Graph Attention Networks for Multiple Pairs of Entities and Aspects Sentiment Analysis in Long Texts

Jie LENG

Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, *China;*
University of Chinese Academy of Sciences, Beijing 100049, *China*
E-mail: lengjie@amss.ac.cn

Xijin TANG

Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, *China;*
University of Chinese Academy of Sciences, Beijing 100049, *China*
E-mail: xjtang@iss.ac.cn

Abstract The goal of sentiment analysis is to detect the opinion polarities of people towards specific targets. For fine-grained analysis, aspect-based sentiment analysis (ABSA) is a challenging subtask of sentiment analysis. The goals of most literature are to judge sentiment orientation for a single aspect, but the entities aspects belong to are ignored. Sequence-based methods, such as LSTM, or tagging schemas, such as BIO, always rely on relative distances to target words or accurate positions of targets in sentences. It will require more detailed annotations if the target words do not appear in sentences. In this paper, we discuss a scenario where there are multiple entities and shared aspects in multiple sentences. The task is to predict the sentiment polarities of different pairs, i.e., (entity, aspect) in each sample, and the target entities or aspects are not guaranteed to exist in texts. After converting the long sequences to dependency relation-connected graphs, the dependency distances are embedded automatically to generate contextual representations during iterations. We adopt partly densely connected graph convolutional networks with multi-head attention mechanisms to judge the sentiment polarities for pairs of entities and aspects. The experiments conducted on a Chinese dataset demonstrate the effectiveness of the method. We also explore the influences of different attention mechanisms and the connection manners of sentences on the tasks.

Keywords sentiment analysis; dependency analysis; graph convolution networks; attention mechanism

1 Introduction

With advanced internet technology and cyberspace, people can express their experiences about the goods they bought on e-commerce sites (e.g., Taobao or Amazon), and films they watched on Douban, Maoyan, or IMDb. Studies have shown that consumers trust online reviews or comments before purchasing a product or service[1]. Merchants or manufacturers

also take advantage of feedback to improve services or product qualities. In reality, people usually describe their experiences in more than one sentence. Sometimes they like to compare one brand with another in terms of the products or aspects. In this case, sentiment polarities for pairs of entities and aspects are more complicated to judge.

Given text and targets, the goal of ABSA is to find the sentiment polarity of a specific aspect. For many datasets, samples are all about one entity, and aspects default to exist in corresponding sentences. For example, datasets from SemEval 2014[1], 2015, and 2016 are separated into the restaurant domain or laptop domain. In a typical case about a restaurant: *"The food, though served with bad service, is great"*, the sentiment polarity is negative when the aspect is *"service"*, and it is positive for the *"food"*. Correspondingly, many existing methods are designed to judge the multi-aspect sentiment orientations, but the entities are ignored[2–11].

Some researchers have expanded the sentence-level sentiment classification problem to document-level multi-aspect sentiment classification (DMSC)[12–14]. Wu, et al. even discuss the aspect-based sentiment classification task on the long document (ABSC-LD)[15]. Recently, some researchers introduce a new subtask of ABSA to extract aspect sentiment triplets (ASTE) from sentences[16, 17]. The triplets include an aspect, sentiment polarity, and the opinions explaining the sentiment. However, all the new tasks are still without consideration of entities.

In this paper, we discuss a more flexible and common task: Entities and aspects are combined and analyzed in long texts, and the target words are not required to appear entirely in context. It is similar to a definition of Multi-Entity Aspect-Based Sentiment Analysis (MEMS-ABSA) from Yang, et al.: "Given entities, and the aspects mentioned in the text, the goal is to predict sentiment polarity towards each (entity, aspect) combination"[18]. For a post in their dataset: *"Tried Pampers. No leakage found but his butt went red. Then I changed to Kao. It is a bit expensive but not allergenic"*, the results aim at judging the pair (*Pampers, leakage*) to be positive, (*Pampers, anti-allergy*) to be negative, while (*Kao, anti-allergy*) to be positive, (*Kao, price*) to be negative, due to *"No leakage found"*, *"butt went red"*, *"a bit expensive"* and *"not allergenic"* in text. Mention that the concerned aspect terms in sentences are not the target aspect words of the sample. The key information that their model relies on, for example, opinion spans and the distances between targets and other words, needs precise and careful manual annotations. It makes their model not practical in large-scale applications. The Stanford Core NLP parser[2] provides rich dependency relations that boost their model performances. But the parser is not flexible to combine with Chinese lexicons and deal with Chinese corpora in specific domains.

