# C³LPGCN:Integrating Contrastive Learning and Cooperative Learning with Prompt into Graph Convolutional Network for Aspect-based Sentiment Analysis

**Ye He, Shihao Zou, Yuzhe Chen, Xianying Huang***
College of Computer Science and Engineering, Chongqing University of Technology
wldsj_cqut@163.com

## Abstract

Aspect-based Sentiment Analysis (ABSA) is a fine-grained task. Recently, using graph convolutional networks (GCNs) to model syntactic information has become a popular topic. In addition, a growing consensus exists to enhance sentence representation using contrastive learning. However, when modeling syntactic information, incorrect syntactic structure may introduce additional noise. Meanwhile, we believe that contrastive learning implicitly introduce label information as priori. Therefore, we propose C$^3$LPGCN, which integrates Contrastive Learning and Cooperative Learning with Prompt into GCN. Specifically, to alleviate the noise when modeling syntactic information, we propose mask-aware aspect information filter, which combines prompt information of template with aspect information to filter the syntactic information. Besides, we propose prompt-based contrastive learning and cooperative learning to utilise the label information further. On the one hand, we construct prompts containing labels for contrastive learning, by which the model can focus more on task-relevant features. On the other hand, cooperative learning further extracts label information by aligning input samples' representation and output distribution with label samples. Extensive experiments on three datasets demonstrate that our method significantly improves the model's performance compared to traditional contrastive learning methods. Moreover, our C$^3$LPGCN outperforms state-of-the-art methods. Our source code and final models are publicly available at github[1].

## 1 Introduction

Aspect-based Sentiment Analysis (ABSA) (Zhang et al., 2021, 2022a) aims to predict the sentiment polarity of a specific aspect in a sentence. Figure 1 shows a restaurant review in which the sentiment expression of "Indian" is "authentic" and the sentiment expression of "prices" is "amazing". Therefore, we discriminate the sentiment polarity of these two aspects as positive.

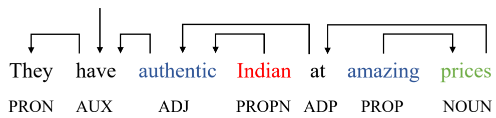



Figure 1: Example of the ABSA task.

In recent years, with the development of deep learning, research on ABSA has resulted in many successes. At the beginning, recurrent neural networks(RNNs) and convolutional neural networks(CNNs) were used to extract effective features from sentences. However, since they cannot model long-distance relationships, the performance of the model drops considerably when aspect words are far away from the corresponding sentiment expression. Then, researchers employed long short-term memory(LSTM) (Tang et al., 2015; Wang et al., 2016) and attention mechanism (Song et al., 2019; Ma et al., 2017) to encode long-distance dependencies, and these studies achieved remarkable results. Recently, with the rise of graph neural networks(GNNs) (Tang et al., 2020; Wang et al., 2020; Li et al., 2021), studies have been conducted to achieve syntactic-based aggregation of word information via graph convolutional network(GCN) (Tian et al., 2021; Sun et al., 2019) to enhance the performance of ABSA further. However, when the syntactic structure of sentences is incomplete or there is no apparent syntactic relationship between aspects and sentiments, syntactic parser may output incorrect syntactic dependency adjacency matrices, thereby leading to noise in the modeling of syntactic information.

In addition, to enable better modelling of aspect and sentiment, many studies employ contrastive learning (Liang et al., 2021; Wang et al., 2022; Liu et al., 2022) to enhance sentence features. Supervised contrastive learning typically construct con-

[1] https://github.com/godlikehhd/C3LPGCN



*Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3237–3247
June 16-21, 2024 ©2024 Association for Computational Linguistics

trastive samples based on labels, where the positive samples for an input sample are other sentences with the same polarity, and negative samples are sentences with different polarities. In unsupervised contrastive learning, data augmentation methods are commonly used to construct positive samples for an input sample, while using other input samples as negative samples. We believe that this approach implicitly introduces label information as a prior into the model.

In this paper, we propose a novel $C^3$LPGCN, integrating Contrastive Learning and Cooperative Learning with Prompt into Graph Convolutional Network. On the one hand, we propose mask-aware aspect information filter(MAF), which combines the information in prompt templates with aspect information, filtering syntactic information through attention mechanism. Prompt tuning (Lester et al., 2021; Wang et al., 2023) is a method to convert downstream task into mask prediction by constructing auxiliary template. Due to the properties of the pre-trained language model(PLM) (Devlin et al., 2018; Liu et al., 2019), we can use the prompt representation to get the location of sentiment expression. And therefore we can alleviate the noise generated by modelling syntactic information.

On the other hand, to further utilize label information, we propose prompt-based contrastive learning and cooperative learning. Specifically, we construct template containing sentiment labels and perform contrastive learning, the positive sample of input is sentence with true label template and the negative samples are sentences with false label templates. By this way, the model can focus more on task- and sentiment-related information during feature learning. While in cooperative learning, we make the representations of input samples and positive samples feature consistent by calculating their KL divergence; at the same time, we pass them through the same network for sentiment analysis and use the output distribution of the positive samples as the label of the input samples. In this way, the prior knowledge contained in the positive samples can be further learned.

