Aspect-Category based Sentiment Analysis with Hierarchical Graph Convolutional Network

Hongjie Cai[1], Yaofeng Tu[2], Xiangsheng Zhou[2], Jianfei Yu[1], Rui Xia[1]*

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology
[2]ZTE Corporation, Shenzhen, China
{hjcai, jfyu, rxia}@njust.edu.cn
{zhou.xiangsheng, tu.yaofeng}@zte.com.cn

Abstract

Most of the aspect based sentiment analysis research aims at identifying the sentiment polarities toward some explicit aspect terms while ignores implicit aspects in text. To capture both explicit and implicit aspects, we focus on aspect-category based sentiment analysis, which involves joint aspect category detection and category-oriented sentiment classification. However, currently only a few simple studies have focused on this problem. The shortcomings in the way they defined the task make their approaches difficult to effectively learn the inner-relations between categories and the inter-relations between categories and sentiments. In this work, we re-formalize the task as a category-sentiment hierarchy prediction problem, which contains a hierarchy output structure to first identify multiple aspect categories in a review sentence, and jointly predict the sentiment for each of the identified categories. Specifically, we propose a Hierarchical Graph Convolutional Network (Hier-GCN), where a lower-level GCN is to model the inner-relations among multiple categories, and the higher-level GCN is to capture the inter-relations between aspect categories and sentiments. Extensive evaluations demonstrate that our hierarchy output structure is superior over existing ones, and the Hier-GCN model consistently achieves the best results on four benchmarks.

1 Introduction

As an important fine-grained subtask in the field of sentiment analysis, Aspect-Based Sentiment Classification (ABSC) aims to detect the sentiment polarities of aspect terms mentioned in a review sentence. For example, in Table 1, given the aspect term *food*, it is expected to identify its corresponding sentiment polarity as positive. The main limitation of ABSC lies in that aspect terms need to be annotated before aspect sentiment classification, which is not applicable to real applications. To address this problem, many studies have been proposed to explore Aspect Term-based Sentiment Analysis (ATSA), which performs aspect term extraction and aspect sentiment classification jointly (Mitchell et al., 2013; Zhang et al., 2015; Li et al., 2019; Luo et al., 2019; Hu et al., 2019).

However, ATSA still suffers from a major obstacle that it only considers explicit aspects but completely ignore implicit aspects in text. Take the review in Table 1 as an example. Although the second clause does not mention any aspect term, it clearly expresses user's negative sentiment towards the service. More importantly, we observe that existing benchmark datasets contain a large amount of such reviews. For instance, Table 2 shows the proportion of reviews with implicit aspects in the Restaurant dataset from SemEval 2015 and 2016, and it is clear that nearly 25% of the examples contain implicit aspects. Since these examples also convey valuable information, we should no longer ignore them, as the previous ATSA methods do.

Motivated by this, we focus on Aspect-Category based Sentiment Analysis (ACSA) in this paper, aiming to perform joint aspect category detection and category-oriented sentiment classification. Compared with ATSA, ACSA has the following two advantages: On the one hand, for each aspect mentioned in a

---

*Corresponding author

review: *The food here is rather good, but only if you like to wait for it.*
aspect term:*food*   **category**:Food   **sentiment**:positive
aspect term:*NULL*   **category**:Service   **sentiment**:negative

Table 1: Example of Aspect-Based Sentiment Analysis.

	Number	**Total**	**Percentage**
Rest-15	623	2499	24.93%
Rest-16	836	3368	24.82%

Table 2: Percentage of implicit aspects in Restaurant datasets of SemEval 2015 and 2016.

review sentence, even if it does not have the corresponding aspect term, there must be a corresponding aspect category so that we can identify user's sentiment over it. On the other hand, from the perspective of application in real scenarios, although ACSA does not extract the aspect terms explicitly, it already meets the demand of opinion summary from multiple aspect level granularity.

However, research in this area is relatively rare, and only a few preliminary studies have been carried out. Schmitt et al. (2018) proposed a joint model by extending sentiment labels with one more dimension to indicate the occurrence of each aspect category, which is shown to outperform traditional pipeline methods. Another feasible solution is to perform Cartesian product for aspect categories and sentiment labels, which essentially performs multi-label sentiment classification for each aspect category. Nevertheless, most of these methods fail to explicitly model the hierarchical relationship between aspect category detection and category-oriented sentiment classification. In particular, when there are many aspect categories, it is difficult for these methods to learn the inner-relations among multiple categories and the inter-relations between categories and sentiments.

In this paper, we re-formalize the task as a category-sentiment hierarchy prediction problem, which contains a two-layer hierarchy output structure. The lower layer is to detect aspect categories, which can be modelled as a multi-label classification problem (i.e., one review may contain more than one category). The higher layer is to perform category-oriented sentiment classification, which can be modelled as a multi-class classification problem for each detected category.

