Aspect-based Sentiment Classification with Aspect-specific Graph Convolutional Networks

Chen Zhang
Beijing Institute of Technology
Beijing, China
gene@bit.edu.cn

Qiuchi Li
University of Padua
Padua, Italy
qiuchili@dei.unipd.it

Dawei Song*
Beijing Institute of Technology
Beijing, China
dwsong@bit.edu.cn

Abstract

Due to their inherent capability in semantic alignment of aspects and their context words, attention mechanism and Convolutional Neural Networks (CNNs) are widely applied for aspect-based sentiment classification. However, these models lack a mechanism to account for relevant syntactical constraints and long-range word dependencies, and hence may mistakenly recognize syntactically irrelevant contextual words as clues for judging aspect sentiment. To tackle this problem, we propose to build a Graph Convolutional Network (GCN) over the dependency tree of a sentence to exploit syntactical information and word dependencies. Based on it, a novel aspect-specific sentiment classification framework is raised. Experiments on three benchmarking collections illustrate that our proposed model has comparable effectiveness to a range of state-of-the-art models[1], and further demonstrate that both syntactical information and long-range word dependencies are properly captured by the graph convolution structure.

1 Introduction

Aspect-based (also known as aspect-level) sentiment classification aims at identifying the sentiment polarities of aspects explicitly given in sentences. For example, in a comment about a laptop saying "*From the speed to the multi-touch gestures this operating system beats Windows easily.*", the sentiment polarities for two aspects *operating system* and *Windows* are *positive* and *negative*, respectively. Generally, this task is formulated as predicting the polarity of a provided (sentence, aspect) pair.

Given the inefficiency of manual feature refinement (Jiang et al., 2011), early works of aspect-based sentiment classification are mainly based on neural network methods (Dong et al., 2014; Vo and Zhang, 2015). Ever since Tang et al. (2016a) pointed out the challenge of modelling semantic relatedness between context words and aspects, attention mechanism coupled with Recurrent Neural Networks (RNNs) (Bahdanau et al., 2014; Luong et al., 2015; Xu et al., 2015) starts to play a critical role in more recent models (Wang et al., 2016; Tang et al., 2016b; Yang et al., 2017; Liu and Zhang, 2017; Ma et al., 2017; Huang et al., 2018).

While attention-based models are promising, they are insufficient to capture syntactical dependencies between context words and the aspect within a sentence. Consequently, **the current attention mechanism may lead to a given aspect mistakenly attending to syntactically unrelated context words as descriptors** (Limitation 1). Look at a concrete example "*Its size is ideal and the weight is acceptable.*". Attention-based models often identify *acceptable* as a descriptor of the aspect *size*, which is in fact not the case. In order to address the issue, He et al. (2018) imposed some syntactical constraints on attention weights, but the effect of syntactical structure was not fully exploited.

In addition to the attention-based models, Convolutional Neural Networks (CNNs) (Xue and Li, 2018; Li et al., 2018) have been employed to discover descriptive multi-word phrases for an aspect, based on the finding (Fan et al., 2018) that the sentiment of an aspect is usually determined by key phrases instead of individual words. Nevertheless, the CNN-based models can only perceive multi-word features as consecutive words with the convolution operations over word sequences, but are **inadequate to determine sentiments depicted by multiple words that are not next to each other** (Limitation 2). In the sentence "*The*

*Corresponding author.

[1]Code and preprocessed datasets are available at https://github.com/GeneZC/ASGCN.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 4567–4577,
Hong Kong, China, November 3–7, 2019.

staff should be a bit more friendly" with *staff* as the aspect, a CNN-based model may make an incorrect prediction by detecting *more friendly* as the descriptive phrase, disregarding the impact of *should be* which is two words away but reverses the sentiment.

In this paper, we aim to tackle the two limitations identified above by using Graph Convolutional Networks (GCNs) (Kipf and Welling, 2017). GCN has a multi-layer architecture, with each layer encoding and updating the representation of nodes in the graph using features of immediate neighbors. Through referring to syntactical dependency trees, a GCN is potentially capable of drawing syntactically relevant words to the target aspect, and exploiting long-range multi-word relations and syntactical information with GCN layers. GCNs have been deployed on document-word relationships (Yao et al., 2018) and tree structures (Marcheggiani and Titov, 2017; Zhang et al., 2018), but how they can be effectively used in aspect-based sentiment classification is yet to be explored.