Previous neural networks models are based on sequences and utilize the distances between words and targets. Recently graph neural networks (GNNs) are used to encode the structural information in graphs for natural language processing[19]. Guo, et al. proposed novel densely connected GCNs to integrate local and non-local features to learn a better structural representation of a graph[20]. Attention mechanisms have become almost a de facto standard in sequence-based methods[21]. Graph attention networks (GATs) incorporate various attention strategies by specifying weights to neighbors for each node[22, 23].

---

[1]https://alt.qcri.org/semeval2014/.
[2]https://stanfordnlp.github.io/CoreNLP/index.html.

In the paper, we handle the distance problem in context automatically through iterations in dependency syntax graphs of sentences without calculations or careful annotations. Apart from dependency information, multi-head attention mechanisms are adapted to modify representations of nodes. In addition, most dependency parsers connect sentences on account of the "root" words in the case of multiple sentences. The first "root" word links to each root word of the following sentences. Here we define the connection manner as "ROOT". For another connection type from Peng, et al., Quirk, et al., and Song, et al., "root" words are connected one by one in order. The new edges are labeled "NEXT"[24–26]. The influences of connections between sentences on graphs are analyzed here.

The contributions of this paper include:

1) We propose a graph-based method to handle sentiment analysis for multiple pairs of entities and aspects in long texts.

2) We compare the effectiveness of three different attention mechanisms and dense connections of GCNs. Different connections between sentences are also discussed.

3) The proposed model ingrates semantic information, syntactic information, node relevance, and task-specific information. It can serve as a baseline for more effective graph-based methods for the task.

2   Related Work

Deep learning methods and neural networks show good performances to deal with the sequences for the ABSA task. Tang, et al. used the target words to segment sequences into parts and discuss the effectiveness of word representations combining target words on the classification results[4]. Yang, et al. applied an attention mechanism to update contextual representations with the given aspects. An attention score indicates the relevance of a word with the given target. The context is represented as the sum of the weighted word vectors generated from the LSTM networks[11]. Based on the hidden representations from LSTM and the distances between words and targets, Yang, et al. leveraged a memory network to update the representations of entities and aspects[18]. They supplied directed dependency relations and shortest dependency paths to improve the performance again[27].

For graph-based methods, it is common practice to leverage dependency syntax analysis to construct text-level graphs[8, 28, 29]. The edges in graphs stand for dependency relations. Sun, et al., Zhang, et al., and Tang, et al. applied GCNs combined with dependency trees to model contextual representations for ABSA and show appealing effectiveness[5, 9, 30]. Wang, et al. adopted two-head attention mechanisms. One is node-level, which takes the dot product of two neighbor nodes as the attention score. The other is edge-level, which assigns weights to the edges based on the em-bedded representations of dependency syntactic relations[5]. Liu, et al. added positive, negative and neutral tags into syntactic graphs as tag nodes, and dynamic heterogeneous graph neural networks are generated. The polarities of aspects are judged according to the connection between word nodes and tag nodes[31]. An, et al. also proposed a heterogeneous aspect graph neural network (HAGNN) to learn the structure and semantic knowledge from intersentence relationships. The heterogeneous graph neural network contains three different kinds of nodes: Word nodes, aspect nodes, and sentence nodes[32]. Liang, et al.

explored a novel solution to construct the graph neural networks via integrating the affective knowledge from SenticNet to enhance the dependency graphs of sentences. Then the dependencies of contextual words and aspect words and the affective information between opinion words and the aspect are totally considered[33].

3   Models

In this section, we introduce some basic GCNs models and existing attention strategies and then present the proposed model for the multi-pair sentiment analysis task. Since a dependency tree is taken as the input, we define two connection manners between sentences to construct a text-level graph at first. For extra propagation of information, the directions of dependency relations are ignored in this paper.

1) Connected as relation "ROOT".

The dependency syntax parser we choose, LTP[3], is widely used for Chinese corpora. It links the first "root" words to others, but other dependency relations of sentences are separated. Figure 1 shows the dependency relation arcs for four successive sentences.
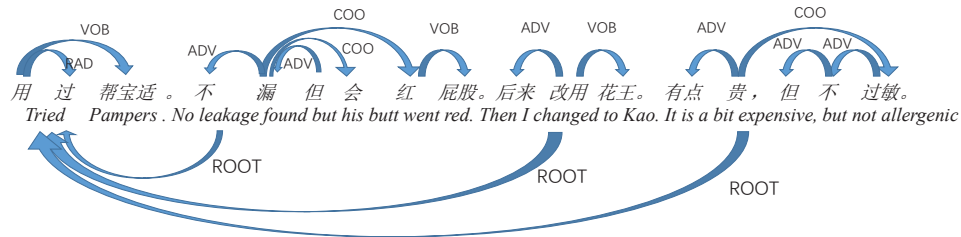

Figure 1   A dependency syntax tree for four sentences. Edges represent conventional intra-sentential dependencies and the connections between the "root" words of adjacent sentences

2) Connected as relation "NEXT".