Under the hierarchy output structure, our model contains three modules: the bottom module leverages BERT to obtain hidden representations of the two sub-tasks respectively. In the middle module, we propose a Hierarchical Graph Convolutional Network (Hier-GCN), where the lower-level GCN is to model the inner-relations among multiple categories, and the higher-level GCN is to capture the inter-relations between categories and category-oriented sentiments. Based on the interactive representations generated from Hier-GCN, the top module performs category-sentiment hierarchy prediction to generate the final output.

We conduct experiments on four benchmark product review datasets from SemEval 2015 and 2016. The results prove that the hierarchy output structure achieves better performance than other existing structures. On this basis, the proposed Hier-GCN architecture can bring additional performance gains, and consistently achieves the best results across the four datasets. Further analysis also proves the effectiveness of Hier-GCN in cases of both explicit aspect and implicit aspect.

2 Problem Formalization

Given a review sentence with n words $\boldsymbol{r}=[w_1,\ldots,w_n]$, Aspect Category-based Sentiment Analysis (ACSA) aims to detect all the mentioned aspect categories, and identify the sentiment for each detected category. Formally, let $\mathcal{C} = \{c_1,\ldots,c_m\}$ be a set of m predefined aspect categories, and $s = \{positive, negative, neutral\}$ be the label set of sentiment polarities. Accordingly, for each input $\boldsymbol{r}$, the goal of ACSA is to generate a set of category-sentiment pairs, denoted as $\{\ldots, (\hat{y}_i^c, \hat{y}_i^s), \ldots\}$, where $\hat{y}_i^c$ is the i-th aspect category mentioned in $\boldsymbol{r}$, $\hat{y}_i^s$ is its corresponding sentiment.

As shown in Fig. 1(a), one possible solution to this task is to consider all the combinations of category-sentiment pairs, denoted by Cartesian Product. Specifically, for each aspect category c_i, we can model its sentiment prediction as a multi-label classification task to generate the output $\hat{y}_i^s \in \{0, 1, 2\}$, which indicates negative, neutral, and positive sentiment polarity, respectively. Note that if $\hat{y}_i^c = 0$, it indicates the absence of the category c_i and its corresponding sentiment polarity. Since Cartesian Product suffers from the risk of generating multiple sentiments for each category, an alternative solution is to add one

Figure 1 region:

(a) Cartesian product

	😊	😐	☹️
Food	1	0	0
Service	0	0	1
Price	0	0	0

(b) Add one dimension

😊	😐	☹️	N/A	
1	0	0	0	Food
0	0	1	0	Service
0	0	0	1	Price

(c) Hierarchy

😊	😐	☹️	😊	😐	☹️
1	0	0	0	0	1

1	1	0
Food	Service	Price

Figure 1: Comparison between hierarchical prediction structure and other available structures. (a) refers to "Cartesian product", (b) refers to "Add one dimension", and (c) refers to "Hierarchy" methods, respectively.

label to the sentiment label space to predict the presence or absence of each category (Schmitt et al., 2018). In this case, the occurrence and the sentiment prediction of each category is unified as a simple multi-class classification problem.

However, both approaches simply unify aspect category detection and category-oriented sentiment classification as one task, while ignoring the internal relationship between the two sub-tasks. Intuitively, if an aspect category is not detected, it is unnecessary to perform sentiment classification for it, and if multiple categories are detected, the relations among their sentiment outputs should be fully utilized. Motivated by this, we re-formalize the ACSA task as a category-sentiment hierarchy prediction problem:

$$p(\boldsymbol{y}^c, \boldsymbol{y}^s | \boldsymbol{r}) = p(\boldsymbol{y}^c | \boldsymbol{r}) p(\boldsymbol{y}^s | \boldsymbol{y}^c, \boldsymbol{r}), \tag{1}$$

where the first component $p(\boldsymbol{y}^c | \boldsymbol{r})$ is modeled as a multi-label classification problem for aspect category detection, and the second component $p(\boldsymbol{y}^s | \boldsymbol{y}^c, \boldsymbol{r})$ is modeled as a multi-class classification problem to perform sentiment classification for each detected category.

3 The Proposed Approach

The overall architecture of the proposed framework is illustrated in Fig. 2. We use BERT as the basic encoder of our model to encode the contextual information of a sentence and generate context-aware category representations and sentiment representations, respectively (Section 3.1); Next, we use two hierarchical graph convolution networks to capture the inner-relation between multiple categories and the inter-relation between categories and sentiment polarities, respectively (Section 3.2); Finally, a hierarchical output and integration module is designed to construct the final prediction from category prediction and sentiment prediction (Section 3.3).