Our contributions can be summarized as follows:

- We propose $C^3$LPGCN mitigate the noise generated when modelling syntactic information by utilizing PLM's prediction. Meanwhile, we propose using label information as an explicit priori to learn aspect- and sentiment-related information adequately.
- We propose MAF, which models the relationship between aspect and sentiment representation by using the information of prompt template, thus mitigating the noise that can occur during modeling syntactic information.
- We propose prompt-based contrastive and cooperative learning, which explicitly incorporates label information as a prior into the model. Prompt-based contrastive learning learns task and sentiment-related features through contrasts based on label templates. Cooperative learning further learns label information by aligning the features of input samples with those of positive samples.
- Extensive experiments on three datasets show that our method can be combined with existing contrastive learning methods to perform better, and our $C^3$LPGCN method outperforms state-of-the-art methods.

## 2 Related Work

### 2.1 Aspect-based Sentiment Analysis

With the development of deep learning, ABSA has achieved good performance. Several studies utilized attention mechanisms and LSTM to extract deep semantic information from sentences. Ma et al. (2017) proposed IAN to model the relation between aspect and context. Song et al. (2019) proposed an attention encoder to map the semantic interactions between aspect and context.

Subsequently, modelling syntactic information became a research hotspot. Li et al. (2021) alleviated the noise generated while modelling syntactic information by interactively incorporating syntactic and contextual information. Zhang et al. (2022b) proposed a self-attention-based aspect-aware attention mechanism to learn aspect-related semantic associations and global semantics. Ma et al. (2023) proposed using Abstract Meaning Representation to replace syntactic dependency trees and strengthen sentence features through an attention mechanism.

### 2.2 Constrastive learning

Contrastive learning enables the model to learn the differences or similarities between samples by constructing contrastive samples, leading to better performance in downstream tasks. Chen et al. (2020) proposed SimCLR, which performs data augmentation on images and uses them as positive samples and uses other images as negative samples by which the contrast loss is optimized. Gao et al. (2021) utilized dropout as data augmentation for contrastive learning and obtained good result. In ABSA, Liang et al. (2021) leveraged contrastive learning to distinguish sentiment features from the perspectives of sentiment polarity and patterns. Liu et al. (2022) proposed eliminating the interference of aspect-irrelevant features through feature distillation and utilising supervised contrastive learning to capture internal information between sentences. Li et al. (2023) conducts supervised contrastive learning on different aspects, reducing the representation differences of aspects within the same relationship category.

### 2.3 Prompt Tuning

Prompt tuning is an approach that transforms downstream tasks into mask prediction tasks. Recently, Schick and Schütze (2020) proposed PET, which uses prompt tuning to make PLM understand the given task and then implements semi-supervised learning on a large scale of unlabeled data by assigning soft labels. Jiang et al. (2020) proposed a method based on encoding transformation to improve the PLM's ability to extract knowledge. Chen et al. (2022b) introduced KnowPrompt, which injects potential knowledge contained in relation labels into learnable prompt construction and uses this for relation extraction.

## 3 Proposed Model

Figure 2 shows an overview of C$^3$LPGCN. In this section, we first introduce the definition of the ABSA task. After that, we will present our proposed C$^3$LPGCN, composed of five components: input construction and embedding layer, contrastive learning, prompt-based cooperative learning, GCN layer and mask-aware aspect information filter layer.

### 3.1 Problem Formulation

For a given sentence $S$ and its corresponding aspect $a$, where $S = \{w_1, w_2, ..., w_n\}$, $a = \{a_1, a_2, ..., a_k\}$, $a$ is a subsequence of $S$, ABSA is to predict the sentiment polarity $y \in \{positive, negative, neutral\}$ of the given aspect. For the sake of simplicity, we perform prediction on one aspect at a time for sentences containing multiple aspects.

### 3.2 Input Construction and Embedding Layer

In contrast to other studies that use sentence-aspect pair as input, we construct prompt templates specific to the ABSA task and concatenate them with the sentence, using them as input of the BERT encoder. For a given sentence $S$ and the aspect $a$, we construct its prompt template:

$$T_{prompt} = [p_1, p_2, ..., a, ..., \text{[MASK]}], \quad (1)$$

where $p_i$ is the constructed template, while [MASK] is the token of PLM's masking process. Taking "No disk is included" as an example, where the aspect is "disk", we can construct a template like "the disk is [MASK]." Then, we can get a sample input for BERT:

$$S_{in} = [\text{[CLS]}, S, \text{[SEP]}, T_{prompt}, \text{[SEP]}] \quad (2)$$

Feeding $S_{in}$ into BERT, we can obtain its representation $H_{in}$ To perform prompt-based contrastive learning and cooperative learning, we construct label samples for the input, i.e., replacing [MASK] with the labels we set in the prompt template. We set up three kinds of label templates based on the real sentiment labels of the training data:

$$\begin{aligned} T_{pos} &= [p_1, p_2, ..., a, ..., L_{pos}], \\ T_{neg1} &= [p_1, p_2, ..., a, ..., L_{neg1}], \\ T_{neg2} &= [p_1, p_2, ..., a, ..., L_{neg2}], \end{aligned} \quad (3)$$

where $L_{pos}$ is the true sentiment label of $S$ and $L_{neg1}$, $L_{neg2}$ are the false sentiment labels we constructed. Similarly concatenating them with $S$ and feeding them into BERT, we can obtain their representations $H_{pos}$, $H_{neg1}$, $H_{neg2}$. It can be seen that the input sample and contrastive samples are identical in form , both can be represented as:

$$H_i = \{h^i_{cls}, h^i_1, ..., h^i_{n+m+2}\} \quad (4)$$

where $i \in \{in, pos, neg1, neg2\}$, $H_i \in \mathbb{R}^{t \times d_{bert}}$, $n$ and $m$ denote the length of $S$ and the template, respectively, $t = m+n+2$. We conducted template experiment in Appendix A.