To fill the gap, this paper proposes an Aspect-specific Graph Convolutional Network (ASGCN), which, to the best of our knowledge, is the first GCN-based model for aspect-based sentiment classification. ASGCN starts with a bidirectional Long Short-Term Memory network (LSTM) layer to capture contextual information regarding word orders. In order to obtain aspect-specific features, a multi-layered graph convolution structure is implemented on top of the LSTM output, followed by a masking mechanism that filters out non-aspect words and keeps solely high-level aspect-specific features. The aspect-specific features are fed back to the LSTM output for retrieving informative features with respect to the aspect, which are then used to predict aspect-based sentiment.

Experiments on three benchmarking datasets show that ASGCN effectively addresses both limitations of the current aspect-based sentiment classification approaches, and outperforms a range of state-of-the-art models.

Our contributions are as follows:

- We propose to exploit syntactical dependency structures within a sentence and resolve the long-range multi-word dependency issue for aspect-based sentiment classification.
- We posit that Graph Convolutional Network (GCN) is suitable for our purpose, and propose a novel Aspect-specific GCN model. To our best knowledge, this is the first investigation in this direction.
- Extensive experiment results verify the importance of leveraging syntactical information and long-range word dependencies, and demonstrate the effectiveness of our model in capturing and exploiting them in aspect-based sentiment classification.

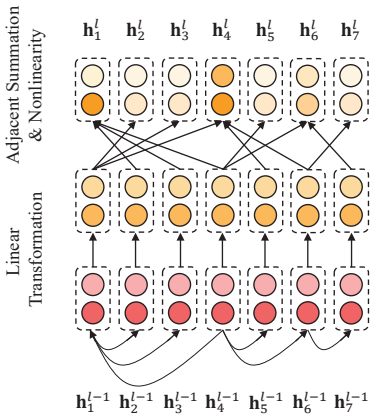

Figure 1: An example of GCN layer.

2 Graph Convolutional Networks

GCNs can be considered as an adaptation of the conventional CNNs for encoding local information of unstructured data. For a given graph with k nodes, an adjacency matrix[2] $\mathbf{A} \in \mathbb{R}^{k \times k}$ is obtained through enumerating the graph. For convenience, we denote the output of the l-th layer for node i as $\mathbf{h}_i^l$, where $\mathbf{h}_i^0$ represents the initial state of node i. For an L-layer GCN, $l \in [1, 2, \cdots, L]$ and $\mathbf{h}_i^L$ is the final state of node i. The graph convolution operated on the node representation can be written as:

$$\mathbf{h}_i^l = \sigma(\sum_{j=1}^{k} \mathbf{A}_{ij} \mathbf{W}^l \mathbf{h}_j^{l-1} + \mathbf{b}^l) \quad (1)$$

where $\mathbf{W}^l$ is a linear transformation weight, $\mathbf{b}^l$ is a bias term, and σ is a nonlinear function, e.g. ReLU. For a better illustration, an example of GCN layer is shown in Figure 1.

As the graph convolution process only encodes information of immediate neighbors, a node in the graph can only be influenced by the neighbouring nodes within L steps in an L-layer GCN. In

[2] $\mathbf{A}_{ij}$ indicates whether the i-th token is adjacent to the j-th token or not.

this way, the graph convolution over the dependency tree of a sentence provides syntactical constraints for an aspect within the sentence to identify descriptive words based on syntactical distances. Moreover, GCN is able to deal with the circumstances where the polarity of an aspect is described by non-consecutive words, as GCN over dependency tree will gather the non-consecutive words into a smaller scope and aggregate their features properly with graph convolution. Therefore, we are inspired to adopt GCN to leverage syntactical information and long-range word dependencies for aspect-based sentiment classification.

3 Aspect-specific Graph Convolutional Network

Figure 2 gives an overview of ASGCN. The components of ASGCN will be introduced separately in the rest of the section.

3.1 Embedding and Bidirectional LSTM

Given a n-word sentence $c = \{w_1^c, w_2^c, \cdots, w_{\tau+1}^c, \cdots, w_{\tau+m}^c, \cdots, w_{n-1}^c, w_n^c\}$ containing a corresponding m-word aspect starting from the $(\tau+1)$-th token, we embed each word token into a low-dimensional real-valued vector space (Bengio et al., 2003) with embedding matrix $\mathbf{E} \in \mathbb{R}^{|V| \times d_e}$, where $|V|$ is the size of vocabulary and d_e denotes the dimensionality of word embeddings. With the word embeddings of the sentence, a bidirectional LSTM is constructed to produce hidden state vectors $\mathbf{H}^c = \{\mathbf{h}_1^c, \mathbf{h}_2^c, \cdots, \mathbf{h}_{\tau+1}^c, \cdots, \mathbf{h}_{\tau+m}^c, \cdots, \mathbf{h}_{n-1}^c, \mathbf{h}_n^c\}$, where $\mathbf{h}_t^c \in \mathbb{R}^{2d_h}$ represents the hidden state vector at time step t from the bidirectional LSTM, and d_h is the dimensionality of a hidden state vector output by an unidirectional LSTM.