3.1 Feature Extraction with BERT

We adopt Bidirectional Encoder Representations from Transformers (BERT) as our sentence encoder, which is a pre-trained on a huge amount of text with masked language model and has been shown to achieve state-of-the-art results on a broad set of NLP tasks. Let $\boldsymbol{H} \in \mathbb{R}^{d \times (n+2)}$ denote the final hidden states generated from BERT, where we insert two special tokens (i.e., [CLS] and [SEP]) at the beginning and the end of each input $\boldsymbol{r}$. For space limitation, we omit a detailed description of BERT and refer readers to (Devlin et al., 2018).

For category representations, we further use m separate self-attention sub-layers on top of $\boldsymbol{H}$ to get the representations of m categories, denoted by $\boldsymbol{C} \in \mathbb{R}^{d \times m}$. Besides, following the practice in (Devlin et al., 2018), we employ the hidden state of the first token [CLS] as the shared sentiment representation for each category, denoted by $\boldsymbol{S} \in \mathbb{R}^d$.

3.2 Hierarchical Graph Convolutional Network (Hier-GCN)

Motivation. As introduced before, we model the task of ACSA as a hierarchy prediction problem with two sub-tasks: Aspect Category Detection and Category-oriented Sentiment Classification. For Aspect Category Detection, since some aspect categories tend to frequently co-occur with each other, it is necessary to model the inner-relations between categories (e.g., in Fig. 1, if food is mentioned in a user
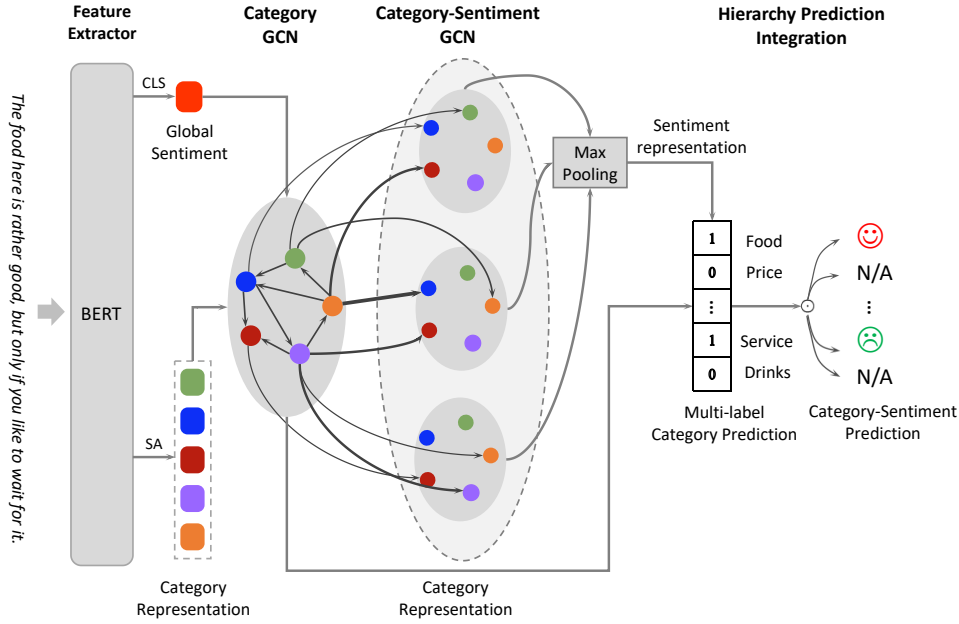
Figure 2: An overview of the proposed Hier-GCN framework. SA denote self-attention modules.

review, the service of the restaurant is likely to be mentioned as well). Similarly, for Category-oriented Sentiment Classification, since the sentiment prediction is highly dependent on the prediction of aspect category, it is crucial to model the inter-relations between categories and sentiment (e.g., in Fig. 1, we only need to predict sentiment for Food and Service categories but not the others). As the graph convolutional network (GCN) can better capture the two kinds of relations by incorporating category-category and category-sentiment co-occurrence as prior knowledge, we employ it to model the inner-realtions and the inter-relations simultaneously. Specifically, we propose a hierarchical GCN model with two sub-layers, where the lower GCN sub-layer is to learn the inner interactions between categories, and the higher GCN sub-layer is to learn the inter interactions between categories and sentiments. The two sub-layers are coupled together as a Hier-GCN structure.

