Dependency Graph Enhanced Dual-transformer Structure for Aspect-based Sentiment Classification

Hao Tang[1] **, Donghong Ji**[1]*****, Chenliang Li**[1] **, Qiji Zhou**[1]
[1]Key Laboratory of Aerospace Information Security and Trusted Computing,
Ministry of Education, School of Cyber Science and Engineering, Wuhan University, China
{tanghaopro,dhji,cllee,qiji.zhou}@whu.edu.cn

Abstract

Aspect-based sentiment classification is a popular task aimed at identifying the corresponding emotion of a specific aspect. One sentence may contain various sentiments for different aspects. Many sophisticated methods such as attention mechanism and Convolutional Neural Networks (CNN) have been widely employed for handling this challenge. Recently, semantic dependency tree implemented by Graph Convolutional Networks (GCN) is introduced to describe the inner connection between aspects and the associated emotion words. But the improvement is limited due to the noise and instability of dependency trees. To this end, we propose a dependency graph enhanced dual-transformer network (named DGEDT) by jointly considering the flat representations learnt from Transformer and graph-based representations learnt from the corresponding dependency graph in an iterative interaction manner. Specifically, a dual-transformer structure is devised in DGEDT to support mutual reinforcement between the flat representation learning and graph-based representation learning. The idea is to allow the dependency graph to guide the representation learning of the transformer encoder and vice versa. The results on five datasets demonstrate that the proposed DGEDT outperforms all state-of-the-art alternatives with a large margin.

1 Introduction

Aspect-based or aspect-level sentiment classification is a popular task with the purpose of identifying the sentiment polarity of the given aspect (Yang et al., 2017; Zhang and Liu, 2017; Zeng et al., 2019). The goal is to predict the sentiment polarity of a given pair (sentence, aspect). Aspects in our study are mostly noun phrases appearing in the input sentence. As shown in Figure 1, where the comment is about the laptop review, the sentiment polarities of two aspects *battery life* and *memory* are positive and negative, respectively. Giving a specific aspect is crucial for sentiment classification owing to the situation that one sentence sometimes contains several aspects, and these aspects may have different sentiment polarities.

Modern neural methods such as Recurrent Neural Networks (RNN), Convolutional Neural Networks (CNN) (Dong et al., 2014; Vo and Zhang, 2015) have already been widely applied to aspect-based sentiment classification. Inspired by the work (Tang et al., 2016a) which demonstrates the importance of modeling the semantic connection between contextual words and aspects, RNN augmented by attention mechanism (Bahdanau et al., 2015; Luong et al., 2015; Xu et al., 2015) is widely utilized in recent methods for exploring the potentially relevant words with respect to the given aspect (Yang et al., 2017; Zhang and Liu, 2017; Zeng et al., 2019; Wang et al., 2016). CNN based attention methods (Xue and Li, 2018; Li et al., 2018) are also proposed to enhance the phrase-level representation and achieved encouraging results.

Although attention-based models have achieved promising performance on several tasks, the limitation is still obvious because attention module may highlight the irrelevant words owing to the syntactical absence. For example, given the sentence *"it has a bad memory but a great battery life."* and aspect *"battery life"*, attention module may still assign a large weight to word *"bad"* rather than *"great"*, which adversely leads to a wrong sentiment polarity prediction.

To take advantages of syntactical information among aspects and contextual words, Zhang et al. (2019) proposed a novel aspect-based GCN method which incorporates dependency tree into the attention models. Actually, using GCN (Kipf and

*Corresponding author.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6578–6588
July 5 - 10, 2020. ©2020 Association for Computational Linguistics

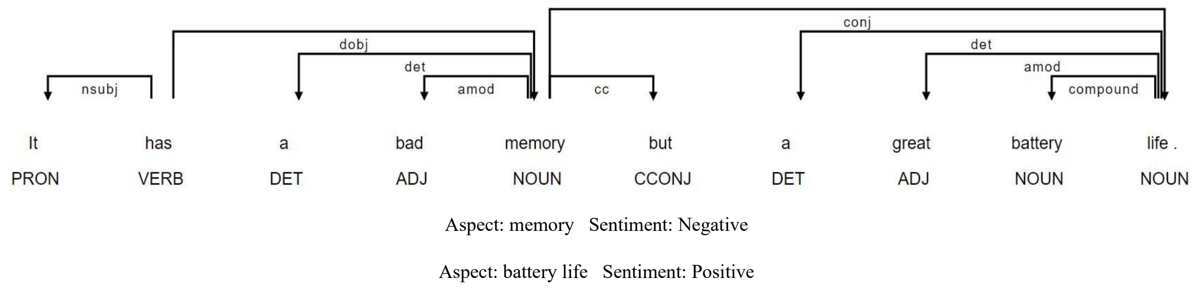


Figure 1: A typical utterance sample of aspect-based sentiment classification task with a proper dependency tree, notice that different aspects may have different sentiment polarities.

