t_a, t_b)$ for each target pair (t_a, t_b). In our model, we designed three relations: opposite, similar, and others. If one is positive and the other is negative, the relation is the opposite. If the sentient polarity of the two is the same, the relation is similar. In other cases, the relation belongs to other categories. The ground-truth relation vectors for relation loss of $R1$ and relation loss $R2$ are the same. For relation loss, we also use cross-entropy as the categorical loss function during training.

For both sentiment loss and relation loss, we add an additional weight parameter to balance the loss before and after the two stages. Finally, the total loss is calculated as the sum of all sentiment loss and relation loss:

$$Loss_{all} = (S1 + R1) + \beta(S2 + R2) \tag{16}$$

where β is a weight parameter. Our goal is to minimize the $Loss_{all}$ during model training. In our model, we set β to 3 which was referenced from Fu et al. [Fu, Li and Ma (2019)].

4 Experiments

4.1 Datasets

Tab. 1 shows the statistics of the dataset restaurant and laptop which from SemEval-2014 task 4[2] [Pontiki, Galanis, Pavlopoulos et al. (2014)]. These two datasets will be used in our experiments to verify the validity of our proposed model. The definition of conflict label is that there is both positive and negative polarity for the same target in a sentence.

It is worth noting that "conflict" samples are few in the dataset, which will make the dataset very unbalanced in the process. So some existing work [Tang, Qin and Liu (2016); Wang, Huang, Zhao et al. (2016); Chen, Sun, Bing et al. (2017); Huang and Carley (2018); Song, Wang, Jiang et al. (2019); Xu, Liu, Shu et al. (2019)] remove "conflicting" samples from the data.

Table 1: Statistics of the experiment datasets

Dataset	Positive	Negative	Neutral	Conflict	Total
Laptop-Train	987	866	460	45	2358
Laptop-Test	341	128	169	16	654
Restaurant-Train	2164	805	633	91	3693
Restaurant-Test	728	196	196	14	1134

4.2 Experiment settings

In our experiment, we use PyTorch [Paszke, Gross, Chintala et al. (2017)] to implemented all models. And fine-tuning the model of $BERT_{base}$[3]. Hyperparameters in the experiment are displayed in Tab. 2. The dropout rate is 0.1, the batch size is 32, the learning rate is 2e-5. max sequence length is 128, the max epoch number is 6, and the size of a hidden layer in GCN is 256.

Table 2: Statistics of the experiment datasets

Parameter	Value
Dropout rate	0.1
Batch size	32
Learning rate	2e-5
Max epoch	6
Max sequence length	128
Optimizer	Adam
GCN hidden size	256
β	3
Number of GCN layer	2
Activation function	ReLU

---

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

[3] https://storage.googleapis.com/bert_models/2018_10_18/uncased_L-12_H-768_A-12.zip

4.3 Results

We use the classification accuracy metric to measure the performance of our model and previous methods. To demonstrate the effectiveness of our model, we compare it to a number of baseline methods, as shown below:

In the Tab. 3, 4-way stands for 4-way classification, i.e., positive, negative, neutral and conflict. 3-way keeps only 3 classes, with conflict data removed from SemEval-2014 datasets. The results with "♭" from BERT-PT paper [Xu, Liu, Shu et al. (2019)], with "‡" are copied from the AEN-BERT paper [Song, Wang, Jiang et al. (2019)], "♮" from GCAE [Xue and Li (2018)], with "†" are retrieved from SDGCN-BERT [Zhaoa, Houb and Wua (2019)], with "♯" are from Hazarika et al. [Hazarika, Poria, Vij et al. (2018)], and those with "℘" are from Ma et al. [Ma, Zeng, Peng et al. (2019)]. "-" indicates not reported in the original paper. For our method or re-implementations from others' code, we run the program for 10 times with random initialization, and show "mean $\pm$ std" as its performance. Best and second-best scores in each column are shown in bold and underlined, respectively.

TD-LSTM Tang et al. [Tang, Qin, Feng et al. (2015)] uses two one-way LSTM networks to model the preceding and the following text of the target word, including the target word. Based on the target word, the direction of the LSTM network on the left is from the beginning of the clause to the target word, and the direction of the LSTM network on the right is from the end of the clause to the target word. The hidden layer states of the two LSTM networks are fused by splicing and then used for the final classification layer.