Motivated by literature about entity relation extraction that applies directed dependency relations and leverages the sentence orders, we reconstruct the dependency graph to make the dependency trees of sentences connected in order through the "root" nodes. Figure 2 illustrates the connection.
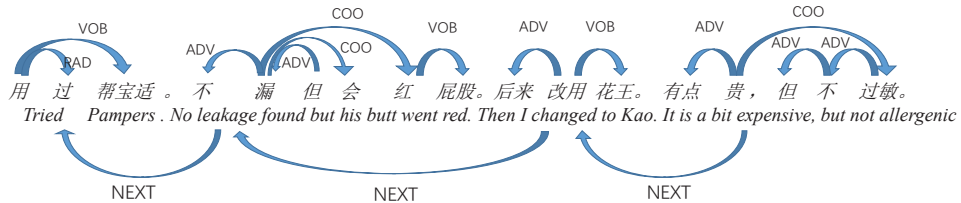

Figure 2   A dependency tree for four sentences. Edges represent conventional intra-sentential dependencies. The "root" words of adjacent sentences are connected in order

---

[3]http://ltp.ai/index.html.

3.1   Graph Convolution Networks (GCNs)

Through successive GCNs operations that allow information to propagate across the network, we want to acquire contextual representations combined with dependency information to classify the sentiment polarity for given entity and aspect pair.

1) **General GCNs**. GCNs are neural networks that operate directly on graph structures[19]. Related nodes in the dependency graph can be represented as an adjacency matrix $A^{n \times n}$. If the edges have directions, specifically, there is a dependency arc from node i to node j, it means $A_{ij} = 1$ and $A_{ji} = 0$. To promote information propagation, we neglect the directionality of edges in the graph. Correspondingly, if there exists a dependency relation between node i and node j_i, then $A_{ij} = 1$ and $A_{ji} = 1$. For node i at the lth layer, the convolution computation takes the neighbors' feature representations h^{l-1} as input and outputs the representation h_i^l. After iterations of N layers, the representation of node i is updated through aggregating information from N-hop neighbors. According to Sun, et al., Zhang, et al.[5, 9], the process can be defined as:

$$h_i^l = \rho \left(\sum_j^n c^i A_{ij} \left(W^l h_j^{l-1} + b^l \right) \right),$$

where ρ is one kind of non-linear activation function (e.g., Relu), c^i is a normalization term chosen as $c^i = 1/d^i$, W^l is the weight matrix and b^l is the bias, d^i denotes the degree of node i in the graph and be calculated as $d^i = \sum_j^n A_{ij}$, h_i^0 is the initial embedding of node i. In practice, a self-loop is added to each node, and the adjacent matrix is modified to be $\tilde{A}$, and $\tilde{A} = A + I_n$. I_n is an identity matrix.

2) **Densely connected GCNs**. Guo, et al. found that although deeper GCNs with more layers will be able to capture richer neighborhood information of a graph, the best performance is achieved with a 2-layer model[20]. To capture non-local information associated with the graphs, they propose the novel densely connected graph convolution networks (DCGCNs). In DCGCNs, the node j in the l-th layer not only receives information from the $(l-1)$-th layer, but also aggregates outputs of all preceding layers. Formally, the representation of node j in the l-th layer is g_j^l, which is the combination of the initial embedding and the representations from layers $1, 2, \cdots, l - 1$:

$$g_j^l = \left[h_j^0, h_j^1, \cdots, h_j^{l-1} \right].$$

Correspondingly, similarly to the expressions of Guo, et al.[20], the convolution operation for node i in each layer is as follows:

$$h_i^l = \rho \left(\sum_j^n c^i A_{ij} \left(W^l g_j^l + b^l \right) \right).$$

3.2   Attention Mechanisms

Attention mechanism has become a significant component in neural networks within diverse application domains. For GCNs, Velickovic, et al. proposed GATs (graph attention networks) that employ self-attention over the node features of neighbors[21]. In our research, we discuss three attention mechanisms for GCNs.

