Structure-aware Generation Model for Cross-Domain Aspect-based Sentiment Classification

Shichen Li, Zhongqing Wang*, Yanzhi Xu, Guodong Zhou
Natural Language Processing Lab, Soochow University, Suzhou, China
scli_21@outlook.com, {wangzq,gdzhou}@suda.edu.cn, yzxuxyz@stu.suda.edu.cn

Abstract

Employing pre-trained generation models for cross-domain aspect-based sentiment classification has recently led to large improvements. However, they ignore the importance of syntactic structures, which have shown appealing effectiveness in classification based models. Different from previous studies, efficiently encoding the syntactic structure in generation model is challenging because such models are pretrained on natural language, and modeling structured data may lead to catastrophic forgetting of distributional knowledge. In this study, we propose a novel structure-aware generation model to tackle this challenge. In particular, a prompt-driven strategy is designed to bridge the gap between different domains, by capturing implicit syntactic information from the input and output sides. Furthermore, the syntactic structure is explicitly encoded into the structure-aware generation model, which can effectively learn domain-irrelevant features based on syntactic pivot features. Empirical results demonstrate the effectiveness of the proposed structure-aware generation model over several strong baselines. The results also indicate the proposed model is capable of leveraging the input syntactic structure into the generation model.

Keywords: Opinion Mining / Sentiment Analysis, Social Media Processing, Text Analytics

1. Introduction

Aspect-based sentiment classification (Hu and Liu, 2004) aims to determine the sentiment polarity of the aspects in a sentence. Consequently, it has aroused much research attention in recent years (Tang et al., 2016; Tay et al., 2018; Zhang et al., 2022). However, a large number of review domains make it intractable to manually annotate enough data in each domain for training domain-specific models. Thus developing automatic cross-domain methods is imperative in this area.

Recent efforts on cross-domain sentiment classification can be separated into two categories: features-based approaches and discriminator-based approaches. Feature-based approaches (Blitzer et al., 2007; Yu and Jiang, 2016; Ziser and Reichart, 2019) utilize a key intuition that domain-specific features could be aligned with the help of domain invariant features. Discriminator-based approaches (He et al., 2018; Du et al., 2020; Xue et al., 2020) aim to determine the diversity between domains and predict the polarity of instances holistically.

More recently, pre-trained generation models learn the cross-domain representations with mixed classification and masked language model (Karouzos et al., 2021; Li et al., 2022a; Ben-David et al., 2022). For example, Zhou et al. (2020) learned domain-invariant sentiment knowledge in the pre-training phase, and Karouzos et al. (2021) employed mixed classification and masked language model to fine-tune the pre-trained model. However, these models represent

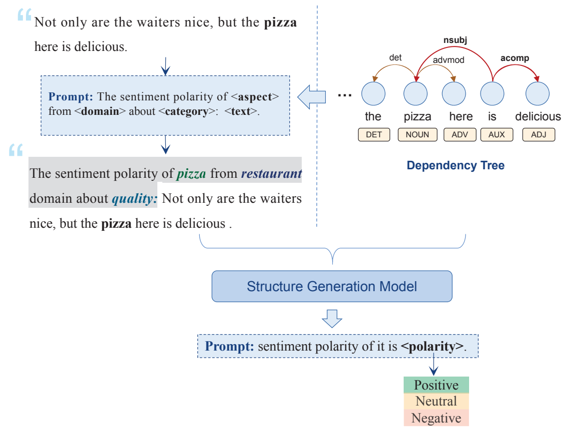

Figure 1: An example of the proposed structure-aware generation model with prompt design and syntax tree.

the sentence as a word sequence and neglect the syntactic relations between words. These syntactic relations are especially important for identifying aspect terms and opinion terms, which are also domain-invariant within the same language. They can be used as *pivot* information to bridge the gap between different domains.

Therefore, we attempt to adopt syntactic relations into a generation model to learn the knowledge from source and target domains. A straightforward method is to transform the structured input into a sequence, which can be directly fed into the generation model. However, the above method suffers from two salient limitations. First, linearized syntactic structures are different in nature from natural language. As a result, knowledge among differ-

* Corresponding author

LREC-COLING 2024, pages 15373–15383

ent domains intuitively cannot be fully transferred. Second, a linearized representation weakens structural information in the original graphs by diluting the explicit connectivity information, and the generation model must infer how edge connections are specified in the sequence.

In this study, we propose a novel *structure-aware generation model* to address the above challenges. The basic idea is shown in Figure 1, we first employ a prompt-driven strategy to implicitly integrate the syntactic structure with domain and category instructions. Specifically, domain instruction is used to learn domain-specific knowledge, while category instruction is employed to bridge the gap between different domains. Furthermore, we explicitly encode the syntactic structure into the pre-trained generative model without contaminating its original distributional knowledge. The main idea is to add layer-wise modules, which extract information from the pre-trained layers and make use of it in syntactic structure encoding. Therefore, deep integration of textual and syntactic knowledge can be achieved.