Table 3: The experiment results of classification accuracy on the Laptop and Restaurant dataset

Method	Laptop		Restaurant	
	3-way	**4-way**	**3-way**	**4-way**
TD-LSTM	68.13‡	62.23$\pm$0.92♮	75.63‡	73.44$\pm$1.17♮
ATAE-LSTM	68.70‡	64.38$\pm$4.52♮	77.20‡	73.74$\pm$3.01♮
MemNet	70.33‡	64.09‡	78.16‡	65.83‡
RAM	74.49‡	71.35‡	80.23‡	70.80‡
IAN	72.10‡	68.49$\pm$0.57♮	78.60‡	76.34$\pm$0.27♮
GCAE	-	69.14$\pm$0.32♮	-	77.28$\pm$0.32♮
Hazarika's model	72.5♯	-	79.0♯	-
Ma's model	73.1℘	-	80.1℘	-
BERT-pair-QA-M	77.93$\pm$0.82	73.71$\pm$1.72	<u>85.12$\pm$0.41</u>	77.31$\pm$1.10
AEN-BERT	78.35$\pm$1.24	73.68$\pm$1.19	81.46$\pm$0.29	71.73$\pm$1.12
BERT-PT	78.07♭	75.08♭	84.95♭	76.96♭
SDGCN-BERT	**81.35**†	-	83.57†	-
TD-BERT	78.87$\pm$1.13	<u>76.62$\pm$0.90</u>	85.10$\pm$0.20	**84.37$\pm$0.28**
TSR-GCN (our model)	<u>79.47$\pm$0.25</u>	**77.25$\pm$0.10**	**86.07$\pm$0.14**	<u>82.54$\pm$0.32</u>

ATAE-LSTM Wang et al. [Wang, Huang, Zhao et al. (2016)] splices the word embedding of the target word and the word embedding of each word in the sentence as the input of LSTM network. This makes the output of the LSTM network contain information from the target word at every moment. Then the hidden layer output of each moment is concatenating with the word embedding of the target word again to further train the weight of the attention matrix. To some extent, the attention mechanism can capture the importance of different contexts for a given target word.

MemNet Tang et al. [Tang, Qin and Liu (2016)] is an end-to-end deep memory network that captures the importance of each word in context for a given target word through multiple computation layers.

RAM Chen et al. [Chen, Sun, Bing et al. (2017)] uses the multi-attention mechanism to obtain the sentiment characteristics between distant words, and then combines the output results of multiple attention through the recurrent neural network, thus enhancing the expression ability of MemNet.

IAN Ma et al. [Ma, Li, Zhang et al. (2017)] adopts two LSTM networks to model the sentence and the target word respectively, and then generates the attention vector of the hidden layer state of the sentence and the hidden layer state of the target word in a way of mutual supervision, and finally takes the concatenating result of the attention vector of the two as the input of the classification layer.

GCAE Xue et al. [Xue and Li (2018)] is a convolutional neural network with gated mechanism. The gated unit composed of Tanh and ReLU can selectively output corresponding sentiment characteristics according to the given target words.

BERT-pair-QA-M Sun et al. [Sun, Huang and Qiu (2019)] uses the given target words to construct an auxiliary question and fine-tune the BERT model by sentence classification task.

AEN-BERT Song et al. [Song, Wang, Jiang et al. (2019)] is an attention encoder network that avoids repetition, using an attention-based encoder to model between context and target words.

BERT-PT Xu et al. [Xu, Liu, Shu et al. (2019)] assumes that the task of sentiment classification of specific target words can be interpreted as a special Machine Reading Comprehension (MRC) problem [Rajpurkar, Zhang, Lopyrev et al. (2016); Rajpurkar, Jia and Liang (2018)], in which all problems are related to the sentiment polarity of a given target word.

Hazarika's model Hazarika et al. [Hazarika, Poria, Vij et al. (2018)] uses LSTM network to process multiple targets in the first stage, and then LSTM is used to aggregate each group of features in the second stage, indicating that the target words in the previous text will affect the target words in the following text.

Ma's model Ma et al. [Ma, Zeng, Peng et al. (2019)] introduces positional attention. In the first stage, the model processes the target words one by one, and in the second stage, the model integrates multiple target words in the whole sentence.

SDGCN-BERT Zhaoa et al. [Zhaoa, Houb and Wua (2019)] takes the target word as the node of the graph, and proposes two ways of composition: one is to connect the nodes based on the right and left adjacent positions, and the other is to connect the nodes in

pairs globally. Then the graph convolutional neural network is used to model multiple target words in a sentence at the same time, and then a bi-directional attention mechanism based on position coding is introduced to obtain the expression of specific target words.

TD-BERT [Gao, Feng, Song et al. (2019)] with positioned output at the target terms-based BERT model, it adopts a straightforward manner to incorporated target information. The model is not only simple but also very effective.