1) **Node similarity**. The similarity-based attention mechanism is general. It assumes that "the attention distribution emphasizes the keys that are relevant for the main task for the

query"[34]. According to Lee, et al.[23], given the features of neighbors and the center word, a normalized attention coefficient is computed by the dot-products between vectors of nodes:

$$\alpha_{ij}^l = \frac{\exp\left(\mathrm{dot}\left(h_i^l, h_j^l\right)\right)}{\sum_j^{N_i} \exp\left(\mathrm{dot}\left(h_i^l, h_j^l\right)\right)}, \quad h_{i,\alpha}^l = \sum_j^{N_i} \alpha_{ij}^l W_\alpha^l h_j^l,$$

where the N_i denote the neighborhood nodes of node i. The node similarity attention mechanism is represented by $\boldsymbol{N}$ in the model names in this paper.

2) **Dependency relation**. As nodes in graphs are linked through dependency relations, Wang, et al. propose to extend the original GATs with a relational head[6]. It is helpful for the effective encoding of syntax information. Here this kind of attention is named $\boldsymbol{R}$ for simplicity. The dependency relations are firstly converted to vector representations, and then the relation-based attention score is computed as:

$$R_{ij}^l = \rho\left(\mathrm{relu}\left(r_{ij} W_{r1} + b_{r1}\right) W_{r2} + b_{r2}\right),$$

$$\beta_{ij}^l = \frac{\exp\left(R_{ij}^l\right)}{\sum_j^{N_i} \exp\left(R_{ij}^l\right)}, \quad h_{i,\beta}^l = \sum_j^{N_i} \beta_{ij}^l \left(W_\beta^l h_j^l + b_\beta^l\right),$$

where the r_{ij} denote the vector of dependency relation between node i and node j.

3) **Entity and aspect relevance**. Inspired by Yang, et al.[18], we proposed a task-specific attention mechanism: The representations of nodes are adapted with enhanced information from entities and aspects. For simplicity, the new attention mechanism is $\boldsymbol{E}$ for short. Since the target entities or aspects are relatively independent of the contexts, tensors of nodes need modifications with the relevance of the concerned entity and aspect in each layer:

$$E_{i,ea}^l = W_1^l \tanh\left(W_2^l \left[h_i^l; h_i^l * \boldsymbol{x}_{\mathrm{entity}}; h_i^l * \boldsymbol{x}_{\mathrm{aspect}}\right]\right) + b_1^l,$$

$$\gamma_{ij}^l = \frac{\exp\left(E_{i,ea}^l\right)}{\sum_j^{N_i} \exp\left(E_{i,ea}^l\right)}, \quad h_{i,\gamma}^l = \sum_j^{N_i} \gamma_{ij}^l \left(W_\gamma^l h_j^l + b_\gamma^l\right),$$

where the $\boldsymbol{x}_{\mathrm{entity}}$ and $\boldsymbol{x}_{\mathrm{aspect}}$ denote the word embeddings of target entity and aspect, respectively.

3.3 Partly Densely Connected GATs

Based on the ideas above, we propose partly densely connected GCNs with multi-head attention mechanisms. An overview of the model architecture is shown in Figure 3. The target entity and aspect take leading roles in generating task-specific and contextual representations for prediction. The word embeddings of the given entity and aspect, $\boldsymbol{x}_{\mathrm{entity}}$ and $\boldsymbol{x}_{\mathrm{aspect}}$ correspondingly, will not be changed. It is different from Yang, et al., which maintains the representation of the context and updates the entity and aspect[18]. After enhanced with the entity and aspect, the initial embeddings of words are fed to a Bi-LSTM to generate semantic representations. For node i, the enhancement operation is element-wise multiplication and concatenation:

$$x_i' = \left[x_i; x_i * \boldsymbol{x}_{\mathrm{entity}}; x_i * \boldsymbol{x}_{\mathrm{aspect}}\right],$$

The outputs of Bi-LSTM, i.e., $h_i^0 = \left[\overrightarrow{x_l'}; \overleftarrow{x_l'}\right]$, are taken as the inputs of GNNs.