Experimental results show that our model outperforms competitive models, achieving new state-of-the-art results on several benchmarks. Deep analysis indicates that the proposed model is capable of leveraging the input syntactic structure into the generation model.

2. Related Works

In this study, we introduce two related topics of this study: aspect-based sentiment classification and cross-domain sentiment classification.

2.1. Aspect-based Sentiment Classification

Recent advances in aspect-based sentiment classification focus on developing various types of neural models, including LSTM (Tang et al., 2016; Ma et al., 2017), convolutional neural networks (Huang and Carley, 2018; Xue and Li, 2018) and pre-trained language models (Sun et al., 2019a; Xu et al., 2019; Jiang et al., 2020).

Some other efforts try to include syntactic information using graph neural networks (Tay et al., 2018; Sun et al., 2019b; Huang and Carley, 2019; Tang et al., 2020). For example, Zhang and Qian (2020) and Li et al. (2021) used graph convolutional networks to learn node representations from a dependency tree and used them together with other features for sentiment classification. For a similar purpose, Huang and Carley (2019) and Wang et al. (2020) used graph attention networks to explicitly establish the dependency relationships between words. Besides, Liang et al. (2022) and Zhang et al. (2022) combined the syntax information of constituent tree and dependency tree with graph neural networks to model aspect-based sentiment classification tasks.

2.2. Cross-Domain Sentiment Classification

Cross-domain sentiment classification has been a long standing attractive research topic due to its real applications where labeled data is only available in a source domain. Previous studies can be separated into two categories: features-based approaches (Blitzer et al., 2007; Yu and Jiang, 2016; Ziser and Reichart, 2019) and discriminator-based approaches (He et al., 2018; Qu et al., 2019; Du et al., 2020; Xue et al., 2020; Li et al., 2022b).

More recently, researchers focused on employing pre-trained language models in cross-domain classification scenarios. For example, Zhou et al. (2020) learned domain-invariant sentiment knowledge in the pre-training phase. Karouzos et al. (2021) employed mixed classification and masked language model loss to fine-tune the pre-trained model, it thus can adapt to the target domain distribution in a robust and sample efficient manner.

Most of the previous studies are for document-level sentiment classification, and only a few studies focus on cross-domain aspect-based sentiment classification. For example, Gong et al. (2020) proposed an end-to-end framework to jointly perform syntactic-based and domain-based adaptation. Zhou et al. (2021) integrated pseudo-label based semi-supervised learning and adversarial training into a unified network. More recently, Li et al. (2022a) and Yu et al. (2021) employed generative models for data augmentation to accomplish aspect-based domain adaptation. Ben-David et al. (2022) used an example-based prompt learning algorithm to alleviate domain discrepancy between different domains.

Different from previous studies, we propose a novel paradigm to transform the cross-domain aspect-based sentiment classification task into a structure-aware generation model. In particular, we address the importance of syntactic structure and propose a structure-aware generation model to implicitly and explicitly encode the syntactic structure into the pre-trained generation model. To the best of our knowledge, this is the first attempt to leverage the pre-trained generation model and syntactic structure for cross-domain sentiment classification.

3. Structure-aware Generation Model

As shown in Figure 2, we propose a structure-aware generation model to generate sentiment polarity from input review text and syntactic structure. Our key idea is that the proposed model should be ca-