3.2.1 Category GCN Sub-Layer

To adapt GCN to model the inner-relations between multiple categories, we construct a directed graph by treating each category as a node, and obtain the adjacent matrix $M^c \in \mathbb{R}^{m \times m}$ by calculating the co-occurrence between every category pair over the entire training corpus. Formally, we use $M^c_{i,j}$ to denote the transition probability of having the j-th category given the i-th category, and $M^c \in \mathbb{R}^{m \times m}$ can be computed as follows:

$$M^c_{i,j} = \begin{cases} \frac{\mathrm{count}(c_i, c_j)}{\mathrm{count}(c_i)+1} & i \neq j, \\ 1 & i = j. \end{cases} \tag{2}$$

where $\mathrm{count}(c_i)$ refers to the number of samples with the i-th category, and $\mathrm{count}(c_i, c_j)$ denotes the number of samples with both the i-th and the j-th category. Note that in most cases, $M^c_{i,j}$ is not equal to $M^c_{j,i}$, which indicates that for a co-occurred category pair (c_i, c_j), the category with high frequency will have a high effect on the category with low frequency but not vice versa. This implies that modeling the inner-relations between categories may help detect more categories with low frequency, which is more significant for large-scale multi-label classification with many low frequency categories.

With the adjacent matrix M^c, we perform the standard graph convolution on the node representations as follows:

$$X_{l+1} = f(W_l X_l M^c + b_l), \tag{3}$$

where C is the initial state of X, i.e., X_0, l refers to the l-th category GCN sub-layer, $W_l \in \mathbb{R}^{d \times d}$ and $b_l \in \mathbb{R}^d$ are the linear transformation weight and bias, and $f(\cdot)$ is a nonlinear activation function, i.e., GELU.

836

After obtaining the output of the $l+1$-th sub-layer $\boldsymbol{X}_{l+1}$, we not only feed it to the next category GCN sub-layer as the input category representation, but also combine it with sentiment representations as the initial state of the $l+1$-th category-sentiment GCN sub-layer. Note that in the final Hier-GCN layer, it is used to perform multi-label classification for aspect category detection.

3.2.2 Category-Sentiment GCN Sub-Layer

Similarly, to employ GCN to model the inter-relations between m categories and $m*3$ category-oriented sentiments, we construct another directed graph by treating the m categories and $m*3$ sentiments as graph nodes. Since our final goal is to predict sentiment for each detected category, we propose to obtain an adjacent matrix for the positive, negative, and neutral sentiment, respectively. We use $\boldsymbol{M}_{i,j}^{c\text{-}s}$ to denote the transition probability that the sentiment polarity of the j-th category given the i-th category:

$$\boldsymbol{M}_{i,j}^{c\text{-}s} = \begin{cases} \frac{\text{count}(c_i,(s|c_j))}{\text{count}(c_i)+1} & i \neq j\,, \\ 1 & i = j\,. \end{cases} \tag{4}$$

where $(s|c_j)$ denotes the sentiment polarity towards the j-th category, and $s \in \{positive, negative, neutral\}$.

Next, we first obtain the fused sentiment-sensitive category representation for each input node to derive $\widehat{\boldsymbol{F}}_l \in \mathbb{R}^{m \times d}$ as follows:

$$\widehat{\boldsymbol{F}}_l = \text{Tanh}(\boldsymbol{W}_l^{c,s}\boldsymbol{X}_{l+1} \oplus \boldsymbol{S}_l + \boldsymbol{b}_l^{c,s}), \tag{5}$$

where $\oplus$ denotes the concatenation operation, and $\boldsymbol{W}_l^{c,s} \in \mathbb{R}^{d \times 2d}$ and $\boldsymbol{b}_l^{c,s} \in \mathbb{R}^d$ refer to the weight and bias.

With the node representation and the three adjacent matrices, the standard graph convolution is then performed on the input nodes to obtain $\widetilde{\boldsymbol{F}}_l \in \mathbb{R}^{m \times d}$:

$$\widetilde{\boldsymbol{F}}_l^s = f(\boldsymbol{W}_l^s \widehat{\boldsymbol{F}}_l \boldsymbol{M}^{c\text{-}s}), \tag{6}$$

where $\boldsymbol{W}_l^s \in \mathbb{R}^{d \times d}$ and bias are learnable parameters.

Finally, we can obtain the category-oriented sentiment representation $\boldsymbol{S}^{l+1}$ as follows:

$$\boldsymbol{S}_{l+1} = \text{pooling}(\text{dense}(\widetilde{\boldsymbol{F}}_l^{pos}); \text{dense}(\widetilde{\boldsymbol{F}}_l^{neg}); \text{dense}(\widetilde{\boldsymbol{F}}_l^{neu})), \tag{7}$$

where $\text{dense}(\cdot)$ is a parameter sharing feed-forward network with tanh as the activation function.