Considering that sentences are multiple for each sample in our task, totally dense connections may result in redundancy. For node j, a neighbor of node i, only the initial h_j^0 are added to iterations of the second layer to preserve the original information, i.e., $g_j^2 = \left[h_j^0; h_j^1 \right]$ in reference to the equation in Section 3.1 for densely connected GCNs. The effects of differences are shown in experiments later. In terms of attention mechanisms, task-specific attention and attention based on node similarity are adopted. Then the model is named NED1GATs for short, in which both $\boldsymbol{N}$ and $\boldsymbol{E}$ stand for different attention heads, $\mathbf{D1}$ means partly dense connection, and $\mathbf{GATs}$ is the abbreviation of graph attention neural networks. We concatenate the results of two attention heads to improve the representations of nodes in graphs:

$$h_i^l = h_{i,\alpha}^l \| h_{i,\gamma}^l.$$

The outputs of tensors of nodes from two attention heads are summed up element-wisely to generate a learned representation for the context:

$$h_c = \text{ sum-pooling } (h_i).$$

Then we concatenate the integrated contextual representation with the vectors of the target entity and aspect, i.e., $\boldsymbol{r} = [h_c; \boldsymbol{x}_{\text{entity}}; \boldsymbol{x}_{\text{aspect}}]$. Finally, a linear classifier is applied to conduct sentiment classification.

$$p(P) = \text{softmax} \left(W_p \boldsymbol{r} + b_p \right).$$

The training loss function is defined as the standard cross-entropy loss:

$$L(\theta) = - \sum_{(S,\text{EA}) \in D} \sum_{P \in \text{EA}} \log p(P),$$

where D is the set of the sample (including multiple sentences) and EA pair, EA indicates all entity-aspect pairs (P) in one sample S. θ contains all the trainable parameters. The task about entity or aspect extraction is not involved in this paper.

Another attention combination replacing the node similarity with the attention based on dependency relation, named $\mathbf{RED1GAT}$, encodes the graphs with information of nodes and edges in parallel operation. The hidden representation of node i is $h_i^l = h_{i,\beta}^l \| h_{i,\gamma}^l$, correspondingly. However, it performs weakly worse under given conditions.
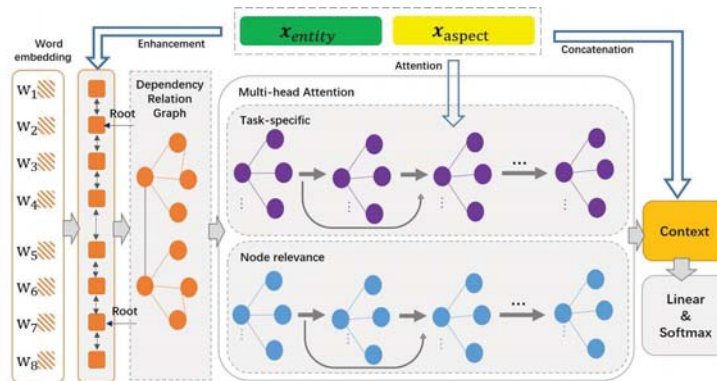

Figure 3   Structure of the proposed partly densely connected graph
attention network with two attention mechanisms

4   Experiments

4.1   Dataset

We evaluate our methods on the Baby Care dataset[18] from *www.babytree.com*, which is one of the largest baby care forums in China. Almost all posts are composed of more than one sentence. Entities and aspects are categorized professionally. They are not the terms mentioned in posts exactly. In the *Pampers-Kao* example above, the terms "*butt went red*" and "*not allergenic*" are categorized as "*anti-allergy*". The numbers of positive, neutral, and negative posts are not equal in the original dataset. We combined the neutral and negative ones to be the "*non-positive*". Then the number of non-positive samples is close to the positive.

We also want to find other available datasets to test the proposed models. But most datasets are single sentences, and each sample only includes one entity or one aspect if it exists. The cases where multiple entities and aspects co-exist in sentences are rare. So we are to apply different conditions to models on the same dataset.

4.2   Comparison Models and Results

Considering that target terms do not appear in sentences for some samples, we only select neural network models that do not rely on the positions of targets in sequences. Models from Tang, et al., combined with popular transformer structures and depending on aspect spans in context[30], are not taken into account temporarily. We mainly compare sequence-based methods based on dependent target word representations and other basic graph-based methods listed below.

Following the previous work[18], we choose a 300-dimensional word embedding of GloVe[35] in the general domain. For graph-based models, LTP[4] is used for dependency parsing. Other experiment settings such as batch size, epoch number, learning rate, dropout, and optimizer remain the same. The metrics are accuracy, precision, recall, and F1.

1) **LSTM**. Sequence-based methods like LSTM are good at capturing semantic information of context. Standard LSTM is to capture all sequence information but not distinguish targets and is often regarded as one baseline[4]. To improve the performance, we append embeddings of entities and aspects to the hidden representation of the last word in context. The concatenation result is to judge the sentiment polarity.

2) **ATAE-LSTM**. The ATET model adopts attention mechanisms to detect important parts in sequence in response to a given aspect. It claims to perform the best in the ABSA task among methods that only update context representations including standard LSTM[4], TD-LSTM[4], TC-LSTM[4], and AE-LSTM[3]. As it only concerns aspects, we add entities in the same way as aspects.