3.2 Obtaining Aspect-oriented Features

Different from general sentiment classification, aspect-based sentiment classification targets at judging sentiments from the view of aspects, and thus calls for an aspect-oriented feature extraction strategy. In this study, we obtain aspect-oriented features by applying multi-layer graph convolution over the syntactical dependency tree of a sentence, and imposing an aspect-specific masking layer on its top.

3.2.1 Graph Convolution over Dependency Trees

Aiming to address the limitations of existing approaches (as discussed in previous sections), we leverage a graph convolutional network over dependency trees of sentences. Specifically, after the dependency tree[3] of the given sentence is constructed, we first attain an adjacency matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ according to the words in the sentence. It is important to note that dependency trees are directed graphs. While GCNs generally do not consider directions, they could be adapted to the direction-aware scenario. Accordingly, we propose two variants of ASGCN, i.e. ASGCN-DG on dependency graphs which are un-directional, and ASGCN-DT concerning dependency trees which are directional. Practically, the only difference between ASGCN-DG and ASGCN-DT lies in their adjacency matrices: The adjacency matrix of ASGCN-DT is much more sparse than that of ASGCN-DG. Such setting is in accordance with the phenomenon that parents nodes are broadly influenced by their children nodes. Furthermore, following the idea of self-looping in Kipf and Welling (2017), each word is manually set adjacent to itself, i.e. the diagonal values of $\mathbf{A}$ are all ones.

The ASGCN variants are performed in a multi-layer fashion, on top of the bidirectional LSTM output in Section 3.1, i.e. $\mathbf{H}^0 = \mathbf{H}^c$ to make nodes aware of context (Zhang et al., 2018). Then the representation of each node is updated with graph convolution operation with normalization factor (Kipf and Welling, 2017) as below:

$$\tilde{\mathbf{h}}_i^l = \sum_{j=1}^{n} \mathbf{A}_{ij} \mathbf{W}^l \mathbf{g}_j^{l-1} \tag{2}$$

$$\mathbf{h}_i^l = \mathrm{ReLU}(\tilde{\mathbf{h}}_i^l / (d_i + 1) + \mathbf{b}^l) \tag{3}$$

where $\mathbf{g}_j^{l-1} \in \mathbb{R}^{2d_h}$ is the j-th token's representation evolved from the preceding GCN layer while $\mathbf{h}_i^l \in \mathbb{R}^{2d_h}$ is the product of current GCN layer, and $d_i = \sum_{j=1}^{n} \mathbf{A}_{ij}$ is degree of the i-th token in the tree. The weights $\mathbf{W}^l$ and bias $\mathbf{b}^l$ are trainable parameters.

It is worth noting that we do not have $\mathbf{h}_i^l$ immediately fed into successive GCN layer, but conduct a position-aware transformation in the first place:

$$\mathbf{g}_i^l = \mathcal{F}(\mathbf{h}_i^l) \tag{4}$$

[3]We use spaCy toolkit: https://spacy.io/.