3.3 Hierarchical Prediction Integration

Based on the final category representation $\boldsymbol{X}_i$ and sentiment representation $\boldsymbol{S}_i$ generated from the L-th layer Hier-GCN, we can obtain the probability of having the i-th category $p_i^c \in \mathbb{R}^1$ and its corresponding sentiment probability distribution $\boldsymbol{p}_i^s \in \mathbb{R}^3$ as follows:

$$p_i^c = p(y_i^c|\boldsymbol{r}) = \text{sigmoid}(\boldsymbol{W}_i^c \boldsymbol{X}_i + b_i^c), \tag{8}$$

$$\boldsymbol{p}_i^s = p(\boldsymbol{y}_i^s|y_i^c, \boldsymbol{r}) = \text{softmax}(\boldsymbol{W}^s \boldsymbol{S}_i + \boldsymbol{b}^s), \tag{9}$$

where the parameters $\boldsymbol{W}^s$ and b^s are used for sentiment classification, which is shared for all categories.

Then, Hierarchical Prediction Module is used to get the final prediction of the i-th category-sentiment pair $(\hat{y}_i^c, \hat{y}_i^s)$:

$$(\hat{y}_i^c, \hat{y}_i^s) = (\mathbb{I}(p_i^c > 0.5), \arg\max \boldsymbol{p}_i^s), \tag{10}$$

where $\mathbb{I}(\cdot)$ is an indicator function.

The loss of ACSA has two parts, in correspondence with Equation (1). The first component (category detection) is modeled as a multi-label classification problem, and the cross-entropy loss, i.e., negative log-likelihood of $p(\boldsymbol{y}^c|\boldsymbol{r})$, is defined as:

$$loss^c = -\sum_{i=1}^m y_i^c \log p_i^c + (1 - y_i^c)\log(1 - p_i^c), \tag{11}$$

	Restaurant-15	Laptop-15	Restaurant-16	Laptop-16
# Train	1102	1397	1680	2037
# Test	572	644	580	572
# Categories	30	198	30	198
# Positive Sentences	1458	1641	1966	2113
# Negative Sentences	663	1091	848	1353
# Neutral Sentences	91	185	137	234

Table 3: Data statistic. # Train and # Test mean the sentence number of train and test sets, respectively. # Categories denotes the number of predefined categories, and # Positive/Negative/Neutral Sentences refers to the number of sentences of corresponding sentiment.

where $y_i^c \in \{0, 1\}$ is the ground truth of the i-th category. The second component (category-oriented sentiment classification) is modeled as a multi-class classification problem for each category, and the cross-entropy loss, i.e., negative log-likelihood of $p(\boldsymbol{y}^s | \boldsymbol{y}^c, \boldsymbol{r})$, is defined as:

$$loss^s = -\sum_{i=1}^{m} \sum_{j=1}^{3} \mathbb{I}(y_{i,j}^s) \log p_{i,j}^s, \tag{12}$$

where $y_{i,j}^s$ is the ground truth of j-th sentiment for the i-th category. The cross-entropy loss is equivalent to negative log-likelihood. Therefore, the final loss function of ACSA, i.e., negative log-likelihood of $p(\boldsymbol{y}^c, \boldsymbol{y}^s | \boldsymbol{r})$, equals the sum of two component cross-entropy losses:

$$loss = loss^c + loss^s. \tag{13}$$

4 Experiments

4.1 Datasets and Experimental Settings

We evaluate our model on the benchmark SemEval 2015 and 2016 datasets (Pontiki et al., 2015; Pontiki et al., 2016), which contain customer reviews from laptop and restaurant domains. To ensure that aspect category can accurately describe an opinion expression object, a category is predefined as specific entity type E and its attribute label A in SemEval 2015 and 2016 datasets. The $E\#A$ pair defines an aspect category. The details of the data sets are summarized in Table 3. We compute the Precision, Recall of the predicted (category, sentiment) pairs respectively, and use Micro-F1 score as the final evaluation metric.

We randomly split the training set into the ratio of 9:1 as training, validation set. The reported results are averaged scores of 10 runs to obtain statistically stable results. We use BERT-base as basic encoder, and refer readers to (Devlin et al., 2018) for the detailed BERT-base model setting. We adopt the same AdamW optimizer in BERT. The learning rate and batch size are set to 5e-5 and 8, respectively. We set the maximum sentence length to 128 for SemEval 2015 and 100 for SemEval 2016, and run 20 epochs for every fold, a dropout rate of 0.1 is used, and the hidden size of Transformer is 768.