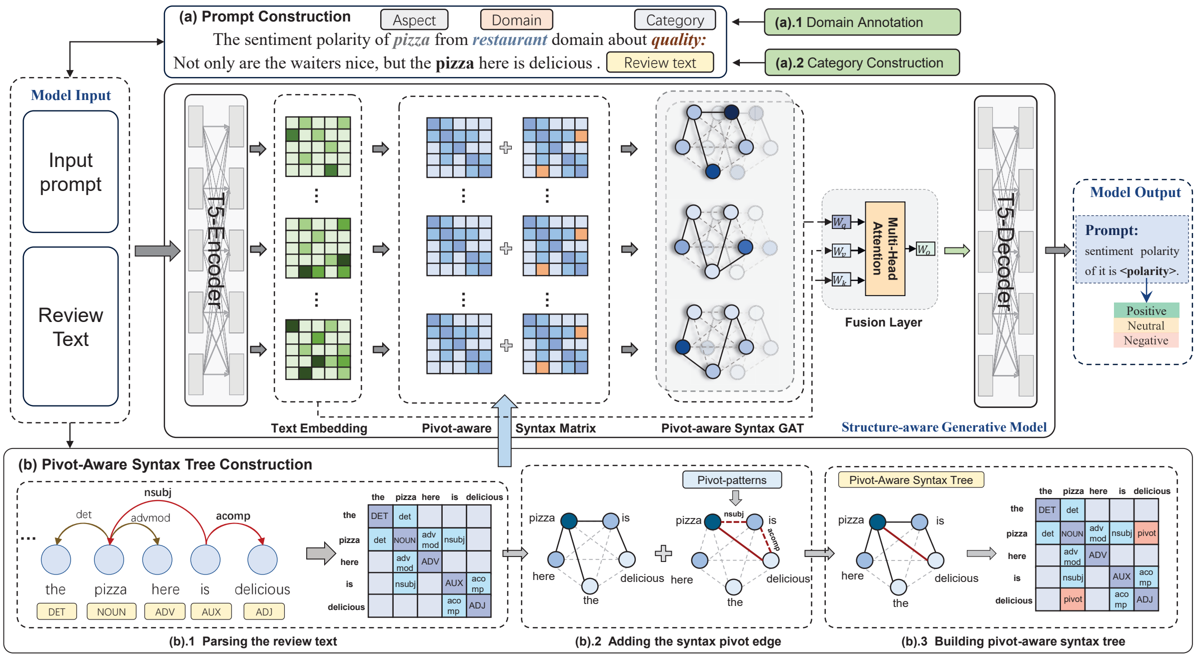

Figure 2: Overview of the proposed model.

pable of leveraging the input syntactic structure into the generation model. For this purpose, the proposed model first employs a prompt-driven strategy to implicitly integrate the syntactic structure with domain and category instructions. Then, it constructs the syntactic structure from the review text and consists of a syntactic structure encoding layer to explicitly capture syntactic structure. Furthermore, it uses a fusion layer to integrate text representation and syntactic structure and generates a natural language sentence with sentiment polarity based on the fused representation in the decoding stage. In the following content of this section, we will discuss these components in more detail.

3.1. Prompt Design

Given a review text x with its domain d and aspect word a, we propose a prompt-driven strategy to implicitly integrate the syntactic structure, and generate the input prompt T_I with domain instruction d_I and category instruction c_I as shown in Figure 2(a). In addition, we utilize a template for output T_O with sentiment polarity y.

Input Prompt

The template of input is defined as "The sentiment polarity of <aspect> from <domain> about <category>: <text>". The first three slots are aspect term a, domain instruction d_I, and category instruction c_I, and the last slot is the review text x.

Domain Instruction is a natural language sequence d_I describing the domain of the review text, and it is used to assign the domain-specific knowledge for the input prompt.

Category Instruction is a natural language sequence c_I describing the pre-defined category from the review text and syntactic structure. In this study, there are six pre-defined categories: "Quality", "Service", "Ambience", "Price", "Performance" and "Miscellaneous". We refer to the definition in SemEval-14 (Pontiki et al., 2014) to define these categories. These categories are generalized and can be used to bridge the gap between different domains. For example, product quality is frequently discussed across all domains, and price also appears in most domains. In particular, we employ an unsupervised method (Gao et al., 2021) with syntactic structure to measure the semantic similarity between the review text and categories, and then we assign the most similar category to the corresponding review text.

Output Prompt

The template of output is "sentiment polarity of it is <polarity>", which contains a slot "<polarity>" to reflect how polarity is in the review text. Therefore, the rich label semantics is naturally fused with the rich knowledge of the pre-trained models in the form of natural sentences, rather than directly treating the desired sentiment polarity as the generation target.

3.2. Text Encoder

Given an input template $X = \{x_1, ..., x_n\}$, consisting of the review text, the domain, and category instruction, we employ a multi-layer transformer encoder to compute the hidden vector representation $H_T = \{h_1, ..., h_n\}$:

$$H_T = \text{Encoder}(\{x_1, ..., x_n\}) \tag{1}$$

where each layer of Encoder is a transformer block with the multi-head attention mechanism.

3.3. Pivot-Aware Syntax Tree Construction

The syntactic structure has been shown effective for aspect-based sentiment classification (Liang et al., 2022; Zhang et al., 2022). In addition, the syntactic relations between aspect and opinion terms have common and domain transferable linguistic characteristics (Klein et al., 2022). It suggests that the syntactic relations can be used as informative pivot features for cross-domain aspect-based sentiment classification.

As shown in Figure 2(b), we construct a pivot-aware syntax tree to capture domain invariant relations between aspect and opinion terms. we first capture the most frequent syntactic relations between aspect and opinion terms on the dependency tree. We then select the top-k common patterns as pivot features from these syntactic relations. Subsequently, given a parsed review text, we utilize the obtained pivot path patterns to construct the pivot-aware syntax tree.

Path Patterns Extraction

Due to the correspondences between aspect terms and opinion terms are not annotated in the benchmark dataset, we take the ACOS dataset(Cai et al., 2021) to extract path patterns.

Given parsed review text, we capture the shortest path between aspect and opinion terms as path patterns, where the pattern is the ordered list of the dependency relation labels occurring throughout the shortest path.

Following a preliminary analysis, we chose 5659 sentences with 1191 path patterns from ACOS dataset in both laptop and restaurant domains. We find that 69% sentences are covered by top-10(top 1%) path patters and top 10% path patterns can cover 76% sentences. Therefore, we select the top-10 common patterns as pivot path patterns to extract more common and domain transferable linguistic characteristics.