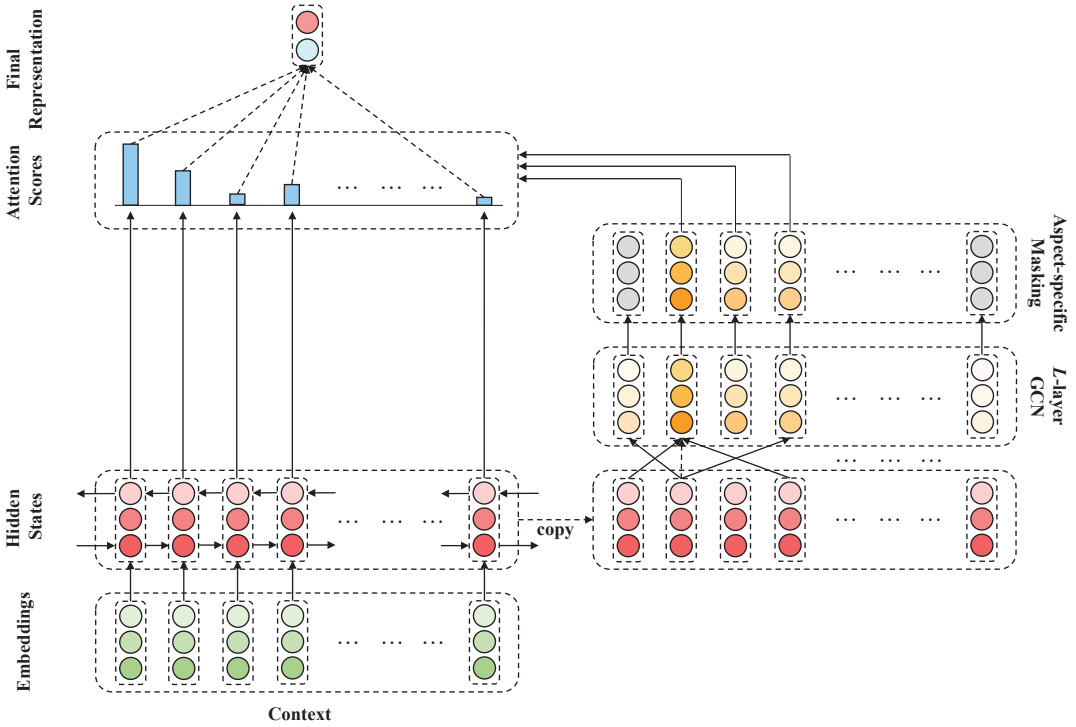

Figure 2: Overview of aspect-specific graph convolutional network.

where $\mathcal{F}(\cdot)$ is a function assigning position weights, widely adopted by previous works (Li et al., 2018; Tang et al., 2016b; Chen et al., 2017), for augmenting the importance of context words close to the aspect. By doing so we aim at reducing the noise and bias that may have naturally arisen from the dependency parsing process. Specifically, the function $\mathcal{F}(\cdot)$ is:

$$q_i = \begin{cases} 1 - \frac{\tau+1-i}{n} & 1 \leq i < \tau+1 \\ 0 & \tau+1 \leq i \leq \tau+m \\ 1 - \frac{i-\tau-m}{n} & \tau+m < i \leq n \end{cases} \quad (5)$$

$$\mathcal{F}(\mathbf{h}_i^l) = q_i \mathbf{h}_i^l \quad (6)$$

where $q_i \in \mathbb{R}$ is the position weight to i-th token. The final outcome of the L-layer GCN is $\mathbf{H}^L = \{\mathbf{h}_1^L, \mathbf{h}_2^L, \cdots, \mathbf{h}_{\tau+1}^L, \cdots, \mathbf{h}_{\tau+m}^L, \cdots, \mathbf{h}_{n-1}^L, \mathbf{h}_n^L\}$, $\mathbf{h}_t^L \in \mathbb{R}^{2d_h}$.

3.2.2 Aspect-specific Masking

In this layer, we mask out hidden state vectors of non-aspect words and keep the aspect word states unchanged:

$$\mathbf{h}_t^L = \mathbf{0} \quad 1 \leq t < \tau+1, \tau+m < t \leq n \quad (7)$$

The outputs of this zero-masking layer are the aspect-oriented features $\mathbf{H}_{\text{mask}}^L = \{\mathbf{0}, \cdots, \mathbf{h}_{\tau+1}^L, \cdots, \mathbf{h}_{\tau+m}^L, \cdots, \mathbf{0}\}$. Through graph convolution, these features $\mathbf{H}_{\text{mask}}^L$ have perceived contexts around the aspect in such a way that considers both syntactical dependencies and long-range multi-word relations.

3.3 Aspect-aware Attention

Based on the aspect-oriented features, a refined representation of the hidden state vectors $\mathbf{H}^c$ is produced via a novel retrieval-based attention mechanism. The idea is to retrieve significant features that are semantically relevant to the aspect words from the hidden state vectors, and accordingly set a retrieval-based attention weight for each context word. In our implementation, the attention weights are computed as below:

$$\beta_t = \sum_{i=1}^{n} \mathbf{h}_t^{c\top} \mathbf{h}_i^L = \sum_{i=\tau+1}^{\tau+m} \mathbf{h}_t^{c\top} \mathbf{h}_i^L \quad (8)$$

$$\alpha_t = \frac{\exp(\beta_t)}{\sum_{i=1}^{n} \exp(\beta_i)} \quad (9)$$

Here, the dot product is used to measure the semantic relatedness between aspect component words and words in the sentence so that aspect-specific masking, i.e. zero masking, could take effect as shown in Equation 8. The final representation for prediction is therefore formulated as:

$$\mathbf{r} = \sum_{t=1}^{n} \alpha_t \mathbf{h}_t^c \tag{10}$$

3.4 Sentiment Classification

Having obtained the representation $\mathbf{r}$, it is then fed into a fully-connected layer, followed by a softmax normalization layer to yield a probability distribution $\mathbf{p} \in \mathbb{R}^{d_p}$ over polarity decision space:

$$\mathbf{p} = \text{softmax}(\mathbf{W}_p \mathbf{r} + \mathbf{b}_p) \tag{11}$$

where d_p is the same as the dimensionality of sentiment labels while $\mathbf{W}_p \in \mathbb{R}^{d_p \times 2d_h}$ and $\mathbf{b}_p \in \mathbb{R}^{d_p}$ are the learned weight and bias, respectively.