Welling, 2017) to encode the information conveyed by a dependency tree has already been investigated in several fields, *e.g.,* modeling document-word relationships (Yao et al., 2019) and tree structures (Marcheggiani and Titov, 2017; Zhang et al., 2018). As shown in Figure 1, an annotated dependency tree of original sentence is provided, and we can observe that word-aspect pairs *(bad, memory)* and *(great, battery life)* are well established.

Direct application of dependency tree has two obvious shortcomings: (1) the noisy information is inevitably introduced through the dependency tree, due to imperfect parsing performance and the casualness of input sentence; (2) GCN would be inherently inferior in modeling long-distance or disconnected words in the dependency tree. It is reported that lower performance is achieved even with the golden dependency tree, by comparing against using only the flat structure (Zhang et al., 2019).

To address these two challenges, we propose a dependency graph enhanced dual-transformer network (named DGEDT) for aspect-based sentiment classification. DGEDT consists of a traditional transformer (Vaswani et al., 2017) and a transformer-like structure implemented via a dependency graph based bidirectional GCN (BiGCN). Specifically, a dual-transformer structure is introduced in DGEDT to fuse the flat representations learnt by the transformer and the graph-based representations learnt based on the dependency graph. These two kinds of representations are jointly refined through a mutual *BiAffine* transformation process, where the dependency graph can guide and promote the flat representation learning. The final flat representations derived by the transformer is then used with an aspect-based attention for sentiment classification. We have conducted extensive experiments over five benchmark datasets. The experimental results demonstrate that the proposed DGEDT achieves a large performance gain over the existing state-of-the-art alternatives.

To the best of our knowledge, the proposed DGEDT is the first work that jointly considers the flat textual knowledge and dependency graph empowered knowledge in a unified framework. Furthermore, unlike other aspect-based GCN models, we aggregate the aspect embeddings from multiple aspect spans which share the same mentioned aspect before feeding these embeddings into submodules. We also introduce an aspect-modified dependency graph in DGEDT.

2 Related Work

Employing modern neural networks for aspect-based sequence-level sentiment classification task, such as CNNs (Kim, 2014; Johnson and Zhang, 2015), RNNs (Castellucci et al., 2014; Tang et al., 2016a), Recurrent Convolutional Neural Networks (RCNNs) (Lai et al., 2015), have already achieved excellent performance in several sentiment analysis tasks. Many attention-based RNN or CNN methods (Yang et al., 2017; Zhang and Liu, 2017; Zeng et al., 2019) are also proposed to handle sequence classification tasks. Tai et al. (2015) proposed a tree-LSTM structure which is enhanced with dependency trees or constituency trees, which outperforms traditional LSTM. Dong et al. (2014) proposed an adaptive recursive neural network using dependency trees. Since being firstly introduced in (Kipf and Welling, 2017), GCN has recently shown a great ability on addressing the graph structure representation in Natural Language Processing (NLP) field. Marcheggiani and Titov (2017) proposed a GCN-based model for semantic role labeling. Vashishth et al. (2018) and Zhang et al.

(2018) used GCN over dependency trees in document dating and relation classification, respectively. Yao et al. (2019) introduced GCN to text classification task with the guidance of document-word and word-word relations. Furthermore, Zhang et al. (2019) introduced aspect-based GCN to cope with aspect-level sentiment classification task using dependency graphs. On the other hand, Chen and Qian (2019) introduced and adapted Capsule Networks along with transfer learning to improve the performance of aspect-level sentiment classification. Gao et al. (2019) introduced BERT into a target-based method, and Sun et al. (2019) constructed BERT-based auxiliary sentences to further improve the performance.

3 Preliminaries

Since Transformer (Vaswani et al., 2017) and GCN are two crucial sub-modules in DGEDT, here we briefly introduce these two networks and illustrate the fact that GCN can be considered as a specialized Transformer.

Assume that there are three input matrices $Q \in R^{n \times d_k}, K \in R^{m \times d_k}, V \in R^{m \times d_v}$, which represent the queries, keys and values respectively. n and m are the length of two inputs.

$$\begin{aligned} Q' &= Attention(Q, K, V) \\ &= softmax(\frac{QK^T}{\sqrt{d_k}})V, \end{aligned} \tag{1}$$

where $Q' \in R^{n \times d_v}$, d_k and d_v are the dimension size of keys and values, respectively. Actually, Transformer adopts multi-head attention mechanism to further enhance the representative ability as follows:

$$h_i = Attention(QW_i^Q, KW_i^K, VW_i^V), \tag{2}$$

$$Q' = Concat([h_1, ...])W^O, \tag{3}$$

where $i \in [1, H]$, H is the head size, $W_i^Q \in R^{d_k \times d_k/H}, W_i^K \in R^{d_k \times d_k/H}, W_i^V \in R^{d_v \times d_v/H}$ and $W^O \in R^{d_v \times d_v}$, and h_i is the i-th head embedding. Then, two normalization layers are employed to extract higher-level features as follows:

$$Q'_1 = Norm(Q' + Q), \tag{4}$$

$$Q'_2 = Norm(Q'_1 + FFN(Q'_1)), \tag{5}$$

where $FFN(x) = Relu(xW_1 + b_1)W_2 + b_2$ is a two-layer multi-layer perceptron (MLP) with the activation function $Relu$, $Norm$ is a normalization layer, Q'_2 is the output vector of this transformer layer. Equations (1)-(5) can be repeated for T times. Note that if $Q = K = V$, this operation can be considered as self alignment.