### 3.3 Contrastive Learning

In this section, we use supervised and prompt-based contrastive learning to improve further the model's ability to model aspects and sentiment expression.

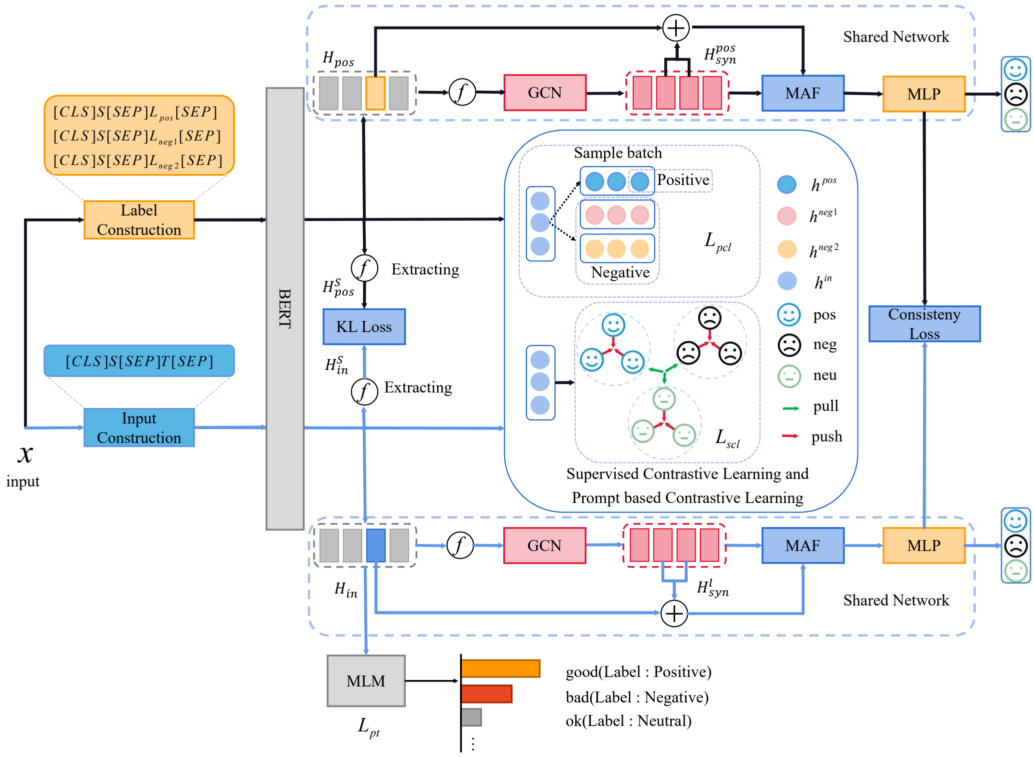



Figure 2: Overall architecture of the proposed C[3]LPGCN.

#### 3.3.1 Supervised Contrastive learning

Same as other supervised contrastive learning methods, for any input sample $H_{in}^i$, we take the samples with the same polarity within a batch $B$ as positive samples. Otherwise, it is negative. Then, the contrastive loss is formulated as follows:

$$\mathcal{L}_{scl} = \frac{1}{N}\sum \mathcal{L}_{sup}(h_i^{in}),$$
$$\mathcal{L}_{sup}(h_i) = -\log \frac{\sum\limits_{y(a_i)=y(a_j)} sim(h_i, h_j)}{\sum\limits_{j\in B} sim(h_i, h_j)}, \tag{5}$$

where $N$ is the batch size and $h_i^{in}$ is the pooled output of $H_{in}^i$, $sim(\cdot)$ is the cosine similarity, $y(a_i) = y(a_j)$ denotes $h_i$ has the same sentiment polarity as $h_j$. With supervised contrastive learning, sentences with the same sentiment polarity are brought closer in feature space, while the distance between sentences with different sentiment polarities is pushed farther apart.

#### 3.3.2 Prompt-based Contrastive Learning

In supervised contrastive learning, we use sentiment polarity to construct contrastive samples, equivalent to implicitly introducing label information as priori into feature learning. To further utilize label information, we propose prompt-based contrastive learning to introduce label information explicitly. For a input sample $H_{in}^i$, we constructed its corresponding label samples $H_{pos}^i$, $H_{neg1}^i$, $H_{neg2}^i$. Thus, our training objective can be formulated as follows:

$$\mathcal{L}_{pcl} = \frac{1}{N}\sum \mathcal{L}_p(h_i^{in}),$$
$$\mathcal{L}_p(h_i^{in}) = -\log \frac{sim(h_i^{in}, h_i^{pos})}{\sum\limits_{j\in B} sim(h_i^{in}, h_j^{all})}, \tag{6}$$

where $h_j^{all}$ denotes all the false label samples we constructed in batch $B$. Compared to supervised contrastive learning, our method explicitly introduces the true sentiment labels, thus allowing the model to learn information related to ABSA more directly during representation learning.

### 3.4 Cooperative Learning

To further utilize the prior knowledge contained in the true label samples, we propose cooperative learning, which consists of two components; on the one hand, for the input representation $H_{in}$ and its true label sample $H_{pos}$, we take the representations of the corresponding parts $H_{in}^S$, $H_{pos}^S$ of the original sentence $S$. After that, we compute the KL divergence between them to learn the prior distribution of true label samples:

$$\mathcal{L}_{KL} = \sum KL(H_{in}^{S} || H_{pos}^{S}) \tag{7}$$

On the other hand, we feed the true label sample and the input sample into the same ABSA network and obtain their predicted distribution $p(a)$, $p_{pos}(a)$, and use the $p_{pos}(a)$ as the label of $p(a)$ to calculate the consistency loss:

$$\begin{aligned} &y_{pos}(a) = argmax(p_{pos}(a)), \\ &\mathcal{L}_{CL} = -\sum_{S}\sum_{a \in A_S} y_{pos}(a) \cdot \log p(a) \end{aligned} \tag{8}$$

where $A_S$ is the aspect collection of the sentence $S$.