3.5 Training

This model is trained by the standard gradient descent algorithm with the cross-entropy loss and L_2-regularization:

$$\text{Loss} = -\sum_{(c,\hat{p}) \in C} \log \mathbf{p}_{\hat{p}} + \lambda \|\Theta\|_2 \tag{12}$$

where C denotes the collection of data sets, $\hat{p}$ is the label and $\mathbf{p}_{\hat{p}}$ means the $\hat{p}$-th element of $\mathbf{p}$, Θ represents all trainable parameters, and λ is the coefficient of L_2-regularization.

4 Experiments

4.1 Datasets and Experimental Settings

Our experiments are conducted on five datasets: one (TWITTER) is originally built by Dong et al. (2014) containing twitter posts, while the other four (LAP14, REST14, REST15, 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 (Pontiki et al., 2016), consisting of data from two categories, i.e. laptop and restaurant. Following previous work (Tang et al., 2016b), we remove samples with conflicting[4] polarities or without explicit aspects in the sentences in REST15 and REST16. The statistics of datasets are reported in Table 1.

For all our experiments, 300-dimensional pretrained GloVe vectors (Pennington et al., 2014) are used to initialize word embeddings. All model weights are initialized with uniform distribution. The dimensionality of hidden state vectors is set to 300. We use Adam as the optimizer with a learning rate of 0.001. The coefficient of L_2-regularization is 10^5 and batch size is 32. Moreover, the number of GCN layers is set to 2, which is the best-performing depth in pilot studies.

The experimental results are obtained by averaging 3 runs with random initialization, where Accuracy and Macro-Averaged F1 are adopted as the evaluation metrics. We also carry out paired t-test on both Accuracy and Macro-Averaged F1 to verify whether the improvements achieved by our models over the baselines are significant.

[4] An opinion target is associated with different sentiment polarities.

Dataset		# 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: Dataset statistics.

4.2 Models for Comparison

In order to comprehensively evaluate the two variants of our model, namely, ASGCN-DG and ASGCN-DT, we compare them with a range of baselines and state-of-the-art models, as listed below:

- SVM (Kiritchenko et al., 2014) is the model which has won SemEval 2014 task 4 with conventional feature extraction methods.
- LSTM (Tang et al., 2016a) uses the last hidden state vector of LSTM to predict sentiment polarity.
- MemNet (Tang et al., 2016b) considers contexts as external memories and benefits from a multi-hop architecture.
- AOA (Huang et al., 2018) borrows the idea of attention-over-attention from the field of machine translation.
- IAN (Ma et al., 2017) interactively models the relationships between aspects and their contexts.

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
SVM	63.40♯	63.30♯	70.49♮	N/A	80.16♮	N/A	N/A	N/A	N/A	N/A
LSTM	69.56	67.70	69.28	63.09	78.13	67.47	77.37	55.17	86.80	63.88
MemNet	71.48	69.90	70.64	65.17	79.61	69.64	77.31	58.28	85.44	65.99
AOA	72.30	70.20	72.62	67.52	79.97	70.42	78.17	57.02	87.50	66.21
IAN	**72.50**	**70.81**	72.05	67.38	79.26	70.09	78.54	52.65	84.74	55.21
TNet-LF	**72.98**	**71.43**	**74.61**	**70.14**	80.42	71.03	78.47	59.47	**89.07**	**70.43**
ASCNN	71.05	69.45	72.62	66.72	**81.73**	**73.10**	78.47	58.90	87.39	64.56
ASGCN-DT	71.53	69.68	74.14†	69.24†	**80.86**‡	**72.19**‡	**79.34**†‡	**60.78**†‡	88.69†	66.64†
ASGCN-DG	72.15†	70.40†	**75.55**†‡	**71.05**†‡	80.77‡	72.02‡	**79.89**†‡	**61.89**†‡	**88.99**†	**67.48**†

Table 2: Model comparison results (%). Average accuracy and macro-F1 score over 3 runs with random initialization. The best two results with each dataset are in bold. The results with ♮ are retrieved from the original papers and the results with ♯ are retrieved from Dong et al. (2014). The marker † refers $p < 0.05$ by comparing with ASCNN in paired t-test and the marker ‡ refers $p < 0.05$ by comparing with TNet-LF in paired t-test.