3) **GCNs**. General GCNs are taken as one baseline for graph-based methods here. The calculation is the same as the Section 3.1. The embeddings of entities and aspects are applied to an attention mechanism to generate enhanced representations of contexts.

4) **DCGCNs**. GCNs are densely connected to capture more structural information on large graphs[20]. An attention mechanism about entities and aspects is also applied.

5) **D1GCNs**. To verify the importance of original information, we append the initial vectors

---

[4]http://ltp.ai/index.html.

to the output representations of the first layer as the inputs of the second layer in GCNs.

Table 1 illustrates performance comparisons of sequence-based and basic graph models on the same dataset. It demonstrates that LSTM has a strong ability to represent long sequences. Limited propagations of local nodes in GCNs are far from generating rich representations for contexts. But densely connected GCNs almost overcome the weakness of general GCNs and dramatically improve their performance. We find that an over-densely connected manner is not the best choice. The partly dense connection, which appends the initial inputs to the second layer, is more helpful for the task. Apart from the partly dense connection, the proposed method combines further semantic information, syntactic information, node relevance, and task-specific information. It outperforms the best.

Table 1 Comparisons of sequence-based and basic graph models on the Baby Care dataset

Metrics	LSTM	**ATAE-LSTM**	GCNs	DGCNs	D1GCNs	**NED1GATs**
Accuracy	75.44	**76.18**	44.00	75.17	75.52	**78.27**
Precision	75.44	**76.17**	34.36	75.33	75.52	**78.73**
Recall	75.44	**76.17**	33.90	75.12	75.52	**78.32**
F1	75.44	**76.17**	32.90	75.10	75.52	**78.21**

4.3 Ablation Study

In the proposed model, two head attentions, dense connection, and Bi-LSTM are taken to generate sufficient representations of texts. We conduct an ablation test to investigate the influences of different components. Table 2 illustrates the results. In the case of the first modification removing the relevance of nodes and target entities and aspects, namely, deleting "*E*" from "**NED1GATs**", each metric value decreases. It means the importance of the task information. To analyze the traditional attention mechanism, we remove the attention head based on node similarity and the modified model is "**ED1GATs**" for short. It makes the performance much worse in comparison with the first modification. We also observe that once

Table 2 Results of ablation study of the model **NED1GATs** dataset

No.	Modification	Accuracy	Precision	Recall	**F1**
	Proposed **NED1GATs**	78.27	78.73	78.32	**78.21**
1	remove the attention head of the entity and aspect relevance (**ND1GATs**)	77.57	77.58	77.57	**77.57**
2	remove the attention head of the node similarity (**ED1GATs**)	77.40	77.42	77.41	**77.40**
3	remove the dense connection (**NEGATs**)	76.56	76.71	76.53	**76.51**
4	remove the Bi-LSTM: without initial semantic information	76.40	76.40	76.43	**76.37**

cancel the dense connection (**NEGATs**), or ignore the semantic information from the initial Bi-LSTM (4th moderation), the performance drops considerably. This study validates that each monument of the model is necessary for the task.

4.4 Effects of Attention Mechanisms

To compare the influences of attention mechanisms presented in Section 3.2, i.e. N, R, E, we conduct additional experiments on general GCNs and densely connected GCNs (DCGCNs) introduced in Section 3.1. "-" denotes the absence of attention mechanisms. "$+N$", "$+R$" and "$+E$" represents additions of attention mechanisms based on node similarity, dependency relation and information of target entity and aspect to the graph neural networks respectively.

From the results in Table 3, both the node similarity attention and the attention regarding the given entity and aspect largely improve the performances of general GCNs. It means that the two attention heads inject useful information into the representations of nodes. However, the attention based on dependency relations, which provide edge information, shows limited ability. While focusing on DCGCNs, we find that the effectiveness from attention heads about node similarity and target words relevance is not evident and even weakens the model. The reason can be that the DGCNs have already provided redundant information about nodes, and it is hard to enhance additional messages at the node level. The dependency relation attention still performances poor. Weighted edges may weaken the linkages between nodes to some content and drop out important information for long texts. Since the number of relations in LTP is fewer than 20, the performance of the relational attention mechanism can be improved through more fine-grained dependency syntax analysis and diverse dependency relations. The result about relational attention is similar to the inference in [6]: "Words with too long dependency distances from the target aspect are unlikely to be useful for this task".