Pivot-Aware Syntax Tree Construction

We consider the dependency tree as an undirected graph $\mathcal{G}_{dep}$ as shown in Figure 2(b).1. Let $\mathcal{N}$ and $\mathcal{R}$ denote the set of nodes and syntactic relations on $\mathcal{G}$, respectively. We can extract dependency triplets from $\mathcal{G}_{dep}$, where a dependency triplet (x_i, r, x_j) denotes that there exists an edge with the syntactic relation $r \in \mathcal{R}$ to link the node $x_i \in \mathcal{N}$ to node $x_j \in \mathcal{N}$. Particularly, the syntactic dependency matrix can be defined as follows,

$$r_{dep}(x_i, x_j) = \begin{cases} Rel(x_i, x_j) & if\ dep(x_i, x_j)_{i \neq j} \\ Pos(x_i, x_j) & if\ i = j \\ 0 & else \end{cases} \tag{2}$$

where $Rel(x_i, x_j)$ belongs to the set of cross-linguistic dependency relations denoted as $\mathcal{R}_{dep}$, which is defined based on universal dependencies as implemented in Spacy. Similarly, $Pos(x_i, x_j)$ belongs to the set of part-of-speech tags denoted as $\mathcal{R}_{pos}$, defined based on POS tags from Spacy. Subsequently, given x_i and x_j belonging to the set of nodes $\mathcal{N}$, and $r_{dep}(x_i, x_j)$, we can derive (x_i, r_{ij}, x_j) belonging to the dependency graph $\mathcal{G}_{dep}$.

Given a text and its aspect term, we capture the shortest paths between the aspect terms and the other tokens in the dependency graph $\mathcal{G}_{dep}$ to get a pattern set $\mathcal{S}_s$, where the patterns are the ordered list of the dependency relation labels occurring throughout the shortest path. Let the extracted pivot path patterns as $\mathcal{S}_p$, we can easily get $\mathcal{S}_p \cap \mathcal{S}_s$ as $\mathcal{S}_t$. Subsequently, we directly link the aspect terms with destination tokens guided by the paths in $\mathcal{S}_t$ as shown in Figure 2(b).2. Specifically, we add dependency triplets (x_a, r_{at}, x_t) to $\mathcal{G}_{dep}$, where x_a is aspect terms, r_{at} is a special edge type meaning pivot, x_t is destination token according to the path in $\mathcal{S}_t$ as shown in Figure 2(b).3.

Ultimately, we construct a pivot aware syntax tree that combines the syntax dependency tree and domain invariant linguistic structure of aspect and opinion terms.

3.4. Syntax Tree Encoder

After we construct the pivot-aware syntax tree G, we use Graph Attention Network (GAT) (Velickovic et al., 2018) to model over it.

In a M-layer GAT, the input of j-th layer is a set of node features $NF^j = \{f_1, f_2, ..., f_N\}$, together with an adjacency matrix A. In this study, the node features are denoted as the words from review text. In addition, the adjacency matrix A is used to represent the syntax tree G. If there is a relation between word i and word j, then A_{ij} will be assigned a value of 1. The output of j-th layer is a new set of node features, $NF^{(j+1)} = \{f'_1, f'_2, ..., f'_N\}$. A GAT operation with K independent attention head can be written as,

$$f'_i = \|_{k=1}^{K} \sigma(\sum_{j \in N_i} \alpha_{ij}^k W^k f_j) \tag{3}$$

where $\|$ denotes concatenation operation, σ is a nonlinear activation function, N_i is the neighbourhood of node i in the graph, α_{ij}^k are the attention coefficients.

At the last layer, averaging will be adopted, and the dimension of final output features is $H_G = \{f_1^{final}, ..., f_n^{final}\}$.

$$f_i^{final} = \sigma(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in N_i}\alpha_{ij}^k W^k f_j) \quad (4)$$

3.5. Fusion with Text and Syntax Representation

We employ a fusion layer to incorporate syntax representation with text representation. In this way, the syntactic structure substantially impacts the node representations, better encoding the input structure without impacting the knowledge learned during pre-training. This can lead to more efficient and better generation results.

In particular, the input of the fusion layer is the text representation H_T and the output of syntax tree encoder H_G. The fusion layer is based on the self-attention (Vaswani et al., 2017) and we compute the attention weight by the following formula:

$$H = \mathrm{softmax}(\frac{H_G H_T^T}{\sqrt{d_m}})H_T \quad (5)$$

where d_m is the dimension of H_T. Based on the fusion layer, we obtain a matrix H, which is a new representation integrating the knowledge from both text and syntax representation.

3.6. Decoding

The decoder predicts the output token-by-token with the sequential input tokens' hidden vectors. At the i-th step of generation, the self-attention decoder predicts the i-th token t_i in the linearized form and decoder state h as:

$$t_i, h_i^d = \mathrm{Decoder}([H; h_1^d, ..., h_{i-1}^d], t_{i-1}) \quad (6)$$

where each layer of $\mathrm{Decoder}$ is a transformer block that contains self-attention with decoder state h_i^d and cross-attention with encoder state H.

The conditional probability of the whole output sequence $p(T|X)$ is progressively combined by the probability of each step $p(t_i|t_{<i}, X)$:

$$p(T|X) = \prod_{i}^{m} p(t_i|t_{<i}, X) \quad (7)$$

where $t_{<i} = \{t_1...t_{i-1}\}$, and $p(t_i|t_{<i}, X)$ is the probability over target vocabulary V normalized by $\mathrm{softmax}$.