- TNet-LF (Li et al., 2018) puts forward Context-Preserving Transformation (CPT) to preserve and strengthen the informative part of contexts.

In order to examine to what degrees GCN would outperform CNN, we also involve a model named ASCNN in the experiment, which replaces 2-layer GCN with 2-layer CNN in ASGCN[5].

4.3 Results

As is shown in Table 2, ASGCN-DG consistently outperforms all compared models on LAP14 and REST15 datasets, and achieves comparable results on TWITTER and REST16 datasets compared with baseline TNet-LF and on REST14 compared with ASCNN. The results demonstrate the effectiveness of ASGCN-DG and the insufficiency of directly integrating syntax information into attention mechanism as in He et al. (2018). Meanwhile, ASGCN-DG performs better than ASGCN-DT by a large margin on TWITTER, LAP14, Rest15 and REST16 datasets. And ASGCN-DT's result is lower than TNet-LF's on LAP14. A possible reason is that the information from parents nodes is as important as that from children nodes, so treating dependency trees as directed graphs leads to information loss. Additionally, ASGCN-DG outperforms ASCNN on all datasets except REST14, illustrating ASGCN is better at capturing long-range word dependencies, while to some extent ASCNN shows an impact brought by aspect-specific masking. We suspect REST14 dataset is not so sensitive to syntactic information. Moreover, the sentences from TWITTER dataset are less grammatical, restricting the efficacy. We conjecture this is likely the reason why ASGCN-DG and ASGCN-DT get sub-optimal results on TWITTER dataset.

[5]In order to ensure the length of input and output is consistent, kernel length is set to 3 and padding is 1.

4.4 Ablation Study

To further examine the level of benefit that each component of ASGCN brings to the performance, an ablation study is performed on ASGCN-DG. The results are shown in Table 3. We also present the results of BiLSTM+Attn as a baseline, which uses two LSTMs for the aspect and the context respectively.

First, removal of position weights (i.e. ASGCN-DG w/o pos.) leads to performance drops on LAP14, REST15 and REST16 datasets but performance boosts on TWITTER and REST14 datasets. Recall the main results on REST14 dataset, we conclude that the integration of position weights is not helpful to reduce noise of user generated contents if syntax is not crucial for the data. Moreover, after we get rid of aspect-specific masking (i.e. ASGCN-DG w/o masking), the model could not keep as competitive as TNet-LF. This verifies the significance of aspect-specific masking.

Compared with ASGCN-DG, ASGCN-DG w/o GCN (i.e. preserving position weights and aspect-specific masking, but without using GCN layers) is much less powerful on all five datasets except F1 metric on TWITTER dataset. However, ASGCN-DG w/o GCN is still slightly better than BiL-

Model	TWITTER		LAP14		REST14		REST15		REST16	
	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1
BiLSTM+Attn	71.24	69.55	72.83	67.82	79.85	70.03	78.97	58.18	87.28	68.18
ASGCN-DG	72.15	70.40	75.55	71.05	80.77	72.02	79.89	61.89	88.99	67.48
ASGCN-DG w/o pos.	72.69	70.59	73.93	69.63	81.22	72.94	79.58	61.55	88.04	66.63
ASGCN-DG w/o mask	72.64	70.63	72.05	66.56	79.02	68.29	77.80	57.51	86.36	61.41
ASGCN-DG w/o GCN	71.92	70.63	73.51	68.83	79.40	69.43	79.40	61.18	87.55	66.19

Table 3: Ablation study results (%). Accuracy and macro-F1 scores are the average value over 3 runs with random initialization.

STM+Attn on all datasets except REST14 dataset, due to the strength of the aspect-specific masking mechanism.

Thus it could be concluded that GCN contributes to ASGCN to a considerable extent since GCN captures syntatic word dependencies and long-range word relations at the same time. Nevertheless, the GCN does not work well as expected on the datasets not sensitive to syntax information, as we have seen in TWITTER and REST14 datasets.

4.5 Case Study

To better understand how ASGCN works, we present a case study with several testing examples. Particularly, we visualize the attention scores offered by MemNet, IAN, ASCNN and ASGCN-DG in Table 4, along with their predictions on these examples and the corresponding ground truth labels.

