Learn from Structural Scope: Improving Aspect-Level Sentiment Analysis with Hybrid Graph Convolutional Networks

Lvxiaowei Xu\(^1\), Xiaoxuan Pang\(^2\), Jianwang Wu\(^\dagger\), Ming Cai\(^1\)*, Jiawei Peng\(^1\)

\(^{1}\)Department of Computer Science and Technology, Zhejiang University
\(^{2}\)Alibaba Group-China Digital Commerce-Tao Cai Cai-Tech Dept-Data Intelligence Growth

\{xlxw, wujw, cm, pengjw\}@zju.edu.cn, pangxiaoxuan.pxx@alibaba-inc.com

Abstract

Aspect-level sentiment analysis aims to determine the sentiment polarity towards a specific target in a sentence. The main challenge of this task is to effectively model the relation between targets and sentiments so as to filter out noisy opinion words from irrelevant targets. Most recent efforts capture relations through target-sentiment pairs or opinion spans from a word-level or phrase-level perspective. Based on the observation that targets and sentiments essentially establish relations following the grammatical hierarchy of phrase-clause-sentence structure, it is hopeful to exploit comprehensive syntactic information for better guiding the learning process. Therefore, we introduce the concept of Scope, which outlines a structural text region related to a specific target. To jointly learn structural Scope and predict the sentiment polarity, we propose a hybrid graph convolutional network (HGCN) to synthesize information from constituency tree and dependency tree, exploring the potential of linking two syntax parsing methods to enrich the representation. Experimental results on four public datasets illustrate that our HGCN model outperforms current state-of-the-art baselines.

1 Introduction

Aspect-level sentiment analysis (ALSA) is a fine-grained classification task, aiming at identifying opinion polarities towards specific entities called targets. Figure 1 shows a review commenting on the restaurant with two target terms “menu” and “food”. The sentiment polarities over them are positive and negative respectively. Since multiple targets may appear in one sentence and convey different sentiments, the main challenge of ALSA task is to effectively filter out noisy or misleading opinion words from irrelevant targets and extract distraction-free text from opinion expressions accurately.

Considerable efforts have been devoted to overcoming the difficulties. Their works can be broadly divided into the following three categories. Attention-based models: Attention mechanism has been widely adopted to learn target-specific features [Wang et al., 2016; Ma et al., 2017] so as to help models concentrate on crucial parts of a sentence for sentiment classification. However, the attention scatters across the whole sentence and is prone to attend on noisy opinion words from other targets. Syntax-based models: Syntax-based method aims to model syntactic structures that exist in the sentence. Among them, dependency tree [Zhang et al., 2019; Sun et al., 2019; Wang et al., 2020] has been mostly used to capture the structural dependencies between targets and the sentiment expressions. Nevertheless, multi-hop dependency relations between target and unrelated opinion words that are susceptible to confusing and misleading the model. Span-based models: Span-based models [Wang and Lu, 2018; Hu et al., 2019; Xu et al., 2020] apply a divide-and-conquer strategy to handle the ALSA task. Specifically, the model extracts an opinion span towards each target based on a soft or hard selection approach, and then focuses on the phrase-level features for sentiment classification. Therefore, it is difficult to generate structurally complete spans for complicated sentences due to lack of syntactic information.

To tackle aforementioned problems, we introduce a concept of structural Scope, which is a well-structured and continuous text region expressing target-specific opinion. Scope delineates the exact context of each target so as to alleviate the distraction problem caused by opinion words from other targets. Compared with opinion span, Scope is characterized by a well-defined structure, which highlights the learning of various syntactic features (e.g., phrase, clause and sentence) related to the target to ensure structural and semantic completeness. Figure 1 demonstrates an example containing two Scopes with their corresponding targets in different structures. It combines a phrase-level relation (a noun phrase connecting the target “menu” and opinion word “great”) and a

*Corresponding author
sentence-level relation (the target “food” linking the remote opinion words “not fresh” outside of the local clause through a coreference relation). This demonstrates that the targets and sentiments establish relations following the grammatical hierarchy of phrase-clause-sentence structure, which substantially filters out noisy opinion words from irrelevant targets.