Domains	Reviews	Training	Testing
Device	2,085	1,394	691
Laptop	2,928	2,297	631
Restaurant	6,536	4,284	2,252
Service	2,726	1,840	886

Table 1: Distribution of reviews across different domains.

3.7. Objective Functions and Training

In this subsection, we show the objective function and training process of the proposed model.

The goal is to maximize the objective text T probability given the review text X. Therefore, we optimize the negative log-likelihood loss function:

$$\mathcal{L} = -\frac{1}{|\tau|}\sum_{(X,T)\in\tau}\log p(T|X;\theta) \quad (8)$$

where θ is the model parameters, and (X, T) is a (*input,output*) pair in training set τ, then

$$\begin{aligned}&\log p(T|X;\theta) = \\ &= \sum_{i=1}^{m}\log p(t_i|t_1, t_2, ...t_{i-1}, X; \theta)\end{aligned} \quad (9)$$

where $p(t_i|t_1, t_2, ..., t_{i-1}, X; \theta)$ is calculated by the decoder.

4. Experiments

In this section, we first introduce the datasets used for evaluation and the baseline methods employed for comparison. We then report the experimental results conducted from different perspectives and analyze the effectiveness of the proposed model with different factors.

4.1. Data and Setting

There are four domains in the dataset: Restaurant (R) is a combination of the restaurant reviews from SemEval 2014/2015/2016 (Pontiki et al., 2014, 2015, 2016); Laptop (L) is from SemEval 2014 (Pontiki et al., 2014); Device (D) consists of all the digital device reviews collected by Toprak et al. (2010); Service (S) contains reviews from web services introduced by Hu and Liu (2004). The distribution of reviews in these domains can be found in Table 1.

We use T5-base[1] as our backbone in the proposed model. We select the best model by early stopping using the accuracy results on the validation dataset (the testing data in the source domain is used as the validation data). We conduct all experiments on a GeForce RTX 3090 GPU with a batch size of 24. The learning rate of the T5 model

[1] https://huggingface.co/t5-base

	BERT	BART	T5	ChatGPT	LLaMA	ADSPT	PADA	ACSC	Ours
D → L	67.83	69.10	70.68	**79.23**	70.40	70.05	<u>71.36</u>	68.94	71.32
D → R	80.37	82.33	81.79	**86.32**	81.86	82.86	83.28	82.19	<u>84.28</u>
D → S	85.21	86.46	87.58	83.63	<u>87.91</u>	87.13	87.23	86.57	**89.28**
L → D	89.58	89.29	<u>92.19</u>	90.30	91.91	92.04	90.40	91.17	**94.21**
L → R	81.13	82.73	79.53	**86.32**	81.04	84.81	84.66	83.57	<u>86.15</u>
L → S	81.94	86.00	85.33	83.63	<u>86.97</u>	83.30	85.20	84.54	**87.02**
R → D	88.28	90.16	<u>93.92</u>	90.30	92.03	91.17	91.10	89.44	**94.65**
R → L	77.81	77.50	78.29	<u>79.23</u>	74.81	78.45	78.34	75.75	**81.30**
R → S	84.54	86.68	87.02	83.63	86.57	83.86	<u>87.79</u>	85.97	**88.49**
S → D	90.74	91.03	<u>93.49</u>	90.30	92.76	91.46	93.04	91.75	**94.79**
S → L	69.73	70.84	70.36	**79.23**	73.18	71.47	69.78	71.16	<u>75.91</u>
S → R	79.75	81.35	81.17	**86.32**	81.43	82.37	82.93	82.28	<u>83.39</u>
Average	81.41	82.79	83.45	<u>84.87</u>	83.40	83.25	83.76	82.94	**85.90**

Table 2: Comparison with baselines.The best results are denoted with **bold** and the second best results are denoted <u>underline</u>

is 2e-4 and the learning rate of the graph attention model is 4e-5. The model parameters are optimized by Adam (Kingma and Ba, 2015) optimizer. The experimental results are obtained by averaging ten runs with random initialization. Accuracy is used as the evaluation metric.

4.2. Main Results

In this subsection, we compare the proposed model with several strong baselines:

- **BERT** is a basic model which simply employs BERT (Devlin et al., 2019) for cross-domain aspect-based sentiment classification.
- **BART**[2] **& T5** are two popular pre-trained text generation models (Lewis et al., 2020; Raffel et al., 2020), we adopt them directly for cross-domain aspect-based sentiment classification.
- **ChatGPT** is a sibling model to InstructGPT (Ouyang et al., 2022), which is trained to follow an instruction in a prompt and provide a detailed response[3]. We utilized it to generate the aspect polarity of the input review using a zero-shot setting with the instruction "*Sentence:{sentence} What is the sentiment polarity of the aspect {aspect} in this sentence.*".
- **LLaMA** (Touvron et al., 2023) is a collection of foundation language models, these models are trained on trillions of tokens and have shown that it is possible to train state-of-the-art models using publicly available datasets exclusively. We use Alpaca-LoRA[4] to fine-tune LLaMA-7B for cross-domain aspect-based sentiment classification with Alpaca instruction format.
- **ADSPT** (Wu and Shi, 2022) adopts soft prompts to learn different vectors from different domains, and employs a domain adversarial training strategy to learn domain-invariant representations between different domains.
- **PADA** (Ben-David et al., 2022) employs prompting mechanism and the encoders of T5 (Raffel et al., 2020) for aspect-based sentiment classification.
- **ACSC** (Liu et al., 2021) employs the pre-trained generation model for aspect-based sentiment classification directly.