4.2 Compared Systems

In Section 2, we have summarized several problem formalization methods including Cartesian, Add one dimension. We use them as baseline systems for comparison. We furthermore implement a pipeline method of category detection and sentiment classification. All compared systems are as follows:

- **Cartesian-BERT**: The Cartesian method with BERT as the sentence encoder.
- **Pipeline-BERT**: Pipeline of individual aspect category detection and category sentiment classification, both with BERT as the review encoder. This method can better model each sub-task, but ignores the associations between two sub-tasks.
- **AddOneDim-LSTM**: The Add one dimension method with LSTM as the sentence encoder. This represents the state-of-the-art work by (Schmitt et al., 2018).

838

Method	Restaurant-16			Laptop-16			Restaurant-15			Laptop-15		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Cartesian-BERT	74.96	63.84	68.94	64.99	27.40	39.54	72.02	49.15	58.42	73.06	21.18	32.83
Pipeline-BERT	43.62	79.06	56.21	31.92	51.56	39.42	38.12	70.00	49.35	36.91	51.62	43.02
AddOneDim-LSTM	61.56	42.82	50.50	-	-	-	54.33	28.44	37.32	-	-	-
AddOneDim-BERT	71.75	67.95	69.79	58.83	39.49	47.23	68.84	55.86	61.67	64.17	39.57	48.94
Hier-BERT	70.97	69.65	70.30	59.51	41.93	49.19	67.46	57.98	62.36	65.47	41.26	50.61
Hier-Transformer-BERT	73.72	73.21	73.45	58.06	48.29	52.72	70.22	59.96	**64.67**	65.63	51.95	57.79
Hier-GCN-BERT	76.37	72.83	**74.55**	61.43	48.42	**54.15**	71.93	58.03	64.23	71.90	54.73	**62.13**

Table 4: Main results of the ACSA task on four benchmark datasets.

- **AddOneDim-BERT**: The Add one dimension method with BERT as the sentence encoder.

- **Hier-BERT**: The hierarchy method with BERT as the sentence encoder.

- **Hier-Transformer-BERT**: As a strong baseline, we use Transformer to model the inner-relations between category and inter-relations between category and sentiment based on Hier-BERT.

- **Hier-GCN-BERT**: Our proposed hierarchy prediction model with Hier-GCN as the relation learning module based on Hier-BERT.

It should be noted that except AddOneDim-LSTM, all the other systems are proposed in this work.

4.3 Main Results

Table 4 shows the results of different systems on two datasets. It can be observed that AddOneDim-LSTM seems difficult to achieve good performance when the number of categories is large. For example, in the Laptop dataset, the training algorithm fails to converge. In comparison, AddOneDim-BERT can perform much better. Cartesian-BERT is less effective probably due to the class-parse problem especially when the number of categories is large (e.g., Recall on Laptop is very low). The precision of the pipeline method is low although its recall is high, because it ignores the relations among sub-tasks and leads to too many sentiment predictions without taking category-sentiment restriction into consideration. This also indicates that our proposed category-sentiment hierarchy structure is more suitable for joint aspect category detection and category-oriented sentiment classification.

We can see that among all systems, our hierarchical methods (Hier, Hier-Transfomer-BERT, Hier-GCN-BERT) achieves significantly better F1 performance than the other baseline systems, among which, Hier-GCN-BERT gets the best performance on three datasets. This indicates that the modeling of hierarchical relationship can better describe the original problem on the basis of hierarchical label prediction.

4.4 Discussions on Hier-GCN

4.4.1 GCN vs Transformer

In Table 4, we further present the result of Hier-Transformer-BERT where we replace GCN with Transformer to learn the relations among categories and sentiments. The layers of Transformer is kept the same as that in Hier-GCN-BERT. The first Transformer layer is used to model the inner-relation between categories, and the second layer is used to capture the inter-relation between category and sentiment. It can be found that Hier-Transformer-BERT is quite efficient but still lower than Hier-GCN-BERT on three datasets, only slightly better on Restaurant 2015. The improvement of GCN upon Transformer is more substantial when the number of categories is larger (e.g., the Laptop dataset), due to the GCN's ability in learning the inner-relations between categories and inter-relations between category and sentiment.

4.4.2 Ablation Study

In Table 5, we conduct experiments to test the effectiveness of each component in Hier-GCN-BERT.

- **C-GCN-BERT**: only category GCN without category-sentiment GCN;
- **CS-GCN-BERT**: only category-sentiment GCN without category GCN.

It can be seen that both C-GCN-BERT and CS-GCN-BERT have incremental effects compared with the basic hierarchy prediction model Hier-BERT. Among two sub-GCNs, category GCN plays a more important role than category-sentiment GCN, because it captures the inner-relations among multiple categories and lays the basis of the following category-sentiment inter-relation learning.