### 3.5 GCN Layer

We leverage syntactic dependency trees to aid the model in learning syntactic features and establish the relationship between aspect and sentiment. We use the LAL-Parser (Mrini et al., 2019) to obtain the adjacency matrix of the dependency tree for the sentence. The syntactic dependency adjacency matrix $A$ for each sentence is constructed by the following rule:

$$A_{ij} = \begin{cases} 1 & \text{if } i = j, \text{(self loop)}, \\ 1 & \text{if } i \text{ and } j \text{ are dependent}, \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Afterwards, we use GCN to aggregate the syntactic information. Given the sentence representation $H_{syn}^{l-1}$ of layer $(l-1)$ and $A$, the $l-$th representation is defined as follows:

$$H_{syn}^{l} = \text{RELU}(AH_{syn}^{l-1}W_{syn}^{l} + b_{syn}^{l}) \tag{10}$$

where $W_{syn}^{l}, b_{syn}^{l}$ are trainable parameters of the $l-$th layer. And $H_{syn}^{0}$ is the part of input representation $H_{in}$ corresponding to the original sentence $S$, that is $H_{in}^{S}$.

### 3.6 Mask-aware Aspect Information filter

We introduce prompt tuning into ABSA to mitigate the noise generated when modelling syntactic information. The training process of PLM shows that the model's prediction of the mask position depends on the contextual information. PLM's prediction of the mask position certainly incorporates the understanding of the prompt we constructed and the sentence. Therefore, we propose mask-aware aspect information filter, which filters syntactic information by combining the prompt-tuned information with aspect information.

Given a masked language model $L$, we feed the representation of input samples $H_{in}$ into it, resulting in predictions for the [MASK] position in the prompt template (e.g., great(positive), terrible(negative)). The process is depicted as follows:

$$\begin{aligned} &H_{MLM} = \text{GELU}(H_{in}W_{MLM} + b_{MLM}), \\ &H_{out} = H_{MLM}W_{out} + b_{out}, \end{aligned} \tag{11}$$

where $H_{MLM} \in \mathbb{R}^{t \times d_{bert}}$, $H_{out} \in \mathbb{R}^{t \times d_{vocab}}$, $W_{MLM}$, $b_{MLM}$, $W_{out}$, $b_{out}$ are trainable parameters. Subsequently, we define a mapping function $\mathcal{M} : \mathcal{Y} \rightarrow \mathcal{V}$ to map the true sentiment labels to the output words of the masked language model. By doing so, we can obtain the predicted probabilities $p_{pt}(a)$ for the true sentiment polarity $y(a)$ of the aspect in the sentence:

$$p_{pt}(a) = p(\text{[MASK]} = \mathcal{M}(y(a))|H_{out}) \tag{12}$$

Subsequently, we utilize cross-entropy as the loss function to fine-tune the PLM and the masked language model:

$$\mathcal{L}_{pt} = -\sum_{S}\sum_{a \in A_S} y(a) \cdot \log\left(p^{pt}(a)\right) \tag{13}$$

In the process of prompt tuning, we utilize MAF to combine the representation of mask position $h_{mask}^{in}$ with aspect information $h_{a}^{in}$ to achieve the filtering of syntactic information. The formulas are as follows:

$$\begin{aligned} &H'_{syn} = H_{syn}^{l}W'_{syn} + b'_{syn}, \\ &h_{MAF} = \Big(\frac{1}{k}\sum_{i=1}^{k} h_{a_i}^{syn} + h_{mask}^{in}\Big)W_a + b_a, \\ &\alpha = \text{softmax}(h_{MAF} \times (H'_{syn})^{T}), \\ &h^{MAF} = \alpha H_{syn}^{l}, \end{aligned} \tag{14}$$

where $k$ is the length of aspect in the PLM, $h_{a_i}^{syn}$, $h_{mask}^{in}$ denote the representation of aspect words and [MASK] position in $H_{syn}^{l}$ and $H_{in}$, respectively. $W'_{syn}$, $b'_{syn}$, $W_a$, $b_a$ are trainable parameters. With this approach, the model can take into account both aspect information and prompt tuning information, thus mitigating the noise generated by modeling errors in syntactic information.

### 3.7 Target Aspect Sentiment Analysis

The final feature representation used for ABSA is obtained by utilizing the representations generated from the aforementioned components. The representation can be described as follows:

$$X_a = h^{in} \oplus h^{MAF} \oplus h^{in}_{mask} \tag{15}$$

where $\oplus$ is concatenation, $h^{in}$ is the pooled output of $H_{in}$ to represent the entire sentence, $h^{MAF}$ is the output of MAF, while $h^{in}_{mask}$ is the representation corresponding to the MASK position during prompt tuning. Then, we feed the obtained representations into a linear classifier with softmax to obtain the probability distribution $p(a)$ of sentiment polarity. The process can be represented as follows:

$$p(a) = \mathrm{softmax}(X_a W_p + b_p) \tag{16}$$

where $W_p$, $b_p$ are trainable parameters.