Three observations motivate us to aggregate syntactic information of constituency tree and dependency tree to learn structural Scope. First, constituency parsing decomposes a sentence into phrases and combines them hierarchically into clauses and sentences, which naturally follows the phrase-clause-sentence structure of Scope and can serve as the foundation to encode structural information. Second, dependency parsing is capable of capturing long-range dependencies such as coreference relation, which can help the model obtain the sentence-level connection. Third, constituency parsing displays the entire sentence structure and relations, while dependency parsing explores relations between words. They present two complementary sides of syntactic relations in sentences. Therefore, it would be beneficial to incorporate constituency tree and dependency tree into the model to learn more comprehensive syntactic relations.

On the basis of above analysis, we present a hybrid graph convolutional network (HGCN) to jointly learn the structural Scope and determine the sentiment polarity. For encoding the constituency tree, we propose the constituency GCN module and the computation is performed in three stages. First, the representations of the constituents are composed by the representations of the words and the embeddings of constituent labels. Second, graph convolutional networks (GCNs) [Kipf and Welling, 2016] are exploited to learn constituent representations. Third, a constituent-token attention mechanism is applied to decompose back the information to word representations. Regarding the dependency tree, we utilize a relational graph convolutional network (RGCN) to obtain syntactic representation with labeled edges. Finally, we incorporate such heterogeneous information to determine the Scope region through the conditional random field (CRF) [Lafferty et al., 2001], and to identify the opinion polarities as well.

The main contributions of this work include:

- We introduce the concept of Scope as a syntactically-informed method to capture the relations between targets and sentiments, which could generate distraction-free opinion expressions for sentiment classification.
- We propose a constituency GCN module equipped with constituent-token attention mechanism to encode structural representations from constituency tree.
- We present a new HGCN model to synthesize constituency relations and dependency relations, exploiting the complementary strengths of two syntax parsing methods to learn syntactic relations for Scope.

2 Model

2.1 Definition of Scope

To eliminate noisy opinion words from unrelated targets, we propose a syntactic structure guided modeling concept called Scope. In general, Scope contains a specific target and its dependent sentiment expression within a continuous and minimum text in a sentence. Specifically, we first define a constituent\(^1\) candidate set \(C_s\) to hold a group of constituents (denoted as \(c_i\)), which include a consecutive text span from target to opinion words in their leaf nodes (denoted as \(Node(c_i)\)). Each \(c_i\) can be categorized into three levels according to grammatical definitions, which are phrase level (e.g., VP, NP), clause level (e.g., SBAR) and sentence level (e.g., S). Scope is then defined as seeking the span of text corresponding to the constituent with the fewest leaf nodes in \(C_s\):

\[
Node(C_s) \arg\min([\text{count}(Node(c_i)), c_i \in C_s])
\]

where \(\text{count}(\cdot)\) calculates the number of nodes excluding extraneous components (e.g., adjunct and punctuation).

2.2 Graph Convolutional Network

We leverage graph convolutional networks over constituency tree and dependency tree to encode structural information. GCNs are neural networks that compute the representation of a node conditioned on its neighboring nodes in a given graph. Each node is updated by aggregating the propagated message from the neighboring nodes through multilayer GCNs. Given a graph \(G\) contains sets of nodes \(\mathcal{V}\) and edges \(\mathcal{E}\). Following the idea of [Kipf and Welling, 2016], we allow \(G\) to have self-loops. After that, we can obtain the adjacency matrix \(A \in \mathbb{R}^{n \times n}\) according to the \(G\), where \(n\) denotes the size of \(\mathcal{V}\). The \(A_{ij}\) in the adjacency matrix reflects the connection between the node \(i\) and node \(j\). Specifically, \(A_{ij} = 1\) if node \(i\) is connected to node \(j\) and in the other scenario \(A_{ij} = 0\). Then GCN is able to propagate information over the paths and update node representations. In general, stacking \(l\) layers of GCN allows aggregation of information across the \(l\)-th order neighborhood. In such an operation, the representation of each node is updated with normalization factor as follow:

\[
h^{(l+1)}_i = \sigma \left( \sum_{j=1}^{n} c_i A_{ij} \left(W^{(l)}h^{(l)}_j + b^{(l)} \right) \right)
\]