	Restaurant-16	Laptop-16	Restaurant-15	Laptop-15
Hier-GCN-BERT	**74.55**	**54.15**	64.23	**62.13**
C-GCN-BERT	74.01	54.09	**64.88**	61.43
CS-GCN-BERT	73.51	52.64	63.40	57.24
Hier-BERT	70.30	49.19	62.36	50.61

Table 5: Ablation Study of Hier-GCN-BERT.

	Restaurant-16	Laptop-16	Restaurant-15	Laptop-15
1 layer	71.80	47.02	60.69	49.27
2 layers	**74.55**	**54.15**	**64.23**	**62.13**
3 layers	72.75	54.01	63.18	60.21

Table 6: Impact of the number of layers in Hier-GCN-BERT.

4.4.3 Impact of the Number of Hier-GCN Layers

We further investigate the impact of the number of Hier-GCN layers L. Considering the over-smoothing problem caused by high co-occurrence nodes, we vary L from 1 to 3, and reports the results in Table 6. The best performance is obtained when $L = 2$. Layers more than 3 in GCN may incorporate too much node co-occurrence information and result in non-discriminative representations.

4.5 Results on Implicit Aspects

As we have mentioned in Table 2, there are nearly 25% of aspect terms are not annotated in the Restaurant 2015 dataset. We refer to them as implicit aspects. To test our approach's ability to capture the implicit aspects, we split the Restaurant 2015 test set into two parts: Explicit Aspects (containing 212 samples) and Implicit Aspects (containing 360 samples), and report the performance of category-sentiment hierarchy prediction in Table 7. We find that although the performance of our approaches on the Implicit Aspects part is lower than the Explicit Aspects part, it can still accurately identify a considerable amount of category and sentiment pairs, and outperform all the compared systems in implicit aspects.

Case study: In order to verify the advantages of our model in detecting categories and predicting category sentiment polarity intuitively, we further study cases of approaches AddOneDim-BERT, Hier-BERT and Hier-GCN-BERT on implicit aspects of Restaurant 2015. As shown in Table 8, review *Once you try it for a special occasion beware.. you can't stop!* implies positive sentiment towards category miscellaneous of restaurant, only AddOneDim-BERT cannot detect the corresponding category. Review *I have never ever had such an unpleasant experience.* implies general negative sentiment towards category restaurant, AddOneDim-BERT cannot detect the mentioned category, while Hier-BERT detects the mentioned category correctly, but predicts the sentiment polarity as positive. Only Hier-GCN-BERT predicts the category-sentiment pairs correctly. Besides, in a complex context with both implicit and explicit aspects, Hier-GCN-BERT can still perform better. Review *Highly impressed from the decor to the food to the hospitality to the great night I had!* has two explicit aspects and two implicit aspects, all approaches can predict the explicit aspects, while only Hier-GCN-BERT finds one more category-sentiment pair.

5 Related Work

Aspect-Based Sentiment Analysis: Aspect-Based Sentiment Analysis (Hu and Liu, 2004; Ding et al., 2008) is an important fine-grained subtask in sentiment analysis (also called opinion mining). Most recent approaches can be categorized into two branches, namely Aspect Term-based Sentiment Analysis (ATSA) and Aspect Category-based Sentiment Analysis (ACSA).

ATSA aims to detect the aspect term and identify their corresponding sentiment polarities given a piece of text. Mitchell et al. (2013) first explored the end-to-end task by applying traditional approaches with shallow features to extract aspect terms and their sentiments jointly. Later, Zhang et al. (2015) extended

Method	Implicit Aspect			Explicit Aspect		
	P	R	$F1$	P	R	$F1$
Cartesian-BERT	67.02	39.89	49.99	74.37	54.53	62.92
Pipeline-BERT	32.43	61.14	42.38	41.71	73.73	53.28
AddOneDim-LSTM	48.81	21.82	30.15	56.84	32.29	41.17
AddOneDim-BERT	63.21	45.75	53.06	72.60	61.46	66.56
Hier-BERT	63.78	46.29	53.63	71.67	62.62	66.84
Hier-GCN-BERT	66.02	46.32	**54.41**	74.74	64.84	**69.44**

Table 7: Performance on Implicit Aspects and Explicit Aspects of Restaurant-15.

Review text	Ground-Truth	AddOneDim-BERT	Hier-BERT	Hier-GCN-BERT
Once you try it for a special occasion beware.. you can't stop!	Restaurant#MISC,NEG	Empty ✗	Restaurant#MISC,NEG	Restaurant#MISC,NEG
I have never ever had such an unpleasant experience.	Restaurant#General,NEG	Empty ✗	Restaurant#General,POS ✗	Restaurant#General,NEG
Highly impressed from the decor to the food to the hospitality to the great night I had!	Ambience#General,POS Food#Quality,POS Service#General,POS Restaurant#General,POS	Ambience#General,POS Food#Quality,POS Empty ✗ Empty ✗	Ambience#General,POS Food#Quality,POS Empty ✗ Empty ✗	Ambience#General,POS Food#Quality,POS Service#General,POS Empty ✗

Table 8: Case study on AddOneDim, Hier and Hier-GCN. Wrong predictions are marked with ✗.

the above approach by designing several neural models, and achieved better performance. Based on this, Li et al. (2019) proposed a unified neural architecture with multi-task learning for the task. Luo et al. (2019) proposed a joint model to solve the problem. Moreover, Hu et al. (2019) designed a span extraction-based neural framework and addressed the problem in an extract-then-classify manner.