### 3.8 Loss Function

We use the loss as follows in the training process for gradient descent:

$$\begin{aligned} \mathcal{L}_{total} &= \mathcal{L}_{pre} + \lambda_1 \mathcal{L}_{pt} + \lambda_2 \mathcal{L}_{KL} \\ &+ \lambda_3 \mathcal{L}_{CL} + \lambda_4 \mathcal{L}_{scl} + \lambda_5 \mathcal{L}_{pcl} \end{aligned} \tag{17}$$

where $\lambda$s are hyperparameter, $L_{pre}$ is the loss of final classifier:

$$L_{pre} = -\sum_{S} \sum_{a \in A_S} y(a) \cdot \log(p(a)) \tag{18}$$

Table 1: Statistics of datasets.

| Dataset | Division | Positive | Negative | Neutral |
|---|---|---|---|---|
| Laptop | Train | 976 | 851 | 455 |
| | Test | 337 | 128 | 167 |
| Restaurant | Train | 2164 | 807 | 637 |
| | Test | 727 | 196 | 196 |
| Twitter | Train | 1507 | 1528 | 3016 |
| | Test | 172 | 169 | 336 |

## 4 Experiments

### 4.1 Datasets

We conducted experiments on three publicly available benchmark datasets. Laptop is a collection of user reviews and opinions about laptops and related products. The Restaurant consists of reviews and opinions about restaurants. Both the Laptop and Restaurant are from SemEval14 (Pontiki et al., 2014). Twitter (Dong et al., 2014) is a collection of tweets. The three datasets consist of sentiment polarities: 'positive', 'negative', and 'neutral'. Laptop and Restaurant include sentences with single and multiple aspects, while the Twitter dataset contains sentences with only one aspect. The statistical information for these three datasets is summarized in Table 1.

### 4.2 Baseline Models

1) **AEN** (Song et al., 2019) proposes an attention-based encoder to model the relationship between aspect and context. 2) **IAN** (Ma et al., 2017) interactively learns the relationship between aspect and their context. 3) **BERT-SPC** (Song et al., 2019) uses the representation of the [CLS] token of BERT for ABSA. 4) **DualGCN** (Li et al., 2021) simultaneously considers syntactic and semantic information for ABSA. 5) **SSEGCN** (Zhang et al., 2022b) proposes aspect-aware attention to learn semantic associations and global semantics. 6) **dotGCN** (Chen et al., 2022a) utilizes reinforcement learning to construct a language-independent discrete latent opinion tree for ABSA. 7) **DLGM** (Mei et al., 2023) proposes leveraging neurons to extract specific language attributes. 8) **APARN** (Ma et al., 2023) utilizes a new semantic structure to replace syntactic dependency tree. 9) **BERT-SCon** (Liang et al., 2021) proposes using supervised contrastive learning to distinguish sentiment features in terms of sentiment polarity and patterns. 10) **AFDEN** (Liu et al., 2022) proposes a distillation module to better learn the aspect-unrelated features and eliminate the interference of aspect-unrelated features 11) **APSCL** (Li et al., 2023) proposes a framework that capturing relationships between aspects and enhances their features through contrastive learning.

### 4.3 Implementation Details

In this experiment, all models we implemented utilize BERT-base-uncased as the pre-trained language model. When calculating the training loss, $\lambda$s is set to (0.01, 0.1, 0.1, 1.0, 0.3). We use the Adam optimizer for gradient descent. The learning rate for the PLM is set to 3e-5, while the learning rate for the other layers is set to 1e-4. In the GCN layer, we set the number of layers for the GCN in the range of [1, 3]. We use **Accuracy** and **Macro-F1** to evaluate the performance of our proposed $C^3$LPGCN as well as the baseline methods. For more implementation details, please refer to our

Table 2: Performance of different methods on the three datasets. "$*$" denotes our implementation. The best results are in **bold**, and the second-best are underlined.

| Category | Models | Laptop | | Restaurant | | Twitter | |
|---|---|---|---|---|---|---|---|
| | | ACC | Macro-F1 | ACC | Macro-F1 | ACC | Macro-F1 |
| w. Contextual information | AEN | 73.51 | 69.04 | 80.98 | 72.14 | 72.83 | 69.81 |
| | AEN+BERT | 79.93 | 76.31 | 83.12 | 73.76 | 74.71 | 73.13 |
| | IAN | 72.10 | - | 78.60 | - | - | - |
| | BERT-SPC | 79.91 | 76.30 | 85.61 | 79.05 | 76.21 | 74.78 |
| w. Syntactic information | DualGCN-BERT | 81.80 | 78.10 | 87.13 | 81.16 | 77.40 | 76.02 |
| | dotGCN-BERT | 81.03 | 78.10 | 86.16 | 80.49 | 78.11 | 77.00 |
| | SSEGCN-BERT | 81.01 | 77.96 | 87,31 | 81.09 | 77.40 | 76.02 |
| | DLGM-BERT | 82.61 | 79.24 | 87.35 | 81.88 | 74.96 | 73.37 |
| | APARN-BERT | 81.96 | 79.10 | 87.76 | **82.44** | **79.76** | **78.79** |
| w. Contrastive learning | BERT-SCon | 80.23 | 76.48 | 86.51 | 80.55 | - | - |
| | APSCL-BERT | 81.02 | 78.47 | 86.86 | 81.28 | - | - |
| | AFDEN | 82.13 | 78.81 | 87.41 | 82.21 | 78.47 | 77.27 |
| ours | BERT+SCL$^*$ | 80.54 | 77.32 | 86.24 | 79.74 | 76.07 | 74.99 |
| | BERT+PCL$^*$ | 81.01 | 76.98 | 86.60 | 79.67 | 76.66 | 75.22 |
| | BERT+C$^3$LP$^*$ | 81.80 | 78.46 | 86.68 | 80.73 | 77.55 | 76.28 |
| | Our C$^3$LPGCN | **82.75** | **79.61** | **87.85** | **82.44** | 79.32 | 78.44 |
| | w/o $\mathcal{L}_{CL}$ | 81.49 | 78.69 | 87.13 | 82.25 | 78.43 | 77.47 |
| | w/o $\mathcal{L}_{KL}$ | 81.65 | 78.04 | 87.22 | 81.71 | 77.25 | 76.18 |
| | w/o $\mathcal{L}_{pcl}$ | 81.17 | 77.68 | 86.86 | 80.60 | 76.66 | 75.55 |
| | w/o MAF | 81.08 | 77.42 | 85.43 | 77.40 | 75.63 | 74.93 |
| | +aspect | 81.33 | 77.70 | 86.33 | 80.38 | 77.10 | 75.73 |
| | +mask | 81.17 | 77.84 | 86.15 | 80.20 | 75.92 | 74.82 |