where \(h^{(l)}_j\) is the hidden representation for node \(j\) that is generated by \(l\)-th layer of the GCN. \(W^{(l)}\) and \(b^{(l)}\) are trainable parameters of weights and bias in \(l\)-th layer, respectively. \(c_i\) is the normalization factor calculated as \(c_i = 1/d_i\) and \(d_i\) denotes the degree of node \(i\) in the graph, which is computed as \(d_i = \sum_{j=1}^{k} A_{ij}\). And \(\sigma\) is a non-linear activation function, (e.g., RELU(\cdot)) [Glorot et al., 2011].

2.3 Hybrid Graph Convolutional Network

Our model aims to extract the Scope relevant to the target term and predict the sentiment polarity jointly. Figure 2 gives an overview of our HGCN model. In the aspect-level sentiment classification task, a sentence with \(n\) words \(\{w_1, \ldots, w_i, w_{i+1}, \ldots, w_j, w_{j+1}, \ldots, w_n\}\) is given, where the target \(\{w_i, \ldots, w_j\}\) is a sub-sequence of the sentence.

\(^1\)Note that we regard non-terminal node above the part-of-speech (POS) level as constituent whereas the terminal node is the word.
Then, BiLSTM [Graves et al., 2013] or BERT [Devlin et al., 2019] is applied as sentence encoder to integrate contextual information into hidden representations of words. For the BiLSTM encoder, we first embed each word into low-dimensional vector space by looking up an embedding matrix \( E_{emb} \in \mathbb{R}^{d_1 \times |V|} \), where \( d_1 \) denotes the dimension of word embedding and \( |V| \) is the size of vocabulary. Then we feed the embedding of words into BiLSTM for obtaining contextual hidden representations \( H = \{ h_1, h_2, \ldots, h_n \} \), where \( h_i \in \mathbb{R}^{d_2} \) is the hidden representations from BiLSTM. As for the BERT, we construct an aspect oriented context “[CLS] sentence [SEP] aspect [SEP]” as input to obtain contextual hidden representations of the sentence. Moreover, there is a gap between word-piece representations of BERT and word-level based syntactic features. Therefore, we compose the representations of the sub-word as word representations with average-pooling to tackle this mismatch. Then the contextual representations are fed into the Constituent GCN (CGCN) and Dependency GCN (DGCN), respectively.

Eventually, we concatenate representations from DGCN and CGCN module to form syntactic representations. After that, a CRF layer is utilized to generate BIO-tags for modeling the Scope with syntactic representations and we aggregate the syntactic representations over the target terms via pooling to determine sentiment polarity jointly. In the following, we will elaborate the details of our proposed HGCN model.

**Constituency GCN (CGCN)**

The constituency tree breaks a sentence into phrases, and then they can be composed hierarchically into clause and sentence level from bottom to top. Intuitively, it is reasonable to utilize constituency tree to encode structural information for modeling Scope. In order to incorporate such structural constituency syntax, we propose CGCN module with multi-stage architecture for obtaining constituency-aware representations. Compared with the SpanGCN model [Marcheggiani and Titov, 2020], we design the constituent-token (C-T) attention to reassign the representation of words as a substitute for BiLSTM, which reduces the computational complexity and attains better performance. As shown in Figure 2, the architecture of CGCN is composed by three sub-modules: constituent encoder, constituent GCN and constituent decoder.

**Constituent Encoder.** The constituency tree is composed of constituents (\( V_c \)) and words (\( V_w \)). In order to obtain representations of constituents, we first embed each constituent label using \( E_{cons} \in \mathbb{R}^{d_c \times |V|} \), where \( d_c \) is the dimension of embedding and \( |V| \) denotes the number of constituents. Then we assign the representation of each constituent \( v \in V \) in this stage relying on the embedding of the constituent label (e.g., NP or VP) and their child nodes set \( \mathcal{S}_v \subseteq (V_c \cup V_w) \). Take Figure 2 as an example. The constituent “S” has three child nodes. Two of them belong to \( V_c \) and the rest one belongs to \( V_w \). Its representation is obtained through averaging the embeddings of its own label, the label embedding of its two child constituents and the representations of word “.”.