As shown in Table 2, these baselines can be roughly separated into three categories: classification-based models (i.e., BERT, ADSPT, PADA), encoder-decoder generation models (i.e., BART, T5, ACSC) and decoder-only large language models (i.e., ChatGPT, LLaMA). Among these baselines, the generation models and large language models cannot outperform classification-based models significantly, it may be due to that they simply employ the text generation model for cross-domain aspect-based sentiment classification directly, and do not use any suitable domain adaptation strategy (e.g., prompt, graph-based model) to integrate syntactic structure.

In contrast, our proposed model outperforms all the baselines significantly ($p < 0.05$). It even outperforms ChatGPT and LLaMA, whose parameters are much larger. This indicates that the generation model using syntactic information can achieve better performance. The result also shows the effectiveness of the syntactic structure and proposed

[2]https://huggingface.co/facebook/bart-base

[3]https://openai.com/chatgpt

[4]https://github.com/tloen/alpaca-lora

Method	Accuracy
Ours	85.90
-Prompt	85.21
-Tree	84.40
-Prompt -Tree	83.45

Table 3: Impact of different factors.

Method	Accuracy
Ours	85.90
-Input Prompt	85.23
-Domain Instruction	85.52
-Category Instruction	85.68
-Output Prompt	85.78

Table 4: Effect of prompt design.

structure-aware generation model, which allows the generation model to better encode the input syntactic structure without impacting the knowledge learned during pre-training.

4.3. Impact of Different Factors

We then employ ablation experiments to analyze the impact of different factors of the proposed model in Table 3.

If we remove the prompt design strategy (-Prompt) of the proposed model, the performance drops to 85.21%. It indicates that the prompt text with domain and category instructions is very important for integrating syntactic structure and capturing domain-specific and cross-domain knowledge. In addition, without the enhancement of the pivot-aware syntax tree (-Tree), the performance drops to 84.40%. This evidence shows that the proposed structure-aware generation model is very helpful for explicitly encoding the syntactic structure and bridging the gap between different domains.

Furthermore, if we remove both the prompt-driven strategy and syntax tree (-Prompt-Tree), the performance is much lower than the proposed model. These observations suggest that both prompt-driven strategy and structure-aware generation model are very important for integrating syntactic structure.

5. Analysis and Discussion

In this section, we give some analysis and discussion to show the importance of different factors in the proposed model.

5.1. Effect of Prompt Design

Since prompt design strategies are employed to implicitly learn the syntactic structure, we first analyze their effects.

From the results in Table 4, we find that all the prompt design strategies are beneficial to capture the cross-domain knowledge in both the input and output sides. If we remove one of them, the performance will be lower than our proposed model. Furthermore, these results also highlight the importance of both domain and category instructions. These instructions can capture implicit syntactic information and enhance the input prompt to better generalize to unknown target domains with their domain specific and cross-domain knowledge.

Method	Accuracy
T5	83.45
Ours (Original)	85.49
Ours (Pivot-Aware)	85.90
Linearization (Original)	83.81
Linearization (Pivot-Aware)	83.86

Table 5: Influence of syntax tree.

5.2. Influence of Syntax Tree

We then analyze the influence of the syntax tree in Table 5. In particular, *Linearization* uses BFS-based traversal algorithm (Xu et al., 2020) to linearize the syntax tree, and then employs T5 to integrate the linearized string and review text for cross-domain aspect-based sentiment classification. *Original* and *Pivot-Aware* integrate either the original syntax tree or the proposed pivot-aware syntax tree for the proposed structure-aware generation model (Ours) and linearization model.

The first observation is that all the models with syntax tree perform better than T5, it shows that the syntax tree is very helpful for generation models to capture the syntactic and cross-domain knowledge. In addition, the pivot-aware syntax tree always outperforms the original syntax tree, it indicates that the pivot-aware syntax tree is more effective for capturing aspect-based domain irrelevant knowledge. Furthermore, our proposed structure-aware generation model achieves better performance than the linearization model. It demonstrates that the proposed structure-aware generation model with graph attention mechanism is more helpful to capture the structure of the syntax tree.