Table 3: Case studies of our C$^3$LPGCN model compared with other baselines

| Sentences | AEN+BERT | DualGCN-BERT | Our C$^3$LPGCN |
|---|---|---|---|
| From the speed to the gestures this operating system beats windows easily. | $(O_\times, O_\times, P_\surd, N_\surd)$ | $(P_\surd, P_\surd, P_\surd, N_\surd)$ | $(P_\surd, P_\surd, P_\surd, N_\surd)$ |
| It has all the expected features and a wide screen and more than roomy keyboard. | $(P_\surd, O_\times, P_\surd)$ | $(P_\surd, P_\surd, N_\times)$ | $(P_\surd, P_\surd, P_\surd)$ |
| I use it mostly for creation (audio) and its reliable. | $(P_\surd, O_F)$ | $(P_\surd, O_F)$ | $(P_\surd, P_\surd)$ |

code.

## 4.4 Main Result

We compared our model with other models, and the results are shown in Table 2. The results show that (1)Our C$^3$LPGCN obtained the best result in the three datasets. (2)Modeling syntactic information performs better than methods that model contextual information, such as attention. (3)Using PLM can make the model perform better, and it's become a consensus to use PLM. (4)Compared to methods that use syntactic information, methods that use contrastive learning methods tend to be simpler in structure and therefore perform slightly less well. (5)In our model, we combine prompt-tuned information and aspect information to filter syntactic information, thus alleviating the noise when modelling syntactic information. Also, we explicitly introduce sentiment label information using our proposed prompt-based contrastive learning and cooperative learning to obtain the best performance. (6)Compared to supervised contrastive learning, prompt-based contrastive learning can also improve the model's performance, and these two methods can be used together for better results.

## 4.5 Ablation Study

To verify the effectiveness of different modules, we performed ablation studies with the following configuration:

- w/o $\mathcal{L}_{CL}$: We no longer align the input samples with the predictions of the positive samples.
- w/o $\mathcal{L}_{KL}$: We no longer compute the KL divergence between input sample and positive sample.
- w/o $\mathcal{L}_{pcl}$: We removed the prompt-based contrastive learning.
- w/o MAF: We removed the mask-aware aspect information filter.
- +aspect: We filtered the syntactic information with the representation of aspect.
- +mask: We filtered the syntactic information with the representation of the mask position in prompt template.

As shown in Table 2. First, the model's perfor-

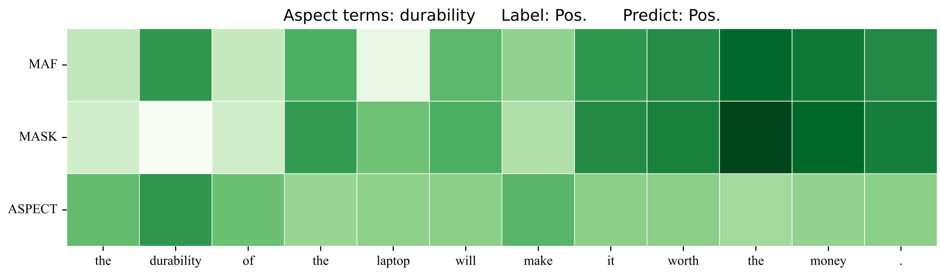



Figure 3: Visualization of attention weights calculated using different information.

mance decreased after removing supervised contrastive learning, suggesting that supervised contrastive learning can learn the similarities and differences between samples. When prompt-based contrastive learning or cooperative learning is removed, the model's effectiveness likewise deteriorates because, with prompt-based contrastive learning and cooperative learning, the model learns information relevant to ABSA from true label samples. When we use only aspect information for sentiment classification, the model becomes less effective due to the noise generated when modelling syntactic information. Similarly, the model does not perform well when using only aspect information for filtering. Whereas, when using only mask information, the model may ignore aspect information, and thus, the model becomes less effective(experiments in Sec 4.7 proved this point).