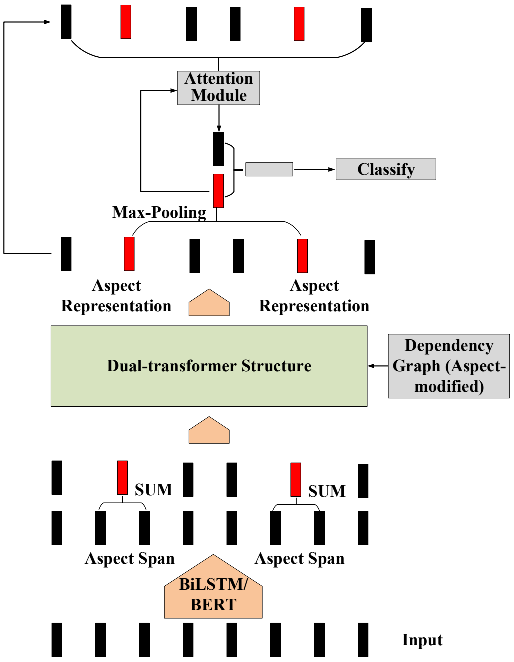


Figure 2: An overall demonstration of our proposed DGEDT. Aspect representation is accumulated from the embeddings in its aspect span, thus the attention module is also aspect-sensitive.

As for GCN, the computation can be conducted as follows when the adjacent matrix of each word in the input is explicitly provided.

$$Q' = Norm(Q + Relu(\frac{1}{|A_{adj}|}A_{adj}QW)), \tag{6}$$

where $A_{adj} \in R^{n \times n}$ is the adjacent matrix formed from the dependency graph, n is the number of words, $Q \in R^{n \times d_k}, W \in R^{d_k \times d_k}$. $\frac{1}{|A_{adj}|}A_{adj}$ is similar to $softmax(\frac{QK^T}{\sqrt{d_k}})$ which is denoted as a generated alignment matrix, except for the main difference that A_{adj} is fixed and discrete. It is obvious that Equation (6) can be decomposed into Equations (1)-(4), and it can be also repeated for T times. In our perspective, GCN is a specialized Transformer with the head size set to one and the generated alignment matrix replaced by a fixed adjacent matrix.

4 DGEDT

The network architecture of our proposed DGEDT is shown in Figure 2. For a given input text, we

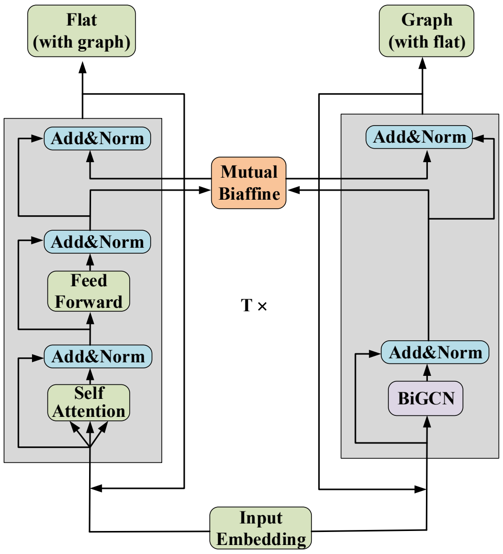


Figure 3: A simplified demonstration of dual-transformer structure, which consists of two sub-modules, one is a standard transformer, another is a transformer-like structure implemented by BiGCN with the supervision of dependency graph.

first utilize BiLSTM or Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2019) as the aspect-based encoder to extract hidden contextual representations. Then these hidden representations are fed into our proposed dual-transformer structure, with the guidance of aspect-modified dependency graph. At last, we aggregate all the aspect representations via max-pooling and apply an attention module to align contextual words and the target aspect. In this way, the model can automatically select relevant aspect-sensitive contextual words with the dependency information for sentiment classification.

4.1 Aspect-based Encoder

We use w_k to represent the k-th word embedding. Bidirectional LSTMs (Schuster and Paliwal, 1997; Hochreiter and Schmidhuber, 1997) (BiLSTM) are applied for the encoder if we do not use BERT.

$$h_1, ... = Encoder([w_1, ...]), \tag{7}$$

where $h_k \in R^h$ is the k-th output of $Encoder$ (BERT or BiLSTM), $k \in [1, N]$ and h is the hidden size, and N is the text length. Note that for a given aspect, there may exist M aspect mentions referring to the same aspect in the text. Also, each aspect mention could contain more than one word. To ease aspect-level representation in the later stage, we choose to collapse each aspect mention as a single word. The summation of the representations of each constituent word within the mention works as its hidden representation. We also develop a span set $span$ with the size N_s. Each span records the start and end position of the given aspect. $span_j$ denotes the j-th aspect span in original text. Note that for non-aspect words, spans involved in the computation are their original positions with the length as one.

$$s_j = SUM([h_{span_j}]), \tag{8}$$

where $j \in [1, N_s]$, $N_s <= N$ denotes the number of words after aspect-based sum operation. s_j is the j-th output of the aspect-based encoder layer. This process can be illuminated by an example transforming *'It has a bad memory but a great battery life'* to *'It has a bad memory but a great [battery life]'*. N is ten and N_s is nine in this case.