5.3. Analysis of Similarity between Domains

The performance of cross-domain models is always influenced by the similarity between the source and target domain. Therefore, we analyze the improvement of the Jaccard similarity score (Ioffe, 2010) on different domain pairs with the enhancement of

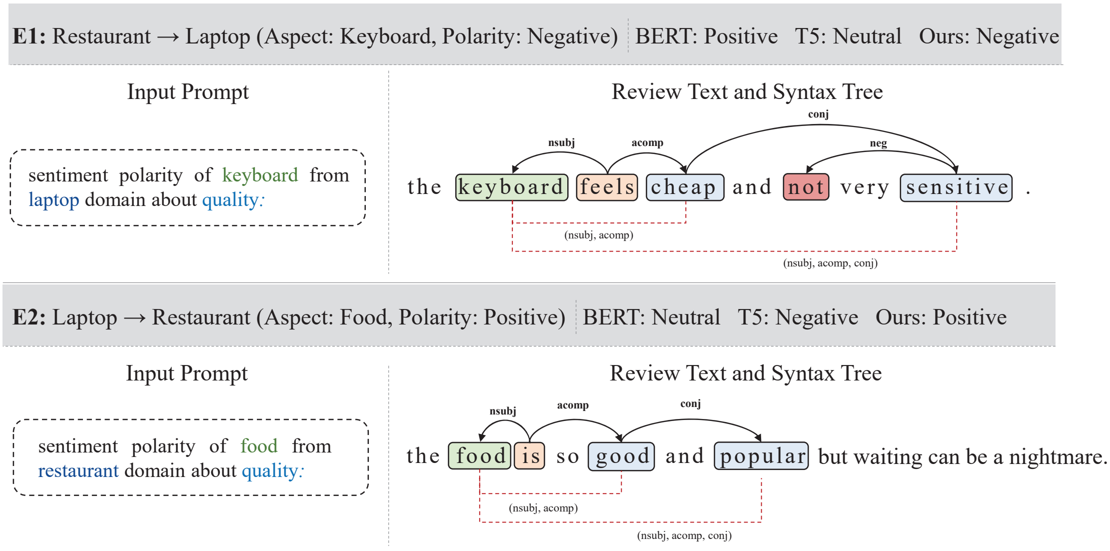

Figure 3: Examples of case study.

Pairs	Text	Prompt	Tree	All
D ↔ L	0.275	0.365	0.406	0.459
D ↔ R	0.154	0.232	0.348	0.371
D ↔ S	0.193	0.282	0.351	0.400
L ↔ R	0.151	0.235	0.342	0.377
L ↔ S	0.208	0.295	0.377	0.422
R ↔ S	0.166	0.256	0.351	0.389
Average	0.191	0.278	0.363	0.403

Table 6: Results of similarity between source and target domain.

the proposed model. The Jaccard similarity score reflects the similarity between domain pairs.

As shown in Table 6, *Text* only employs original review text to calculate the similarity between the source and target domain; *Prompt* and *Tree* employ the prompt text and linearized pivot-aware syntax tree to calculate the similarity, respectively; *All* integrates both prompt text and linearized tree to calculate the similarity. From the results, we find that the similarity of prompt text and pivot-aware syntax tree are both higher than the original review text. In addition, the similarity achieves the highest when we integrate both prompt text and syntax tree to calculate the similarity. The results show that our proposed model with prompt design and syntax tree is truly useful for bridging the gap between different domains. Additionally, the results also indicate that the proposed model with syntactic structure can effectively alleviate domain discrepancy and have a powerful ability in cross-domain aspect-based sentiment classification.

5.4. Case Study

To further investigate the meaningfulness of our proposed model, we choose two examples from the testing data in Figure 3. In particular, we choose BERT (Devlin et al., 2019) and T5 (Raffel et al., 2020) as the baseline methods.

The first example is very easy for the baselines to give a wrong answer, since "cheap" is near the aspect term "keyboard", and it always expresses positive sentiment polarity in the source domain. However, the true meaning of the review is about the quality of the keyboard rather than price, while the category "quality" can be detected by the proposed model through the syntax tree. Therefore, with the guidance of the prompt text with category instruction and syntax tree, the proposed model can find "cheap" and "not sensitive" express the negative opinion to the aspect term "keyboard", and then give the correct answer.

Since the overall sentiment of the second review is ambiguous, it also would let the baselines give wrong answers. For example, the sentiment polarity of the aspect term is misclassified by T5 as negative due to the unrelated opinion word "nightmare". In addition, BERT may concentrate more on the sentiment polarity of the whole sentence rather than the aspect term. Meanwhile, based on the prompt text and the syntax tree, it is easy for the proposed structure-aware generation model to find that the review expresses a positive opinion toward the aspect "food".

The results indicate that the proposed model can capture cross-domain knowledge and implicit structures in different domains, and is very effective for cross-domain aspect-based sentiment classification.

6. Conclusion

In this study, we propose a novel paradigm to transform the cross-domain aspect-based sentiment classification task into a natural language generation task, using natural language sentences to represent the output. In addition, we address the importance of syntactic structure and propose a novel structure-aware generation model to explicitly and implicitly encode the syntactic structure into the pre-trained generation model. Empirical studies show the effectiveness of our proposed model over several strong baselines. The results also indicate that the proposed structure-aware generation model can effectively capture syntactic structure.