**Constituent GCN.** In this stage, it enables the interactive information passing from each constituent and ensures information of the child nodes, which can be integrated into the representation of their immediate parent node and vice versa. GCN is operated on the graph constructed by constituency tree where the nodes represent all constituents in set \( V_c \) and the edges correspond to the connection between constituents and their child nodes in both directions and each constituent is connected to itself on account of the self-loop. After that, structural syntactic representation \( H^c = \{ h_1^c, h_2^c, \ldots, h_m^c \} \) can be obtained from Equation 2 with layer normalization applied, where \( m \) refers to the number of constituents.

**Constituent Decoder.** After the message passing in GCN module, structural syntactic information are injected into the representation of constituents. At this point, we propose the constituent-token (C-T) attention mechanism to assign the information back to each word. In our implementation, the
attention weights indicate the contribution of the word to the representation of corresponding constituent which can be computed in constituent encoder stage. Then the allocation coefficient (i.e. attention scores) can be computed as below:

\[
A^{cons} = \text{softmax} \left( \frac{QW_q \times (KW_k)^T + M}{\sqrt{d}} \right)
\]

(3)

where the matrices \(Q\) and \(K\) are the representation of words and the representation of constituents composed by constituent encoder, respectively. \(W_q\) and \(W_k\) denote the trainable parameters, while \(d\) is the dimensionality of the hidden representations. In addition, \(M\) stands for a mask matrix to ensure that information can only be injected back to the words from their parent constituent. Here, the dot product is adopted to calculate relatedness between words and constituents.

Then we can obtain constituency representations of words based on attention scores, which can be formulated as:

\[
H^{cons} = A^{cons} H^c
\]

(4)

After the above three-stage process, we obtain the constituency representations \(H^{cons} = \{h_1^{cons}, h_2^{cons}, \ldots, h_n^{cons}\}\).

Dependency GCN (DGNC)

The DGNC is used to capture long-range dependencies such as coreference relationships. Intuitively, different dependency relations should have distinctive impacts on adjacent nodes. However, most of the previous work fails to consider the relationship of the edges. Inspired by [Schlichtkrull et al., 2018], we utilize relational graph convolutional network (RGCN) to encode dependency structure with labeled edges.

Firstly, in order to integrate the relationship information of the edges, we embed each kind of edge label using a randomly initialized embedding matrix \(E_{rel} \in \mathbb{R}^{d_e \times |M|}\) where \(d_e\) refers to the dimension of word embedding and \(|M|\) is the number of relation types. Then we can obtain the representation of the relation between node \(i\) and node \(j\) as \(r_{ij}\). We consider the representation \(r_{ij}\) as the gate-value to determine how much of the information will be updated to neighboring nodes. Thus the adjacency matrix \(A_{ij}^d\) takes the form:

\[
A_{ij}^d = \sigma (r_{ij} W_r + b_r)
\]

(5)

where \(W_r\) and \(b_r\) denote trainable parameters of weights and bias. \(\sigma(\cdot)\) refers to the sigmoid function. After that, we regard the hidden representation \(H\) from BiLSTM or BERT encoder as initial node representations in DGNC. After that, we can compute dependency graph representation \(H^{deps} = \{h_1^{deps}, h_2^{deps}, \ldots, h_n^{deps}\}\) based on Equation 2.

Finally, we apply concatenation operation on CGCN and DGNC to obtain the syntactic representations \(H^{syn}\) for Scope modeling. Next an average pooling operation is utilized to generate the sentiment representation \(H^{senti}\) for the ALSA task. Then \(H^{senti}\) is fed into a linear layer and mapped to probabilities of different polarities by a softmax function, i.e.,

\[
p(y) = \text{softmax} \left( W_p H^{senti} + b_p \right)
\]

(6)

where \(W_p\) and \(b_p\) are the learned weights and bias.