4.2 Dual-transformer Structure

After obtaining the contextual hidden representations from the aspect-based encoder, we develop a dual-transformer structure to fuse the flat textual knowledge and dependency knowledge in a mutual reinforcement manner. Specifically, as demonstrated in Figure 3, dual-transformer structure consists of a multi-layer Transformer and a multi-layer BiGCN.

Bidirectional GCN: We design a BiGCN by considering the direction of each edge in the dependency graph. Note that dependency graph is constructed on the word-level. Hence, similar to aspect-level representation performed in Section 4.1, we merge the edges corresponding to the constituent word of the given aspect in the adjacent matrix, resulting in an aspect-level adjacent matrix. Then, we derive the graph-based representations for the input text as follows:

$$Q^t_{out} = Relu(\frac{1}{|A^{out}_{adj}|} A^{out}_{adj} Q_t W_{out}), \tag{9}$$

$$Q^t_{in} = Relu(\frac{1}{|A^{in}_{adj}|} A^{in}_{adj} Q_t W_{in}), \tag{10}$$

$$\begin{aligned} Q_{t+1} = Norm(Q_t + Relu([Q^t_{out} \\ , Q^t_{in}]W_O + b_O)), \end{aligned} \tag{11}$$

$$Q_{t+1} = BiGCN(Q_t, A^{out}_{adj}, A^{in}_{adj}), \tag{12}$$

where A^{out}_{adj} and A^{in}_{adj} are outgoing and incoming aspect-level adjacent matrices gathered from the dependency graph respectively. Here, we concatenate the representations of two directions to produce the final output in each iteration, while other similar methods conduct the merging only in the last iteration. $BiGCN$ represents Equations (9)-(11). We use a simple method to merge the adjacent matrix of the words in the same aspect span as follows:

$$A'_{adj_i} = MIN(\vec{1}, SUM([A_{adj_{span_i}}])), \tag{13}$$

where A_{adj} can be replaced by A^{out}_{adj} and A^{in}_{adj}, and we can thus get $A^{out'}_{adj}$ and ${A^{in}_{adj}}'$. Each span records the start and end position of the given aspect. $span_i$ denotes the i-th span in original text.

BiAffine Module: Assume that there are two inputs $S_1 \in R^{n \times h}$ and $S_2 \in R^{n' \times h}$, we introduce a mutual BiAffine transformation process to interchange their relevant features as follows:

$$A_1 = softmax(S_1 W_1 S_2^T), \tag{14}$$

$$A_2 = softmax(S_2 W_2 S_1^T), \tag{15}$$

$$S'_1 = A_1 S_2, \tag{16}$$

$$S'_2 = A_2 S_1, \tag{17}$$

$$S'_1, S'_2 = Biaffine(S_1, S_2), \tag{18}$$

where $W_1, W_2 \in R^{h \times h}$. Here, S'_1 can be considered as a projection from S_2 to S_1, and S'_2 follows the same principle. $Biaffine$ represents Equations (14)-(17). A_1 and A_2 are temporary alignment matrices projecting from S_2 to S_1 and S_1 to S_2, respectively.

The Whole Procedure: We can then assemble all the sub-modules mentioned above to construct our proposed dual-transformer structure, and the detailed procedures are listed below:

$$S_t^{Tr'} = Transfomer(S_t^{Tr}), \tag{19}$$

$$S_t^{G'} = BiGCN(S_t^G, A^{out'}_{adj}, {A^{in}_{adj}}'), \tag{20}$$

$$S_t^{Tr''}, S_t^{G''} = Biaffine(S_t^{Tr'}, S_t^{G'}), \tag{21}$$

$$S_{t+1}^{Tr} = Norm(S_t^{Tr'} + S_t^{Tr''}), \tag{22}$$

$$S_{t+1}^{G} = Norm(S_t^{G'} + S_t^{G''}), \tag{23}$$

where $S_0^{Tr} = S_0^G = H$, and $H \in R^{N_s \times h}$ denotes the contextual hidden representations $\{s_1, ...\}$ from the aspect-based encoder. $Transfomer$ represents the process denoted by Equations (1)-(5). Equations (19)-(23) can be repeatedly calculated for T times and $t \in [0, T]$. We choose S_T^{Tr} (flat (with graph) in Figure 3) as the last representation, because S_T^G (graph (with flat) in Figure 3) heavily depends on the dependency graph.