The first sample "*great food but the service was dreadful!*" has two aspects within one sentence, which may hinder attention-based models from aligning the aspects with their relevant descriptive words precisely. The second sample sentence "*The staff should be a bit more friendly.*" uses a subjunctive word "*should*", bringing extra difficulty in detecting implicit semantics. The last example contains negation in the sentence, that can easily lead models to make wrong predictions.

MemNet fails in all three presented samples. While IAN is capable of differing modifiers for distinct aspects, it fails to infer sentiment polarities of sentences with special styles. Armed with position weights, ASCNN correctly predicts the label for the second sample as the phrase *should be* is close to the aspect *staff*, but failed for the third one with a longer-range word dependency. Our ASGCN-DG correctly handles all the three samples, implying that GCN effectively integrates syntactic dependency information into an enriched semantic representation. In particular, ASGCN-DG makes correct predictions on the second and the third sample, both having a seemingly biased focus. This shows ASGCN's capability of capturing long-range multi-word features.

5 Discussion

5.1 Investigation on the Impact of GCN Layers

As ASGCN involves an L-layer GCN, we investigate the effect of the layer number L on the final performance of ASGCN-DG. Basically, we vary the value of L in the set {1,2,3,4,6,8,12} and check the corresponding Accuracy and Macro-Averaged F1 of ASGCN-DG on the LAP14 dataset. The results are illustrated in Figure 3.

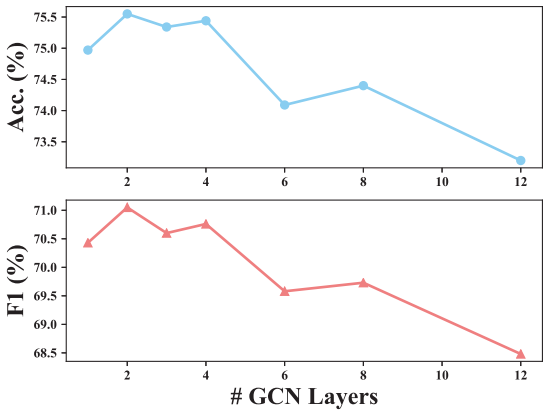

Figure 3: Effect of the number of GCN layers. Accuracy and macro-F1 scores are the average value over 3 runs with random initialization.

On both metrics, ASGCN-DG achieves the best performance when L is 2, which justifies the selection on the number of layers in the experiment sec-

Model	Aspect	Attention visualization	Prediction	Label
MemNet	food	great food but the service was dreadful !	negative✗	positive
	staff	The staff should be a bit more friendly .	positive✗	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	positive✗	negative
IAN	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	positive✗	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	neutral✗	negative
ASCNN	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	negative✓	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	positive✗	negative
ASGCN-DG	food	great food but the service was dreadful !	positive✓	positive
	staff	The staff should be a bit more friendly .	negative✓	negative
	Windows 8	Did not enjoy the new Windows 8 and touchscreen functions .	negative✓	negative

Table 4: Case study. Visualization of attention scores from MemNet, IAN, ASCNN and ASGCN-DG on testing examples, along with their predictions and correspondingly, golden labels. The marker ✓ indicates correct prediction while the marker ✗ indicates incorrect prediction.

tion. Moreover, a dropping trend on both metrics is present as L increases. For large L, especially when L equals to 12, ASGCN-DG basically becomes more difficult to train due to large amount of parameters.

5.2 Investigation on the Effect of Multiple Aspects

In the datasets, there might exist multiple aspect terms in one sentence. Thus, we intend to measure whether such phenomena would affect the effectiveness of ASGCN. We divide the training samples in LAP14 and REST14 datasets into different groups based on the number of aspect terms in the sentences and compute the training accuracy differences between these groups. It is worth noting that the samples with more than 7 aspect terms are removed as outliers because the sizes of these samples are too small for any meaningful comparison.

It can be seen in Figure 4 that when the number of aspects in the sentences is more than 3, the accuracy becomes fluctuated, indicating a low robustness in capturing multiple-aspect correlations and suggesting the need of modelling multi-aspect dependencies in future work.

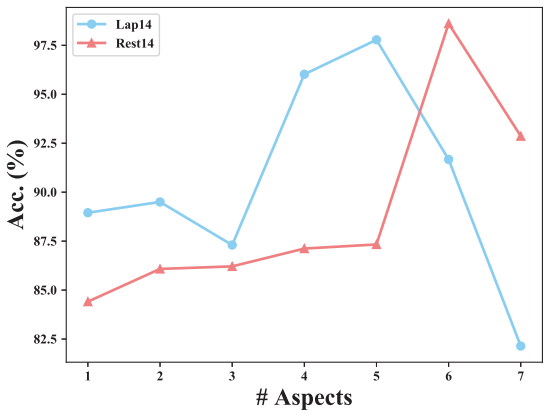

Figure 4: Accuracy versus the number of aspects (# Aspects) in the sentences.