### Table 1: Statistics for the four experimental datasets.

| Dataset | Train Positive | Test Positive | Train Neutral | Test Neutral | Train Negative | Test Negative |
|---------|----------------|---------------|---------------|--------------|----------------|---------------|
| Rest14  | 2164 727       | 637 196       | 807 196       |              |                |               |
| Lap14   | 976 337        | 455 167       | 851 128       |              |                |               |
| Rest15  | 912 326        | 36 34          | 256 182       |              |                |               |
| Rest16  | 1657 611       | 101 44        | 748 204       |              |                |               |

2.4 Training and Testing

During the training process, we define a joint loss for selection of Scope and sentiment classification after obtaining the syntactic representation from CGCN and DGNC:

\[
L(\Theta) = L_{polarity} + \gamma L_{scope} + \lambda \|\Theta\|_2
\]

(7)

where \(\gamma\) is the coupling co-efficiency that regulates the two losses. \(\Theta\) represents all trainable parameters in our proposed HGCN model and \(\lambda\) refers to the coefficient of \(L_2\)-regularization. Moreover, \(L_{polarity}\) denotes a standard three-way cross-entropy loss defined for the ALSA task. \(L_{scope}\) is defined as the negative conditional log-likelihood loss of the CRF layer for Scope modeling. We can formulate them as:

\[
L_{polarity} = - \sum_i \hat{y}_i \log p(y_i)
\]

(8)

\[
L_{scope} = - \log P(y \mid u; W, b)
\]

(9)

where \(\hat{y}_i\) denotes the ground truth, \(i\) is the \(i\)-th sentiment polarity and \(W, b\) are trainable parameters of weights and bias.

In addition, as for the Scope selection task, Viterbi algorithm is utilized to predict the most likely label assignment of sequence during the testing process.

3 Experiments

3.1 Experimental Setups.

Datasets. We conduct experiments of aspect-level sentiment analysis task on four benchmark datasets. As the statistics of the datasets shown in Table 1, there are restaurant-domain (Rest14, Rest15, Rest16) datasets and laptop-domain (Lap14) taken from SemEval [Pontiki et al., 2014; Pontiki et al., 2015; Pontiki et al., 2016]. In order to utilize Scope, we develop a semi-automated annotation tool and then ask 4 experienced annotators and 2 verifiers to annotate the datasets with BIO-tags of Scope. It is worth mentioning that our tool can automatically complete the preliminary annotation based on syntax and designed rules. Then annotators make fine refinements on this basis and only 26.4% of the samples require manual adjustments. There are more details in Appendix A.

Baselines. Baselines can be categorized into three groups.

- **1) Attention-based method.** 
  
  ATAE-LSTM [Wang et al., 2016] uses aspect embedding and attention mechanism in ALSA tasks. AOA [Huang et al., 2018] utilizes two LSTMs and an interactive attention mechanism to generate representations for the aspect and sentence. MGAN [Fan et al., 2018] proposes a multigrained attention network to
• Attention-based methods
  ATAEE-LSTM 77.01 70.09 76.60 84.36
  AOA 79.97 72.62 78.17 87.50
  MGAN 79.96 71.81 78.15 85.29
• Syntactic-based methods
  ASGCN 80.80 74.61 79.43 88.02
  CDT 81.74 74.65 77.94 83.88
  BiGDCN 80.83 74.96 80.33 88.47
  InterGCN 81.26 75.06 79.54 88.52
  DGEDT 80.89 73.04 78.32 86.52
  RGAT 81.50 73.72 78.78 88.21
  DualGCN 82.85 76.66 80.16 88.19
• Span-based methods
  SA-LSTM 77.64 70.67 76.63 85.96
  MCRF-SA 80.94 74.11 78.71 87.76
  HGCN (Ours) 84.09 78.64 82.66 89.84
  (82.91) (76.82) (80.81) (88.92)
• Models with BERT
  BERT+Finetune 84.30 77.45 81.85 90.24
  DGEDT+BERT 85.99 78.88 82.95 90.94
  RGAT+BERT 85.91 79.10 83.15 91.39
  DualGCN+BERT 86.29 79.51 83.78 91.43
  HGCN+BERT (Ours) 87.41 81.49 85.61 93.02
  (86.45) (79.59) (83.91) (91.72)

Table 2: Accuracy (%) comparison on baselines over 10 runs with random initialization. The best result for each dataset are in bold. For our models, the upper results represent the best performance and the lower are the average performance among 10 runs.

capture word-level interactions between target term and context. 