4.3 Aspect-based Attention Module

Given M aspect representations can be obtained through the above mentioned procedure, we can derive the final aspect representation by Max-Pooling operation. Here, we utilize an attention mechanism to identify relevant words with respect to the aspect. However, these would be M aspect representations which are all highly relevant to the aggregated aspect representation. To avoid that these aspect mentions from being assigned with too high weight, we utilize a mask mechanism to explicitly set the attention values of aspect mentions to zeros. Let $\mathcal{I}$ be the index set of these M aspect mentions, we form $Mask$ vector as follows:

$$Mask_i = \begin{cases} -inf, & \text{if } i \in \mathcal{I}; \\ 0, & \text{if } other. \end{cases} \tag{24}$$

We then calculate the probability distribution p of the sentiment polarity as follows:

$$h^f = MaxPooling([{S_T^{Tr}}_i | i \in \mathcal{I}]), \tag{25}$$

$$a^f = softmax(h^f W_3 {S_T^{Tr}}^T + Mask), \tag{26}$$

$$h'^f = Relu([h^f, a^f S_T^{Tr}]W' + b'), \tag{27}$$

$$p = softmax(h'^f W_p + b_p), \tag{28}$$

where W_3, W', W_p and b', b_p are learnable weights and biases, respectively.

4.4 Loss Function

The proposed DGEDT is optimized by the standard gradient descent algorithm with the cross-entropy loss and L2-regularization:

$$Loss = -\sum_{(d, y_p) \in D} log(p_{y_p}) + \lambda ||\theta||_2, \tag{29}$$

where D denotes the training dataset, y_p is the ground-truth label and p_{y_p} means the y_p-th element of p. θ represents all trainable parameters, and λ is the coefficient of the regularization term.

5 Experiments

5.1 Datasets

Our experiments are conducted on five datasets, including one (Twitter) which is originally built by Dong et al. (2014), and the other four datasets (Lap14, Rest 14, Rest 15, Rest16) are respectively from SemEval 2014 task 4 (Pontiki et al., 2014), SemEval 2015 task 12 (Pontiki et al., 2015) and SemEval 2016 task 5 (Hercig et al., 2016), consisting

Dataset	Category	Pos	Neu	Neg
Twitter	Train	1561	3127	1560
	Test	173	346	173
Lap14	Train	994	464	870
	Test	341	169	128
Rest14	Train	2164	637	807
	Test	728	196	196
Rest15	Train	912	36	256
	Test	326	34	182
Rest16	Train	1240	69	439
	Test	469	30	117

Table 1: Detailed statistics of five datasets in our experiments.

of data from two categories: laptop and restaurant. The statistics of datasets are demonstrated in Table 1.

5.2 Experiment Setup

We compare the proposed DGEDT[*] with a line of baselines and state-of-the-art alternatives, including LSTM, MemNet (Tang et al., 2016b), AOA (Huang et al., 2018), IAN (Ma et al., 2017), TNet-LF (Li et al., 2018), CAPSNet (Chen and Qian, 2019), Transfer-CAPS (Chen and Qian, 2019), TG-BERT (Gao et al., 2019), AS-CNN (Zhang et al., 2019) and AS-GCN (Zhang et al., 2019). We conduct the experiments with our proposed DGEDT with BiLSTM as the aspect-based encoder, and DGEDT +BERT with BERT as the aspect-based encoder. Several simplified variants of DGEDT are also investigated: DGEDT(Transformer) denotes that we keep standard Transformer and remove the BiGCN part, DGEDT(BiGCN) denotes that we keep BiGCN and remove the Transformer part. The layer number or iteration number (*i.e.,* T) of all available models is set to three for both Transformer and GCN. We use Spacy toolkit[†] to generate dependency trees.

5.3 Parameter Settings

We use BERT-base English version (Devlin et al., 2019), which contains 12 hidden layers and 768 hidden units for each layer. We use Adam (Kingma and Ba, 2014) as the optimizer for BERT and our model with the learning rate initialized by 0.00001 and 0.001 respectively, and decay rate of learning is set as 0.98. Except for the influence of decay rate, the learning rate decreases dynamically according to the current step number. Batch shuffling is applied to the training set. The hidden size of our basic BiLSTM is 256 and the size of all embeddings is set as 100. The vocab size of BERT is 30,522. The batch size of all model is set as 32. As for regularization, dropout function is applied to word embeddings and the dropout rate is set as 0.3. Besides, the coefficient λ for the L2-norm regularization is set as 0.0001. We train our model up to 50 epochs and conduct the same experiment for 10 times with random initialization. Accuracy and Macro-Averaged F1 are adopted as the evaluation metrics. We follow the experimental setup in (Zhang et al., 2019; Chen and Qian, 2019) and report the average maximum value for all metrics on testing set. If the model is not equipped with BERT, then we use word vectors that were pre-trained from Glove (Pennington et al., 2014).