Table 3 Comparisons of different attention mechanisms for graph models on the task

Metrics	GCNs				DCGCNs			
	-	$+N$	$+R$	$+E$	-	$+N$	$+R$	$+E$
Accuracy	44	76.78	49.89	76.29	75.17	74.77	51.14	75.8
Precision	34.36	76.99	49.81	76.31	75.33	75.05	51.17	75.8
Recall	33.9	76.81	49.82	76.3	75.12	74.72	51.17	75.79
F1	32.9	76.75	49.41	76.29	75.1	74.67	51.1	75.8

4.5 Effects of Connection Manners

To our knowledge, there is no literature thinking about the orders of sentences. We conduct experiments to observe the influences of the different connection manners on graph-based methods for our task. Since the connections between sentences determine the structures of graphs and the densely connected GCNs capture the whole structure information, we change the dense connections of the proposed model **NED1GATs** and keep the attention mechanisms to design comparison models. "**NEGATs**" mean no dense connection, and "**NEDGATs**" indicate fully dense connections in iterations.

From Table 4, we find that the "ROOT" connection manner, which is common practice

in general parsers, has robust performance. But the "NEXT" one, which connects sentences in order, performs better in the partly dense connections of GCNs. It shows that the design of dense connections of iteration layers should be related to connection manners between the sentences. The observation can be helpful for more complicated graph-based models for long texts.

Table 4 Comparisons of different connection manners between sentences

Metrics	NEGATs		NED1GATs		NEDGATs	
	ROOT	NEXT	ROOT	NEXT	ROOT	NEXT
Accuracy	77.59	76.56	78.11	78.27	77.65	75.64
Precision	77.67	76.71	78.11	78.73	77.68	76.29
Recall	77.57	76.53	78.11	78.32	77.63	75.57
F1	77.57	76.51	78.11	78.21	77.63	75.45

5 Conclusions

In this research, we propose a graph-based method to deal with sentiment analysis for multi-pairs of entities and aspects in multiple sentences. With syntax-aware graphs, syntactic information is embedded. Partly dense connections between iterations are also adopted to capture non-local information. The representations of nodes in graphs are adapted based on Bi-LSTM and further enhanced through two head attentions during iterations. The experiments verify the effectiveness of the model. In addition, we compare different attention mechanisms and find that contextual and task-specific representations are equally crucial for a long text. Two connection manners between sentences are discussed in long texts. The orders of sentences need consideration for partly dense connections of iterations of graphs. The research provides valuable hints for more complicated graph-based methods in dealing with long texts. The proposed model can serve as a baseline for more effective graph-based methods.

In future work, we will try to mitigate the effect of different parsers. More fine-grained dependency parsers can contribute to more accurate connections and relations between words. We will also attempt to add edges indicating adjacent relations in original sequences to promote information propagation in a graph. In addition, we will try to adopt self-supervised methods to build a more effective model and transfer the graph-based method to deal with long texts for more flexible tasks in the follow-up work.

References

[1] Hussein D M E - D M. A survey on sentiment analysis challenges. Journal of King Saud University — Engineering Sciences, 2018, 30(4): 330–338.

[2] Phan M H, Philip O O. Modelling context and syntactical features for aspect-based sentiment analysis.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Lin-Guistics, Online, 2020: 3211–3220.

[3] Wang Y Q, Huang M L, Zhu X Y, et al. Attention-based LSTM for aspect-level sentiment classification. Proceedings of the 2016 Conference on Empirical Meth-ods in Natural Language Processing, Association for Computational Linguistics, Austin, 2016: 606–615.

[4] Tang D Y, Qin B, Feng X C, et al. Effective LSTMs for target-dependent senti-ment classification. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, The COLING 2016 Organizing Committee, Osaka, 2016: 3298–3307.

[5] Sun K, Zhang R C, Mensah S, et al. Aspect-level sentiment analysis via convolution over dependency tree. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, 2019: 5678–5687.

[6] Wang K, Shen W Z, Yang Y Y, et al. Relational graph attention network for aspect-based sentiment analysis. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computa-tional Linguistics, Online, 2020: 3229–323.

[7] Liu J M, Zhang Y. Attention modeling for targeted sentiment. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Valencia, 2017, 2: 572–577.

[8] Huang B X, Carley K M. Syntax-aware aspect level sentiment classification with graph attention net-works. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, 2019: 5468–5476.

[9] Zhang C, Li Q C, Song D W. Aspect-based sentiment classification with aspect-specific graph convolutional networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, 2019: 4567–4577.

[10] Tang D Y, Qin B, Liu T. Aspect level sentiment classification with deep memory network. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, 2016: 214–224.