2) Syntactic-based method. ASGCN [Zhang et al., 2019] and CDT [Sun et al., 2019] first apply graph convolutional network for encoding aspect-specific word representations. BiGDCN [Zhang and Qian, 2020] adopts hierarchical graph structure to encode dependency and word co-occurrence information. InterGCN [Liang et al., 2020] employs a GCN over a dependency tree to learn aspect representations. RGAT, RGAT+BERT [Wang et al., 2020] utilize a relational graph attention network to encode the pruned dependency trees. DGEDT, DGEDT+BERT [Tang et al., 2020] provide a dependency graph enhanced dual-transformer network to encode heterogeneous information. DualGCN, DualGCN+BERT [Li et al., 2021] incorporate syntactic structure and semantic relevance to generate features.

3) Span-based method. SA-LSTM [Wang and Lu, 2018] presents a segmentation attention model to distill the sentiment semantics for capturing the structural dependencies with external opinion data. MCRF-SA [Xu et al., 2020] proposes a hard selection approach to determine the opinions by self-critical reinforcement learning with annotated opinions.

Training Details. We utilize Stanford Parser [Manning et al., 2014] for constituency and dependency parsing. As for our HGCN model, we utilize 300-d GloVe vectors [Pennington et al., 2014] to initialize word embeddings. Then the embeddings are fed into the BiLSTM model, whose hidden size is set to 100-d. The size of dependency relation embeddings are set to 30, while constituency embeddings are set to 100. We use the Adam optimizer with the learning rate of 0.01. HGCN is trained for 100 epochs with batch size 32. The regularization coefficients $\lambda$ is set to 1e-4 and co-efficiency $\gamma$ is set to 3e-2. As for HGCN+BERT, we use the fine-tuned BERT with officially released pre-trained BERT parameters.

3.2 Results and Analysis

Main Performances. The overall results are shown in Table 2, from which several observations can be derived. First, the HGCN model consistently outperforms all compared baselines on different datasets. Second, compared with attention-based models, our model is significantly better since it focuses directly on the text regions that are semantically associated with the corresponding target. Therefore, it can eliminate noises introduced by the attention mechanism. Third, the performance of HGCN is also better than syntax-based models. It demonstrates that utilizing constituency and dependency information simultaneously enables the model to capture more structural syntactic information as a way to enhance the representations and HGCN can also avoid multi-hop between target and unrelated opinion words in dependency parsing. As for span-based models, our HGCN is able to model more complete text region compared to them and ensures semantic integrity, which allows our model to have a more robust semantic basis for determining the sentiment polarity. In addition, after we integrate our model with BERT, it obtains further improvement and reaches a new state-of-the-art. In a conclusion, these results can illustrate the effectiveness of our HGCN for capturing structural syntactic information in aspect-level sentiment analysis.

Effect of Multiple Aspects. In the datasets, there exists the scenario where a sentence may have multiple targets. Therefore, models may be misled by the opinion words of irrelevant targets and determine the sentiment polarity incorrectly. We conduct experiments on three typical models from different categories mentioned above. As shown in Figure 3 (In Rest15, there are no sentences with 5 targets), syntax-based...
More importantly, HGCN w/o CRF means that we remove syntactic relations can improve the robustness of the model. This indicates that aggregating syntactic information from two complementary sides of syntactic relations can improve the robustness of the model. More importantly, HGCN w/o CRF means that we remove the subtask of Scope modeling so that HGCN cannot utilizing external Scope data. However, our HGCN w/o CRF still outperforms the majority of the baselines shown in Table 2. It proves that our motivation of synthesizing constituency tree and dependency tree can learn more comprehensive syntactic information. In addition, the joint method (i.e., our HGCN) is better than the HGCN w/o CRF and achieves the best performance. This confirms that the joint scheme of Scope modeling and sentiment polarity determination is effective.

### 4 Related Work

Traditional sentiment analysis works are based on sentence-level or document-level while aspect-level sentiment analysis is a more fine-grained sentiment task. Therefore, it has received extensive research attentions in recent years.