5.4 Overall Results

As shown in Table 2, our model DGEDT outperforms all other alternatives on all five dataset. BERT makes further improvement on the performance especially in Twitter, Rest14 and Rest 15. We can conclude that traditional Transformer DGEDT(Transformer) obtains better performance than DGEDT(BiGCN) in the most datasets. DGEDT employs and combines two sub-modules (traditional Transformer and dependency graph enhanced GCN) and outperforms any single sub-module. Using dependency tree indeed contributes to the performance when acting as a supplement rather than a single decisive module.

5.5 Ablation Study

Note that the performance of individual modules is already reported in Table 2. As shown in Table 3, we investigate and report four typical ablation conditions. '–Mask' denotes that we remove the aspect-based attention mask mechanism, and '–MultiAspect' denotes that we only use the aspect representation of the first aspect mention instead of MaxPooling them. We can see that these two procedures provide slight improvement. '–BiGCN(+GCN)' means that we remove the bidirectional connection and only use original GCN, the results show that bidirectional GCN outperforms original GCN owing to the adequate connection information. '–BiAffine' indicates that we remove the BiAffine process and use all the outputs of dual-transformer structure, we can thus conclude that BiAffine process is critical for our model, and utilizing simple concatenation of the

[*] available at https://github.com/tomsonsgs/DGEDT-senti-master.

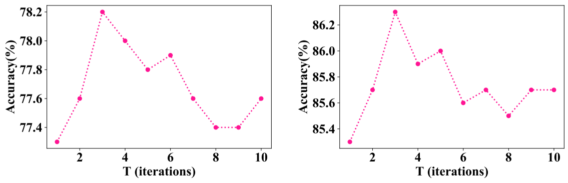
[†] available at https://spacy.io/

Model	Twitter		Lap14		Rest14		Rest15		Rest16	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
LSTM	69.6	67.7	69.3	63.1	78.1	67.5	77.4	55.2	86.8	63.9
MemNet	71.5	69.9	70.6	65.2	79.6	69.6	77.3	58.3	85.4	66.0
AOA	72.3	70.2	72.6	67.5	80.0	70.4	78.2	57.0	87.5	66.2
IAN	72.5	70.8	72.1	67.4	79.3	70.1	78.6	52.7	84.7	55.2
TNet	73.0	71.4	74.6	70.1	80.4	71.0	78.5	59.5	89.1	70.4
AS-CNN	71.1	69.5	72.6	66.7	81.7	73.1	78.5	58.9	87.4	64.6
CAPSNet	–	–	72.7	68.8	78.8	69.7	–	–	–	–
Transfer-CAPS	–	–	73.9	70.2	79.3	70.9	–	–	–	–
AS-GCN	72.2	70.4	75.6	71.1	80.8	72.0	79.9	61.9	89.0	67.5
DGEDT(Transformer)	74.1	72.7	76.0	71.4	82.8	73.9	81.0	64.9	90.0	72.6
DGEDT(BiGCN)	72.8	71.0	76.2	71.8	81.8	72.5	80.4	62.9	89.4	70.4
DGEDT	**74.8**	**73.4**	**76.8**	**72.3**	**83.9**	**75.1**	**82.1**	**65.9**	**90.8**	**73.8**
TG-BERT	76.7	74.3	78.9	74.4	85.1	78.4	–	–	–	–
DGEDT-BERT	**77.9**	**75.4**	**79.8**	**75.6**	**86.3**	**80.0**	**84.0**	**71.0**	**91.9**	**79.0**

Table 2: Overall performance of accuracy and F1 on five datasets, *AS* means aspect-based.

Ablation	Twitter	Lap14	Rest14	Rest15	Rest16
	Acc	Acc	Acc	Acc	Acc
DGEDT	**74.8**	**76.8**	**83.9**	**82.1**	**90.8**
–Mask	74.5	76.7	83.5	82.0	90.5
–MultiAspect	74.5	76.4	83.4	81.8	90.4
–BiGCN (+GCN)	74.3	76.2	83.2	81.4	90.2
–BiAffine	73.0	75.4	82.4	81.0	89.6

Table 3: Overall ablation results of accuracy on five datasets.



(a) Lap14 Dataset.

(b) Rest14 Dataset.

Figure 4: A demonstration of accuracy-T curves on Lap14 and Rest 14 datasets respectively: T is the iteration number.

outputs of Transformer and BiGCN is worse than DGEDT(Transformer).