6 Related Work

Constructing neural network models over word sequences, such as CNNs (Kim, 2014; Johnson and Zhang, 2015), RNNs (Tang et al., 2016a) and Recurrent Convolutional Neural Networks (RC-

NNs) (Lai et al., 2015), has achieved promising performances in sentiment analysis. However, the importance but lack of an effective mechanism of leveraging dependency trees for capturing distant relations of words has also been recognized. Tai et al. (2015) showed that LSTM with dependency trees or constituency trees outperformed CNNs. Dong et al. (2014) presented an adaptive recursive neural network using dependency trees, which achieved competitive results compared with strong baselines. More recent research showed that general dependency-based models are difficult to achieve comparable results to the attention-based models, as dependency trees are not capable of catching long-term contextualized semantic information properly. Our work overcomes this limitation by adopting Graph convolutional networks (GCNs) (Kipf and Welling, 2017) .

GCN has recently attracted a growing attention in the area of artificial intelligence and has been applied to Natural Language Processing (NLP). Marcheggiani and Titov (2017) claimed that GCN could be considered as a complement to LSTM, and proposed a GCN-based model for semantic role labeling. Vashishth et al. (2018) and Zhang et al. (2018) used graph convolution over dependency trees in document dating and relation classification, respectively. Yao et al. (2018) introduced GCN to text classification utilizing document-word and word-word relations, and gained improvements over various state-of-the-art methods. Our work investigates the effect of dependency trees in depth via graph convolution, and develops aspect-specific GCN model that integrates with the LSTM architecture and attention mechanism for more effective aspect-based sentiment classification.

7 Conclusions and Future Work

We have re-examined the challenges encountering existing models for aspect-specific sentiment classification, and pointed out the suitability of graph convolutional network (GCN) for tackling these challenges. Accordingly, we have proposed a novel network to adopt GCN for aspect-based sentiment classification. Experimental results have indicated that GCN brings benefit to the overall performance by leveraging both syntactical information and long-range word dependencies.

This study may be further improved in the following aspects. First, the edge information of the syntactical dependency trees, i.e. the label of each edge, is not exploited in this work. We plan to design a specific graph neural network that takes into consideration the edge labels. Second, domain knowledge can be incorporated. Last but not least, the ASGCN model may be extended to simultaneously judge sentiments of multiple aspects by capturing dependencies between the aspects words.

Acknowledgments

This work is supported by The National Key Research and Development Program of China (grant No. 2018YFC0831704), Natural Science Foundation of China (grant No. U1636203, 61772363), Major Project of Zhejiang Lab (grant No. 2019DH0ZX01), and the European Unions Horizon 2020 Research and Innovation Programme under the Marie Skodowska-Curie grant agreement No. 721321.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461.

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 (volume 2: Short papers)*, volume 2, pages 49–54.

Chuang Fan, Qinghong Gao, Jiachen Du, Lin Gui, Ruifeng Xu, and Kam-Fai Wong. 2018. Convolution-based memory network for aspect-based sentiment analysis. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1161–1164. ACM.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018. Effective attention modeling for aspect-level sentiment classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1121–1131.

Binxuan Huang, Yanglan Ou, and Kathleen M Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, pages 197–206. Springer.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 151–160. Association for Computational Linguistics.

Rie Johnson and Tong Zhang. 2015. Semi-supervised convolutional neural networks for text categorization via region embedding. In *Advances in neural information processing systems*, 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)*, pages 1746–1751.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence*.

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 (Volume 1: Long Papers)*, pages 946–956.

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

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*, pages 1412–1421.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4068–4074. AAAI Press.

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*, pages 1506–1515.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 19–30.

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 2015)*, pages 486–495.

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 2014)*, pages 27–35.

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 (Volume 1: Long Papers)*, volume 1, pages 1556–1566.

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

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*, pages 214–224.

Shikhar Vashishth, Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018. Dating documents using graph convolution networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1605–1615.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Yequan Wang, Minlie Huang, Li Zhao, et al. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.

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 (Volume 1: Long Papers)*, pages 2514–2523.

Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen. 2017. Attention based lstm for target dependent sentiment classification. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2018. Graph convolutional networks for text classification. *arXiv preprint arXiv:1809.05679*.

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*, pages 2205–2215.