Different from **ATSA**, **ACSA** targets at identifying the aspect category as well as its corresponding sentiment. Most existing studies on **ACSA** focused on its sub-tasks, namely, the aspect category detection (ACD) and category-oriented sentiment classification (CSC). For the task of ACD, (Zhou et al., 2015; Schouten et al., 2017) proposed algorithms to predict categories. More recently, Movahedi et al. (2019) proposed an attention-based neural network method to identify different aspect categories based on different topics. For the task of CSC, (Ruder et al., 2016; Wang et al., 2016; Xue and Li, 2018; Tay et al., 2018; Sun et al., 2019) proposed neural networks to make the most of context and aspect category information. (Ma et al., 2018; Hu et al., 2018) proposed attention-based networks to perform category and sentiment co-classification. Apart from the researches on the two subtasks, there are a few studies focusing on the end-to-end ACSA task. Schmitt et al. (2018) proposed a joint model and transform the task into a multi-class classification problem. However, most of these studies fail to explore the relations between ACD and CSC. Therefore, we propose a hierarchical graph network to achieve this goal.

Hierarchical Classification & GCN: The hierarchical classification problem has been studied in some previous work. Silla and Freitas (2011) defined the task of hierarchical classification and presented a new perspective about hierarchical classification approaches, Aly et al. (2019) investigated simple shallow capsule networks for hierarchical multi-label text classification. GCN can be used for non-Euclidean Structure data, Bruna et al. (2013) proposed spectral graph convolutional neural networks, and Kipf and Welling (2016) designed a convolutional architecture via a localized first-order approximation of spectral graph convolutions. Inspired by this work, we propose a BERT-based Hier-GCN model, which decomposes the inner-relations between categories and the inter-relations between sentiment polarities into two hierarchically related sub-graphs and get the final predictions in hierarchy manner.

6 Conclusion

In this paper, we first examined the limitations of existing approaches for the task of Aspect Category-based Sentiment Analysis, and proposed a hierarchy output structure to first detect the aspect categories, and then perform sentiment classification for each detected category. Given the output structure, we further proposed a Hierarchical Graph Convolutional Network to capture the inner-relations between multiple categories as well as the inter-relations between categories and sentiments. Experimental results on four benchmark datasets show that our hierarchy output structure performs significantly better than any existing output structure, and Hier-GCN can outperform a number of highly competive approaches including a strong hierarchical Transformer model proposed by us.

Acknowledgments

References

Rami Aly, Steffen Remus, and Chris Biemann. 2019. Hierarchical multi-label classification of text with capsule networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 323–330.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. 2013. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2018. Can: constrained attention networks for multi-aspect sentiment analysis. *arXiv preprint arXiv:1812.10735*.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019. Open-domain targeted sentiment analysis via span-based extraction and classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 537–546.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019. A unified model for opinion target extraction and target sentiment prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6714–6721.

Huaishao Luo, Tianrui Li, Bing Liu, and Junbo Zhang. 2019. Doer: Dual cross-shared rnn for aspect term-polarity co-extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 591–601.

Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *Aaai*, pages 5876–5883.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654.

Sajad Movahedi, Erfan Ghadery, Heshaam Faili, and Azadeh Shakery. 2019. Aspect category detection via topic-attention network. *arXiv preprint arXiv:1901.01183*.

Maria Pontiki, Dimitrios Galanis, Harris 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, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *10th International Workshop on Semantic Evaluation (SemEval 2016)*.

Sebastian Ruder, Parsa Ghaffari, and John G Breslin. 2016. A hierarchical model of reviews for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 999–1005.

Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1114.

Kim Schouten, Onne Van Der Weijde, Flavius Frasincar, and Rommert Dekker. 2017. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. *IEEE transactions on cybernetics*, 48(4):1263–1275.

Carlos N Silla and Alex A Freitas. 2011. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1-2):31–72.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing bert for aspect-based sentiment analysis via constructing auxiliary sentence. *arXiv preprint arXiv:1903.09588*.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

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*, pages 606–615.

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.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 612–621.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2015. Representation learning for aspect category detection in online reviews. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*.