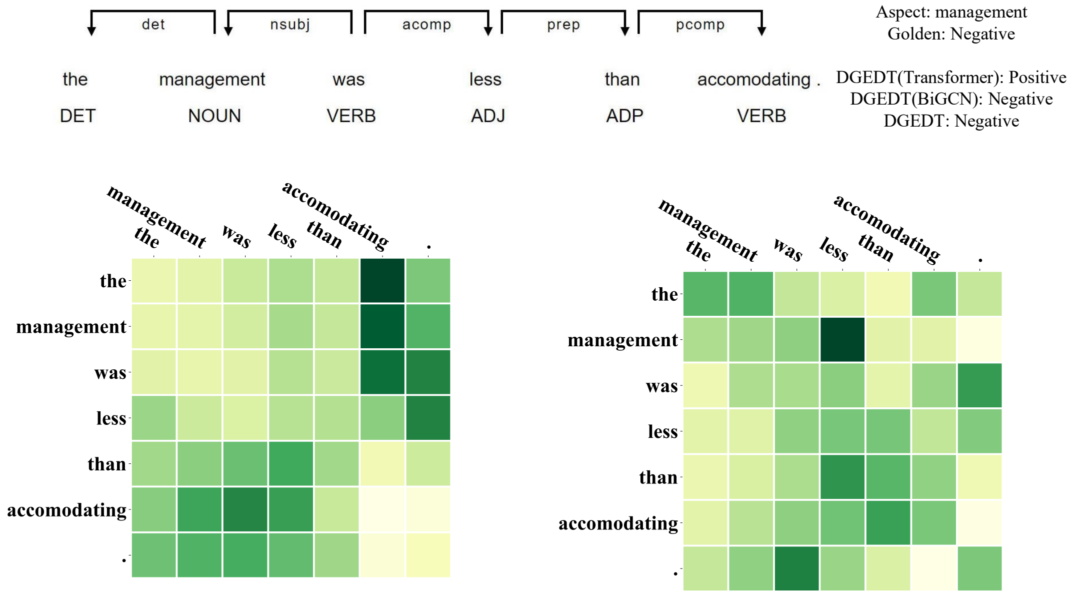
5.6 Impact of Iteration Number

As shown in Figure 4, we find that three is the best iteration number for Lap14 and Rest14. Dependency information will not be fully broadcasted when the iteration number is too small. The model will suffer from over-fitting and redundant information passing, which results in the performance drop when iteration number is too large. So, numerous experiments need to be conducted to figure out a proper iteration number.

5.7 Case Study and Attention Distribution Exploration

As shown in Figure 5, DGEDT and DGEDT(BiGCN) output correct prediction *Negative* while DGEDT(Transformer) fails for the sentence *The management was less than accommodating*. To figure out the essential cause, we demonstrate the attention of self alignment in Figure 5. We can see that for the aspect *management*, DGEDT(Transformer) mainly focuses on *accommodating*, which is a positive word at document level. Thus, DGEDT(Transformer) obtains an incorrect prediction *Positive*. In the dependency tree, *less* which is often regarded as a negative word has a more related connection with aspect *management*, so DGEDT(BiGCN) outputs right sentiment *Negative*. With the assistance of supplementary dependency graph, DGEDT also obtains right prediction *Negative* owing to the high attention value between *management* and *less*.

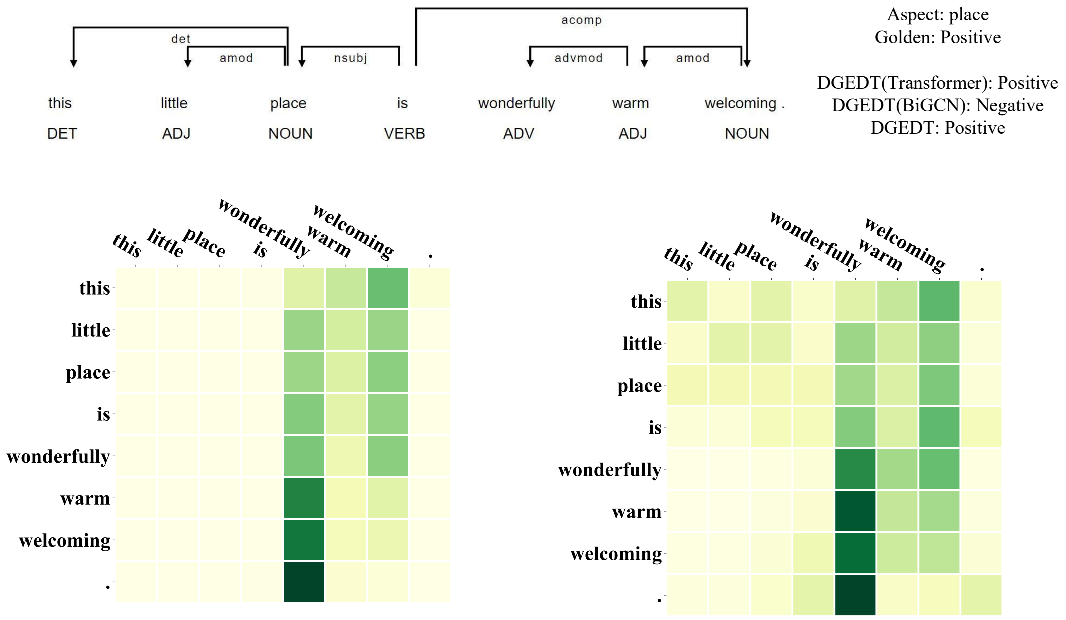
As shown in Figure 6, DGEDT and DGEDT(Transformer) output correct prediction *Positive* while DGEDT(BiGCN) fails for the sentence *This little place is wonderfully warm welcoming*. To figure out the essential cause, we demonstrate the attention of self alignment and dependency tree in Figure 6. We can see that for the aspect *place*, DGEDT(Transformer) mainly focuses on *wonderfully*, which is a positive word at document level. Thus, DGEDT(Transformer) obtains a correct prediction *Positive*. In the dependency tree, *little* which is often regarded as a negative word has a more related connection with aspect *place*, so DGEDT(BiGCN) outputs incorrect sentiment *Negative*. With the disturbance of inappropriate dependency tree, DGEDT still



(a) The attention matrix of self alignment by DGEDT(Transformer).