In general, opinion words are not particularly far from the target term. Thus among neural networks methods, most recent works utilize various attention-based models [Wang et al., 2016; Huang et al., 2018; Fan et al., 2018] to discriminate sentiment polarity via words nearby the target term implicitly.

Another trend in recent researches are concerned with leveraging dependency parse tree to establish the connection between target terms and opinion words [Tang et al., 2020]. Typically, GCN is utilized to obtain aspect-oriented features from syntactic trees [Zhang et al., 2019; Sun et al., 2019; Zhang and Qian, 2020; Li et al., 2021; Liang et al., 2020; Wang et al., 2020; Tang et al., 2020] since it has the ability on addressing the graph structure representation.

Some other efforts try to extract snippet of opinion words directly from the sentence [Wang and Lu, 2018; Xu et al., 2020]. These works focus on the explicit extraction of target-opinion pairs to determine sentiment polarity.

In order to obtain constituent representations, we thus propose the constituency GCN model and constituent-token attention in this work for incorporating constituency parse tree inspired by the graph-based models used in semantic role labeling (SRL) task [Marcheggiani and Titov, 2020].

### 5 Conclusion

In this paper, we provide a novel perspective that captures relations between targets and sentiments following the grammatical hierarchy of phrase-clause-sentence structure, aiming
at filtering out noisy opinion words from irrelevant targets. Therefore, we introduce the definition of Scope, which is a structural text region related to the target term. To learn comprehensive syntactic relations for Scope, we propose a hybrid graph convolutional network to synthesize information from constituency tree and dependency tree. Experiments on extensive datasets demonstrate that our model outperforms the baselines and achieves the state-of-the-art performance.

References

[Devlin et al., 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In ACL, pages 4171–4186, 2019.

[Fan et al., 2018] Feifan Fan, Yansong Feng, and Dongyan Zhao. Multi-grained attention network for aspect-level sentiment classification. In EMNLP, pages 3433–3442, 2018.

[Glorot et al., 2011] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In JMLR, pages 315–323, 2011.

[Graves et al., 2013] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In 2013 IEEE international conference on acoustics, speech and signal processing, pages 6645–6649, 2013.

[Hu et al., 2019] Mengting Hu, Shiwan Zhao, Honglei Guo, Renhong Cheng, and Zhong Su. Learning to detect opinion snippet for aspect-based sentiment analysis. In CoNLL, pages 970–979, 2019.

[Huang et al., 2018] Binxuan Huang, Yanglan Ou, Kathleen M Carley. 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, 2018.

[Kipf and Welling, 2016] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. ICLR, 2016.

[Lafferty et al., 2001] John D Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In ICML, pages 282–289, 2001.

[Li et al., 2021] Ruifan Li, hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. Dual graph convolutional networks for aspect-based sentiment analysis. In ACL, pages 6319–6329, 2021.

[Liang et al., 2020] Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In COLING, pages 150–161, 2020.

[Ma et al., 2017] Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. Interactive attention networks for aspect-level sentiment classification. In IJCAI, 2017.

[Ma et al., 2014] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In ACL, pages 55–60, 2014.

[Marcheggiani and Titov, 2020] Diego Marcheggiani and Ivan Titov. Graph convolutions over constituent trees for syntax-aware semantic role labeling. In EMNLP, pages 3915–3928, 2020.

[Pennington et al., 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In EMNLP, pages 1532–1543, 2014.

[Pontiki et al., 2014] Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androustopoulos, and Suresh Manandhar. Semeval-2014 task 4: Aspect based sentiment analysis. In SemEval, pages 27–35, 2014.

[Pontiki et al., 2015] Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androustopoulos. Semeval-2015 task 12: Aspect based sentiment analysis. In SemEval, pages 486–495, 2015.

[Pontiki et al., 2016] Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androustopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. Semeval-2016 task 5: Aspect based sentiment analysis. In International workshop on semantic evaluation, pages 19–30, 2016.

[Schlichtkrull et al., 2018] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In European semantic web conference, pages 593–607. Springer, 2018.

[Sun et al., 2019] Kai Sun, Richong Zhang, Samuel Men-Sah, Yongyi Mao, and Xudong Liu. Aspect-level sentiment analysis via convolution over dependency tree. In EMNLP-IJCNLP, pages 5679–5688, 2019.