[11] Yang, M, Tu W T, Wang J X, et al. Attention based LSTM for target dependent sentiment classification. Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. AAAI Press, San Francisco, 2017: 5013–5014.

[12] Ji Y, Liu H, He B, et al. Diversified multiple instance learning for document-level multi-aspect sentiment classification. Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Online, 2020: 7012–7023.

[13] Shi T, Rakesh V, Wang S, et al. Document-level multi-aspect sentiment classifica-tion for online reviews of medical experts. Proceedings of the 28th ACM Interna-tional Conference on Information and Knowledge Management. New York: Association for Computing Machinery, 2019: 2723–2731.

[14] Wang Z, Cao J. Multi-task learning network for document-level and multi-aspect sentiment classification. 2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC), 2020: 171–177.

[15] Wu Z, Gao J, Li Q, et al. Make aspect-based sentiment classification go further: Step into the long-document-level. Applied Intelligence, 2021: 1–20.

[16] Peng H, Xu L, Bing L, et al. Knowing what, how and why: A near complete so-lution for aspect-based sentiment analysis. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(5): 8600–8607.

[17] Chen Z, Huang H, Liu B, et al. Semantic and syntactic enhanced aspect senti-ment triplet extraction. Findings of the Association for Computational Linguis-tics: ACL-IJCNLP, 2021: 1474–1483.

[18] Yang J, Yang R, Wang C, et al. Multi-entity aspect-based sentiment analysis with context, entity and aspect memory. Proceedings of the AAAI Conference on Arti-ficial Intelligence, 2018, 32(1): 6029–6036.

[19] Xu K Y L, Hu W H, Leskovec J, et al. How powerful are graph neural networks? Proceedings of International Conference on Learning Representations, 2019: 1–17.

[20] Guo Z J, Zhang Y, Teng Z Y, et al. Densely connected graph convolutional net-works for graph-to-sequence learning. Transactions of the Association for Computational Linguistics, 2019, 7: 297–312.

[21] Vaswan A, Shazeer N, Parmar N, et al. Attention is all you need. Proceedings of the 31st International

Conference on Neural Information Processing Systems, 2017: 6000–6010.

[22] Velickovic P, Cucurull G, Casanova A, et al. Graph attention networks. arXiv preprint arXiv: 1710.10903, 2017.

[23] Lee J B, Rossi R A, Kim S C, et al. Attention models in graphs: A survey. ACM Transactions on Knowledge Discovery from Data, 2019, 13(6): 62.1–62.25.

[24] Peng N Y, Poon H, Quirk C, et al. Cross-sentence *n*-ary relation extraction with graph LSTMs. Transactions of the Association for Computational Linguistics, 2017, 5: 101–115.

[25] Quirk C, Poon H. Distant supervision for relation extraction beyond the sentence boundary. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, 2017, 1: 1171–1182.

[26] Song L F, Zhang Y, Wang Z G, et al. *N*-ary relation extraction using graph-state LSTM. Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018: 2226–2235.

[27] Yang J, Yang R Q, Lu H Y, et al. Multi-entity aspect-based sentiment analysis with context, entity, aspect memory and dependency information. ACM Transactions on Asian and Low-Resource Language Information Processing, 2019, 18(4): 47:1–47:22.

[28] Guo Z J, Zhang Y, Lu W. Attention guided graph convolutional networks for relation extraction. Proceedings of the 57th Conference of the Association for Computational Linguistics, 2019: 241–251.

[29] Zhang C, Li Q C, Song D W. Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019: 1145–1148.

[30] Tang H, Ji D H, Li C L, et al. Dependency graph enhanced dual-transformer structure for aspect based sentiment classification. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020: 6578–6588.

[31] Liu S, Li W, Wu Y F, et al. Jointly modeling aspect and sentiment with dynamic heter-ogeneous graph neural networks. arXiv: 2004.0642, 2020.

[32] An W, Tian F, Chen P, et al. Aspect-based sentiment analysis with heterogeneous graph neural network. IEEE Transactions on Computational Social Systems. DOI: 10.1109/TCSS.2022.3148866.

[33] Liang B, Su H, Gui L, et al. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. Knowlege-Based Systems, 2022, 235(C): 1–11.

[34] Chaudhari S, Mithal V, Polatkan G, et al. An attentive survey of attention models. ACM Transactions on Intelligent Systems and Technology, 2021, 12(5): 53:1–53:32.

[35] Pennington J, Socher R, Manning C. GloVe: Global vectors for word representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014: 1532–1543.