(b) The attention matrix of self alignment by DGEDT.

Figure 5: **Case Study 1:** A testing example demonstrates that the information of dependency tree contributes to the classification performance, our dual-transformer model generates a proper attention distribution with the assistance of dependency tree. Darker cell color indicates higher attention value, the aspect is *management* and golden sentiment is *Negative*.



(a) The attention matrix of self alignment by DGEDT(Transformer).

(b) The attention matrix of self alignment by DGEDT.

Figure 6: **Case Study 2:** A testing example demonstrates that the information of dependency tree may be harmful for the classification performance, and our dual-transformer model still obtains a proper attention distribution. Darker cell color indicates higher attention value, the aspect is *place* and golden sentiment is *Positive*.

obtains right prediction *Positive* owing to the high attention value between *place* and *wonderfully*.

We can see from two examples above that DGEDT is capable of achieving the proper balance between dependency graph enhanced BiGCN and traditional Transformer according to different situations.

6 Conclusion

Recently neural structures with syntactical information such as semantic dependency tree and constituent tree are widely employed to enhance the word-level representation of traditional neural networks. These structures are often modeled and described by TreeLSTMs or GCNs. To introduce Transformer into our task and diminish the error induced by incorrect dependency trees, we propose a dual-transformer structure which considers the connections in dependency tree as a supplementary GCN module and a Transformer-like structure for self alignment in traditional Transformer. The results on five datasets demonstrate that dependency tree indeed promotes the final performance when utilized as a sub-module for dual-transformer structure.

In future work, we can further improve our method in the following aspects. First, the edge information of the dependency trees needs to be exploited in later work. We plan to employ an edge-aware graph neural network considering the edge labels. Second and last, domain-specific knowledge can be incorporated into our method as an external learning source.

Acknowledgments

We thank the reviewers for their valuable comments. This work is supported through the grants from National Natural Science Foundation of China (NSFC-61772378), the National Key research and Development Program of China (No.2017YFC1200500) and the Major Projects of the National Social Science Foundation of China (No.11&ZD189).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Giuseppe Castellucci, Simone Filice, Danilo Croce, and Roberto Basili. 2014. UNITOR: aspect based sentiment analysis with structured learning. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 761–767. The Association for Computer Linguistics.

Zhuang Chen and Tieyun Qian. 2019. Transfer capsule network for aspect level sentiment classification. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 547–556. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 49–54. The Association for Computer Linguistics.

Zhengjie Gao, Ao Feng, Xinyu Song, and Xi Wu. 2019. Target-dependent sentiment classification with BERT. *IEEE Access*, 7:154290–154299.

Tom'avs Hercig, Tomás Brychcín, Lukás Svoboda, and Michal Konkol. 2016. UWB at semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 342–349. The Association for Computer Linguistics.

S Hochreiter and J Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Binxuan Huang, Yanglan Ou, and Kathleen M. Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. In *Social, Cultural, and Behavioral Modeling - 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, July 10-13, 2018, Proceedings*, volume 10899 of *Lecture Notes in Computer Science*, pages 197–206. Springer.

Rie Johnson and Tong Zhang. 2015. Semi-supervised convolutional neural networks for text categorization via region embedding. In *Advances in Neural Information Processing Systems 28: Annual*

Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, pages 919–927.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1746–1751. ACL.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 2267–2273. AAAI Press.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 946–956. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1412–1421. The Association for Computational Linguistics.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, pages 4068–4074. ijcai.org.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1506–1515. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Processing*, 45(11):2673–2681.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 380–385. Association for Computational Linguistics.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1556–1566. The Association for Computer Linguistics.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3298–3307. ACL.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 214–224. The Association for Computational Linguistics.

Shikhar Vashishth, Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha P. Talukdar.

2018. Dating documents using graph convolution networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1605–1615. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1347–1353. AAAI Press.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 606–615. The Association for Computational Linguistics.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 2048–2057. JMLR.org.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2514–2523. Association for Computational Linguistics.

Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen. 2017. Attention based LSTM for target dependent sentiment classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 5013–5014. AAAI Press.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7370–7377. AAAI Press.

Jiangfeng Zeng, Xiao Ma, and Ke Zhou. 2019. Enhancing attention-based LSTM with position context for aspect-level sentiment classification. *IEEE Access*, 7:20462–20471.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. *CoRR*, abs/1909.03477.

Yue Zhang and Jiangming Liu. 2017. Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 2: Short Papers*, pages 572–577. Association for Computational Linguistics.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2205–2215. Association for Computational Linguistics.