Experimental results are given in Tab. 2. We can find that models designed based on BERT have a significant improvement in the classification accuracy than models designed based word embedding, indicating that the BERT model is indeed more capable of semantic expression. The BERT model fully considers the context information of the sentence where the target word is located in the process of training, and the corpus of the training is larger.

Our model TSR-GCN achieve state-of-the-are performance on Laptop datasets 4 classification and Restaurant datasets 3 classifications, and in the other two classification tasks have also achieved good results in the second place. This shows that the design idea of our model is feasible, and we will get better results if we can design a more detailed auxiliary task such as relation classification.

5 Conclusion

In this paper, the proposed TSR-GCN model is used to deal with multi-targets simultaneously in the same sentence. The Graph Convolutional Networks can consider the internal relation between the target words, then we create a multi-task learning combination by constructing an auxiliary relation classification task, which makes the model have further improved the classification effect. Experiments support that our model shows very good results compared with other methods on the SemEval 2014 Task 4 datasets.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Bahdanau, D.; Cho, K.; Bengio, Y. (2014): Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

Chen, P.; Sun, Z.; Bing, L.; Yang, W. (2017): Recurrent attention network on memory for aspect sentiment analysis. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 452-461.

Defferrard, M.; Bresson, X.; Vandergheynst, P. (2016): Convolutional neural networks on graphs with fast localized spectral filtering. *Advances in Neural Information Processing Systems*, pp. 3844-3852.

Dehmamy, N.; Barabási, A. L.; Yu, R. (2019): Understanding the representation power of graph neural networks in learning graph topology. arXiv preprint arXiv:1907.05008.

Devlin, J.; Chang, M. W.; Lee, K.; Toutanova, K. (2018): Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dong, L.; Wei, F.; Tan, C.; Tang, D.; Zhou, M. et al. (2014): Adaptive recursive neural network for target-dependent twitter sentiment classification. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, vol. 2, pp. 49-54.

Fu, T. J.; Li, P. H.; Ma, W. Y. (2019): GraphRel: modeling text as relational graphs for joint entity and relation extraction. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1409-1418.

Gao, Z.; Feng, A.; Song, X.; Wu, X. (2019): Target-dependent sentiment classification with bert. *IEEE Access*, vol. 7, pp. 154290-154299.

Hamilton, W.; Ying, Z.; Leskovec, J. (2017): Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems*, pp. 1024-1034.

Hazarika, D.; Poria, S.; Vij, P.; Krishnamurthy, G.; Cambria, E. et al. (2018): Modeling inter-aspect dependencies for aspect-based sentiment analysis. *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 2, pp. 266-270.

Hochreiter, S.; Schmidhuber, J. (1997): Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735-1780.

Hou, X.; Huang, J.; Wang, G.; Huang, K.; He, X. et al. (2019): Selective attention based graph convolutional networks for aspect-level sentiment classification. arXiv preprint arXiv:1910.10857.

Howard, J.; Ruder, S. (2018): Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146.

Huang, B.; Carley, K. (2018): Parameterized convolutional neural networks for aspect level sentiment classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 1091-1096.

Huang, B.; Carley, K. M. (2019): Syntax-aware aspect level sentiment classification with graph attention networks. arXiv preprint arXiv:1909.02606.

Huang, L.; Ma, D.; Li, S.; Zhang, X.; Wang, H. (2019): Text level graph neural network for text classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 3435-3441.

Jiang, L.; Yu, M.; Zhou, M.; Liu, X.; Zhao, T. (2011): Target-dependent twitter sentiment classification. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, pp. 151-160.

Kipf, T. N.; Welling, M. (2016): Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Kiritchenko, S.; Zhu, X.; Cherry, C.; Mohammad, S. (2014): Nrc-canada-2014: detecting aspects and sentiment in customer reviews. *Proceedings of the 8th International Workshop on Semantic Evaluation*, pp. 437-442.

Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. (2015): Gated graph sequence neural networks. arXiv preprint arXiv:1511.05493.

Ma, D.; Li, S.; Zhang, X.; Wang, H. (2017): Interactive attention networks for aspect level sentiment classification. arXiv preprint arXiv:1709.00893.

Ma, X.; Zeng, J.; Peng, L.; Fortino, G.; Zhang, Y. (2019): Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis. *Future Generation Computer Systems*, vol. 93, pp. 304-311.

Neviarouskaya, A.; Prendinger, H.; Ishizuka, M. (2009): Sentiful: generating a reliable lexicon for sentiment analysis. *3rd International Conference on Affective Computing and Intelligent Interaction and Workshops*, pp. 1-6.

Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E. et al. (2017): Automatic differentiation in pytorch.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C. et al. (2018): Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

Pontiki, M.; Galanis, D.; Pavlopoulos, J.; Papageorgiou, H.; Androutsopoulos, I. et al. (2014): Semeval-2014 task 4: aspect-based sentiment analysis. *Proceedings of the 8th International Workshop on Semantic Evaluation*, pp. 27-35.

Qiu, G.; Bing, L.; Bu, J.; Chen, C. (2009): Expanding domain sentiment lexicon through double propagation. *International Joint Conference on Artificial Intelligence*.

Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. (2018): Improving language understanding by generative pre-training. *https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf.*

Rajpurkar, P.; Jia, R.; Liang, P. (2018): Know what you don't know: unanswerable questions for squad. arXiv preprint arXiv:1806.03822.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; Liang, P. (2016): Squad: 100,000+questions for machine comprehension of text. arXiv preprint arXiv:1606.05250.

Rathnayaka, P.; Abeysinghe, S.; Samarajeewa, C.; Manchanayake, I.; Walpola, M. J. et al. (2019): Gated recurrent neural network approach for multilabel emotion detection in microblogs. arXiv preprint arXiv:1907.07653.

Song, Y.; Wang, J.; Jiang, T.; Liu, Z.; Rao, Y. (2019): Attentional encoder network for targeted sentiment classification. arXiv preprint arXiv:1902.09314.

Sukhbaatar, S.; Weston, J.; Fergus, R. (2015): End-to-end memory networks. *Advances in Neural Information Processing Systems*, pp. 2440-2448.

Sun, C.; Huang, L.; Qiu, X. (2019): Utilizing Bert for aspect-based sentiment analysis via constructing auxiliary sentence. arXiv preprint arXiv:1903.09588.

Taboada, M.; Brooke, J.; Tofiloski, M.; Voll, K.; Stede, M. (2011): Lexicon-based methods for sentiment analysis. *Computational Linguistics*, vol. 37, no. 2, pp. 267-307

Tang, D.; Qin, B.; Feng, X.; Liu, T. (2015): Effective LSTMS for target-dependent sentiment classification. arXiv preprint arXiv:1512.01100.

Tang, D.; Qin, B.; Liu, T. (2016): Aspect level sentiment classification with deep memory network. arXiv preprint arXiv:1605.08900.

Vo, D. T.; Zhang, Y. (2015): Target-dependent twitter sentiment classification with rich automatic features. *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Wagner, J.; Arora, P.; Cortes, S.; Barman, U.; Bogdanova, D. et al. (2014): Dcu: aspect-based polarity classification for SemEval task 4. *Proceedings of the 8th International Workshop on Semantic Evaluation*, pp. 223-229.

Wang, Y.; Huang, M.; Zhu, X.; Zhao, L. (2016): Attention-based lstm for aspect-level sentiment classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 606-615.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M. et al. (2016). Google's neural machine translation system: bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.

Xu, H.; Liu, B.; Shu, L.; Yu, P. S. (2019): Bert post-training for review reading comprehension and aspect-based sentiment analysis. arXiv preprint arXiv:1904.02232.

Xue, W.; Li, T. (2018): Aspect based sentiment analysis with gated convolutional networks. arXiv preprint arXiv:1805.07043.

Yao, L.; Mao, C.; Luo, Y. (2019): Graph convolutional networks for text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 7370-7377.

Zhang, C.; Li, Q.; Song, D. (2019): Aspect-based sentiment classification with aspect specific graph convolutional networks. *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pp. 4560-4570.

Zhang, M.; Zhang, Y.; Vo, D. T. (2016): Gated neural networks for targeted sentiment analysis. *Thirtieth AAAI Conference on Artificial Intelligence*.

Zhang, Y.; Qi, P.; Manning, C. D. (2018): Graph convolution over pruned dependency trees improves relation extraction. arXiv preprint arXiv:1809.10185.

Zhang, Y.; Wang, Q.; Li, Y.; Wu, X. (2018): Sentiment classification based on piecewise pooling convolutional neural network. *Computers, Materials & Continua*, vol. 56, no. 2, pp. 285-297.

Zhaoa, P.; Houb, L.; Wua, O. (2019): Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. arXiv preprint arXiv:1906.04501.

Zhu, H.; Lin, Y.; Liu, Z.; Fu, J.; Chua, T. S. et al. (2019): Graph neural networks with generated parameters for relation extraction. arXiv preprint arXiv:1902.00756.