### 4.6 Case Study

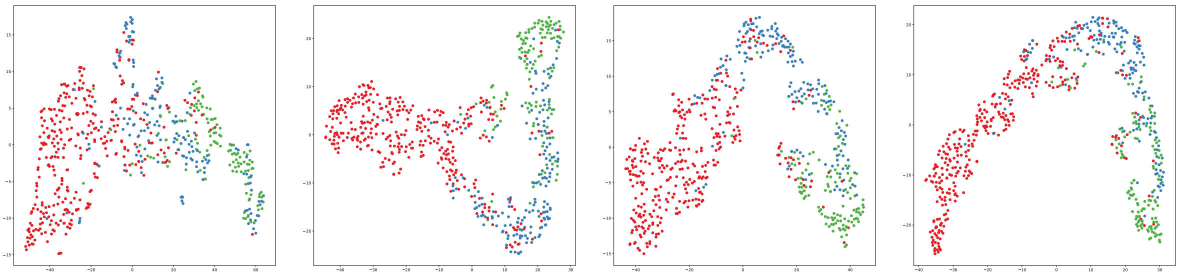
We conducted a case analysis as shown in Table 3. The notations P, N and O represent positive, negative and neutral sentiment, respectively.The results indicate that modelling syntactic information leads to better results when there is a long distance between the aspect and sentiment expression. This is because, compared to direct attention-based aggregation, GCN enables more accurate aggregation of aspect and corresponding sentiment expression. On the other hand, when there is no explicit syntactic relationship between aspect and sentiment expression, our proposed C$^3$LPGCN outperforms other models because our model not only considers syntactic information but also incorporates sentiment expression modelling information from PLM and uses contrastive learning and cooperative learning to enhance sentence features further.

### 4.7 Attention Visualization

To explore the impact of our proposed MAF, we investigated the differences in attention weights using different information for filtering. We visualized the attention weights using sentence "*the durability of the laptop will make it worth the money.*" from the laptop dataset, where the aspect is "*durability*". As shown in Figure 3. When using only the aspect information, the model assigns the highest weight to the aspect, which indicates that the model focused more on the aspect. On the other hand, when using the mask information, the model assigns the highest weight to the sentiment expression "*it worth the money.*" It is shown that by prompt tuning, it is possible to obtain the reason for the sentiment prediction, i.e., the sentiment expression. In contrast, our proposed MAF combines the MASK position and aspect information to consider both aspect and sentiment expression. Therefore, it somewhat alleviates the noise caused by wrong syntactic information.

### 4.8 Feature Visualization

To verify the effectiveness of prompt-based contrastive learning, we performed the visualization shown in Figure 4 using t-SNE (van der Maaten and Hinton, 2008). The results show that the feature distribution of different sentiments is tighter when using only BERT. After using supervised contrastive learning, the boundary distance between different sentiments increases significantly, indicating that the model learns the similarities and differences between different samples through supervised contrastive learning. Similarly, when using prompt-based contrastive learning, the feature distances of different sentiments become larger due to using false label samples of other sentences as negative samples. Still, the feature distribution of the same polarity is also slightly larger than supervised contrastive learning because there is only one positive sample for an input sample. After combining supervised contrastive learning with prompt-based contrastive learning, the boundaries of different sentiments become more obvious, and the distribu-



(a) BERT (b) BERT+SCL (c) BERT+PCL (d) BERT+SCL+PCL

Figure 4: Feature visualization of using different methods

tion of the same sentiment becomes tighter.

## 5 Conclusion

In this paper, we propose C$^3$LPGCN. On the one hand, to mitigate the noise that may arise when modelling syntactic information, we propose mask-aware aspect information filter, which filters syntactic information by combining prompt-tuned representations with aspect information. On the other hand, we propose prompt-based contrastive learning and cooperative learning methods that explicitly introduce label information. Extensive experiments on three datasets demonstrate the effectiveness of our approach.

## 6 Limitation

In this paper, we employed a manual construction approach for prompt templates. The uncertainty associated with manual construction leads to varying effects of different templates on the model performance. In future work, we plan to explore the use of continuous prompts. Additionally, we aim to extend our prompt-based contrastive learning and cooperative learning to a broader range of natural language processing tasks.

The high computational complexity is the main issue currently faced by this method. Despite the model's structure being very simple, the construction of three samples for each sentence for prompt-based contrastive learning and cooperative learning significantly increases the GPU storage and time used during the training process. In prompt-based contrastive and cooperative learning, each sentence and its constructed samples differ only in the label word. In the future, we will explore how to remove redundant parts of the samples to reduce memory usage and accelerate training.


## Acknowledgements

This work is supported in part by the National Natural Science Foundation of China [grant number 62141201], Natural Science Foundation of Chongqing [grant number CSTB2022NSCQ-MSX1672].


## References


Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022a. Discrete opinion tree induction for aspect-based sentiment analysis. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2051–2064.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022b. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web Conference 2022*, pages 2778–2788.

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*.

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)*, pages 49–54.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, pages 6894–6910. Association for Computational Linguistics (ACL).

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-factr: Multilingual factual knowledge retrieval from pretrained language models. *arXiv preprint arXiv:2010.06189*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059.

Pan Li, Ping Li, and Xiao Xiao. 2023. Aspect-pair supervised contrastive learning for aspect-based sentiment analysis. *Knowledge-Based Systems*, 274:110648.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. 2021. Dual graph convolutional networks for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6319–6329.

Bin Liang, Wangda Luo, Xiang Li, Lin Gui, Min Yang, Xiaoqi Yu, and Ruifeng Xu. 2021. Enhancing aspect-based sentiment analysis with supervised contrastive learning. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 3242–3247.

Rui Liu, Jiahao Cao, Nannan Sun, and Lei Jiang. 2022. Aspect feature distillation and enhancement network for aspect-based sentiment analysis. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1577–1587.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

Fukun Ma, Xuming Hu, Aiwei Liu, Yawen Yang, Shuang Li, Philip S. Yu, and Lijie Wen. 2023. AMR-based network for aspect-based sentiment analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 322–337, Toronto, Canada. Association for Computational Linguistics.