[Tang et al., 2020] Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In ACL, pages 6578–6588, 2020.

[Wang and Lu, 2018] Bailin Wang and Wei Lu. Learning latent opinions for aspect-level sentiment classification. In AAAI, 2018.

[Wang et al., 2016] Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. Attention-based lstm for aspect-level sentiment classification. In ACL, pages 606–615, 2016.

[Wang et al., 2020] Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. Relational graph attention network for aspect-based sentiment analysis. In ACL, pages 3229–3238, 2020.

[Xu et al., 2020] Lu Xu, Lidong Bing, Wei Lu, and Fei Huang. Aspect sentiment classification with aspect-specific opinion spans. In EMNLP, pages 3561–3567, 2020.
A Datasets and Annotation Procedures

Consider the sample sentence “Great food but the service was dreadful!” with “food” and “service” as targets. The Scope of target “food” can be “Great food” under definition of the Scope. As for the BIO-tags in this case, “Great” is marked as “B” and “food” is marked as “I” while other tokens in the sentence are marked as “O”. To simplify the annotation procedures, we develop a semi-automated annotation tool as shown in Figure 5. Our tool can automatically complete the preliminary annotation based on syntax and designed rules. The auto-labeled Scope is set as blue blocks in Figure 5. Therefore, the annotators do not need to annotate all samples and only make fine refinements on the inappropriate pre-annotation results. According to our statistics, only 26.4% of the samples require manual adjustments. This demonstrates that Scope annotation procedures are easily accessible and we will open the annotation tool.

B Case Study

In this section, we investigate the behaviour of HGCN, CDT and RGAT on case examples. As we can see from the first example in Table 4, the CDT model is mistakenly identified since it is affected by an incorrect connection in the dependency tree where the edges connect “service” and “great” with 2-hops. Due to the gate mechanism, RGAT can filter out the noisy opinion word “great” by utilizing the edge labels, thus responding correctly. Our HGCN can exploit region information provided by the structural Scope to ensure the correctness of the determination. As for the second example, our model reasonably generates the Scope related to the target term “location” and identifies the sentiment polarity correctly. However, due to the edge between target term and “treasure” in dependency tree, RGAT and CDT are both misled. The third example shows that RGAT and CDT are all confused by another target “blueberry” in the sentence with its opinion word “wonderful”. By comparison, our model is able to determine sentiment polarity correctly since it delineates the exact context of each target so as to filter out noisy opinion words from irrelevant targets. These examples all illustrate the effectiveness of our Scope.

C Error Analysis

To analyze the limitation of our model, we trace back the error cases in the test sets, and identify three categories of reasons: short-sighted error, interruption error and representation error. The short-sighted error comes from the reason that the Scope fails to attend implicit transitive descriptions related to the target term. For example, the sentence

“For someone who used to hate Indian food, Baluchi’s has changed my mind.” with its target term “Indian food”, our Scope ignores the latter part of the sentence which contains the semantic transitions. Thus our model considers this to be an expression of negative sentiment polarity resulting in misidentification. As for the interruption error, some grammatical connection of the target term in the sentence (e.g., parenthetical text) can be interruptive, which may interfere with the constituency parsing and the modeling of Scope. Consequently, the Scope selection will suffer from missing essential information or ill-structured problems, like separate segments. Most of the other errors can be summarized as representation error. Though the model can correctly delineate the Scope, it is still unable to determine the sentiment polarity accurately, especially for the expression of negation and modality, which remains challenging for neural networks.

Table 4: The words highlighted in red and blue denote the given targets and their Scope generated by HGCN. The notations P, N and O represent positive, negative and neutral sentiment, respectively.

| Sentence                                           | HGCN | CDT | RGAT |
|----------------------------------------------------|------|-----|------|
| 1. Great food but the service was dreadful!        | P✓N✓| P✓P✗| P✓N✓ |
| 2. Unfortunately, unless you live in the neighborhood, it’s not in a convenient location but is more like a hidden treasure. | N✓   | P✗  | P✗  |
| 3. My friend had a burger and i had these wonderful blueberry pancakes. | O✓P✗| P✗P✓| P✗P✓|