Xiaoyong Mei, Yougen Zhou, Chenjing Zhu, Mengting Wu, Ming Li, and Shirui Pan. 2023. A disentangled linguistic graph model for explainable aspect-based sentiment analysis. *Knowledge-Based Systems*, 260:110150.

Khalil Mrini, Franck Dernoncourt, Trung Bui, Walter Chang, and Ndapa Nakashole. 2019. Rethinking self-attention: An interpretable selfattentive encoder-decoder parser. *arXiv preprint arXiv:1911.03875*.

Maria Pontiki, Haris Papageorgiou, Dimitrios Galanis, Ion Androutsopoulos, John Pavlopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. *SemEval 2014*, page 27.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Targeted sentiment classification with attentional encoder network. In *Artificial Neural Networks and Machine Learning–ICANN 2019: Text and Time Series: 28th International Conference on Artificial Neural Networks, Munich, Germany, September 17–19, 2019, Proceedings, Part IV 28*, pages 93–103. Springer.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 5679–5688.

Duyu Tang, Bing Qin, Xiaocheng Feng, Ting Liu, et al. 2015. Target-dependent sentiment classification with long short term memory. *arXiv preprint arXiv:1512.01100*.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6578–6588.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 2910–2922.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Bing Wang, Liang Ding, Qihuang Zhong, Ximing Li, and Dacheng Tao. 2022. A contrastive cross-channel data augmentation framework for aspect-based sentiment analysis. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6691–6704, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. *arXiv e-prints*, pages arXiv–2004.

Shijie Wang, Jianlong Chang, Zhihui Wang, Haojie Li, Wanli Ouyang, and Qi Tian. 2023. Fine-grained retrieval prompt tuning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 2644–2652.

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.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards generative aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 504–510.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022a. A survey on aspect-based sentiment analysis: tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022b. Ssegcn: Syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4916–4925.

# A Template Analysis

Table 4: The prompt templates and labels we constructed manually in our experiments, $< a >$ denotes the aspect. We concatenate them with the original sentence to form the input and labeling samples

| Index | **Template** | **Label words** |
|---|---|---|
| $t_0$ | The sentiment of $< a >$ is [MASK] | P:positive, N:negative, O:neutral |
| $t_1$ | The sentiment of $< a >$ is [MASK] | P:nice, N:bad, O:none |
| $t_2$ | The sentiment of $< a >$ is [MASK] | P:negative, N:neutral, O:positive |
| $t_3$ | The $< a >$ is [MASK] | P:good, N:terrible, O:ok |
| $t_4$ | The $< a >$ is [MASK] | P:positive, N:negative, O:neutral |
| $t_5$ | How about $< a >$ ? it is [MASK] | P:good, N:terrible, O:ok |
| $t_6$ | What do you think of the $< a >$ ? it is [MASK] | P:good, N:terrible, O:ok |

Table 5: Experimental results on the three datasets with different templates. The best results are in **bold**.

| **Template** | **Laptop** | | **Restaurant** | | **Twitter** | |
|---|---|---|---|---|---|---|
| | **ACC** | **Macro-F1** | **ACC** | **Macro-F1** | **ACC** | **Macro-F1** |
| $t_0$ | **82.75** | **79.61** | **87.85** | **82.41** | **79.32** | **78.44** |
| $t_1$ | 81.80 | 78.97 | 87.04 | 81.16 | 76.96 | 75.74 |
| $t_2$ | 81.48 | 77.86 | 86.15 | 79.29 | 76.96 | 75.83 |
| $t_3$ | 81.33 | 78.41 | 86.24 | 80.15 | 74.89 | 73.33 |
| $t_4$ | 81.65 | 78.72 | 86.15 | 80.57 | 75.63 | 74.46 |
| $t_5$ | 82.28 | 78.86 | 85.43 | 79.23 | 75.18 | 74.09 |
| $t_6$ | 81.80 | 78.60 | 86.51 | 80.38 | 76.96 | 75.86 |

We investigated the impact of different templates on model performance, and the results are shown in Table 5. The table presents the results obtained using the prompts constructed in Table 4. The following observations can be made: (1) Different prompts and label words have a significant influence on the model's performance. (2) When using the same prompt, different label words yield varying results. However, compared to the performance differences resulting from using different label words and the same prompt, the differences are relatively smaller. This indicates that the selection of the template plays a more crucial role. (3) When the semantic meaning of the label word is completely opposite to the sentiment label, there is a certain decrease in model performance. However, since our model also extracts other features of the sentence, the extent of performance degradation is limited.

# B Complexity Analysis

Table 6: Comparison of model's complexity. We compared the results of models among all baseline models that provided information on the quantity of parameters and training time, * indicates the results from our implementations.

| **Models** | **Params(M)** | **Training time(s/epoch)** | **Parser** |
|---|---|---|---|
| DualGCN* | 112 | 35 | LAL-Parser |
| SSEGCN* | 110 | 32 | LAL-Parser |
| APARN | 130 | 480 | Spring, LEAMR |
| Our $C^3$LPGCN | 122 | 202 | LAL-Parser |

We also analyzed the complexity of our model, primarily comparing parameters such as the quantity of parameters, training time, and the parsers used. The results are presented in Table 6. It can be observed that our model has a roughly similar number of parameters compared to other models. However, due to constructing three samples for each input for contrastive and cooperative learning, our model's training time is significantly longer compared to DualGCN and SSEGCN. Nevertheless, it can be seen that compared to the best-performing model in the comparison, APARN, our method has advantages in both training time and the number of parsers used.