7. Acknowledgement

We would like to thank the anonymous reviewers for their insightful and valuable comments. This work is supported by the National Natural Science Foundation of China (No. 62076175, No. 61976146).

8. References

Eyal Ben-David, Nadav Oved, and Roi Reichart. 2022. PADA: example-based prompt learning for on-the-fly adaptation to unseen domains. *Trans. Assoc. Comput. Linguistics*.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of ACL*.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, and Jianxin Liao. 2020. Adversarial and domain-aware BERT for cross-domain sentiment analysis. In *Proceedings of ACL*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of EMNLP*.

Chenggong Gong, Jianfei Yu, and Rui Xia. 2020. Unified feature and instance based domain adaptation for aspect-based sentiment analysis. In *Proceedings of EMNLP*.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018. Adaptive semi-supervised learning for cross-domain sentiment classification. In *Proceedings of EMNLP*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of SIGKDD*.

Binxuan Huang and Kathleen M. Carley. 2018. Parameterized convolutional neural networks for aspect level sentiment classification. In *Proceedings of EMNLP*.

Binxuan Huang and Kathleen M. Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. In *Proceedings of EMNLP-IJCNLP*.

Sergey Ioffe. 2010. Improved consistent sampling, weighted minhash and L1 sketching. In *Proceedings of ICDM*.

Bin Jiang, Jing Hou, Wanyue Zhou, Chao Yang, Shihan Wang, and Liang Pang. 2020. Metnet: A mutual enhanced transformation network for aspect-based sentiment analysis. In *Proceedings of COLING*.

Constantinos Karouzos, Georgios Paraskevopoulos, and Alexandros Potamianos. 2021. UDALM: unsupervised domain adaptation through language modeling. In *Proceedings of NAACL-HLT*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Ayal Klein, Oren Pereg, Daniel Korat, Vasudev Lal, Moshe Wasserblat, and Ido Dagan. 2022. Opinion-based relational pivoting for cross-domain aspect term extraction. In *Proceedings of ACL*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of ACL*.

Junjie Li, Jianfei Yu, and Rui Xia. 2022a. Generative cross-domain data augmentation for aspect and opinion co-extraction. In *Proceedings of NAACL*.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard H. Hovy. 2021. Dual graph convolutional networks for aspect-based sentiment analysis. In *Proceedings of ACL*.

Shichen Li, Zhongqing Wang, Xiaotong Jiang, and Guodong Zhou. 2022b. Cross-domain sentiment classification using semantic representation. In *Findings of EMNLP*.

Shuo Liang, Wei Wei, Xian-Ling Mao, Fei Wang, and Zhiyong He. 2022. Bisyn-gat+: Bi-syntax aware graph attention network for aspect-based sentiment analysis. In *Findings of ACL*.

Jian Liu, Zhiyang Teng, Leyang Cui, Hanmeng Liu, and Yue Zhang. 2021. Solving aspect category sentiment analysis as a text generation task. In *Proceedings of EMNLP*.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of IJCAI*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arxiv:2203.02155*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia V. Loukachevitch, Evgeniy V. Kotelnikov, Núria Bel, Salud María Jiménez Zafra, and Gülsen Eryigit. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of NAACL-HLT*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of NAACL-HLT*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of COLING*.

Xiaoye Qu, Zhikang Zou, Yu Cheng, Yang Yang, and Pan Zhou. 2019. Adversarial category alignment network for cross-domain sentiment classification. In *Proceedings of NAACL-HLT*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019a. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of NAACL-HLT*.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019b. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of EMNLP-IJCNLP*.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING*.

Hao Tang, Donghong Ji, Chenliang Li, and Qiji Zhou. 2020. Dependency graph enhanced dual-transformer structure for aspect-based sentiment classification. In *Proceedings of ACL*.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2018. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. In *Proceedings of AAAI*.

Cigdem Toprak, Niklas Jakob, and Iryna Gurevych. 2010. Sentence and expression level annotation of opinions in user-generated discourse. In *Proceedings of ACL*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arxiv:2302.13971*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *Proceedings of ICLR*.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of ACL*.

Hui Wu and Xiaodong Shi. 2022. Adversarial soft prompt tuning for cross-domain sentiment analysis. In *Proceedings of ACL*.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In *Proceedings of NAACL-HLT*.

Lu Xu, Lidong Bing, Wei Lu, and Fei Huang. 2020. Aspect sentiment classification with aspect-specific opinion spans. In *Proceedings of EMNLP*.

Qianming Xue, Wei Zhang, and Hongyuan Zha. 2020. Improving domain-adapted sentiment classification by deep adversarial mutual learning. In *Proceedings of AAAI*.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *Proceedings of ACL*.

Jianfei Yu, Chenggong Gong, and Rui Xia. 2021. Cross-domain review generation for aspect-based sentiment analysis. In *Findings of ACL*.

Jianfei Yu and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *Proceedings of EMNLP*.

Mi Zhang and Tieyun Qian. 2020. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *Proceedings of EMNLP*.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022. SSEGCN: syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *Proceedings of NAACL*.

Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. 2020. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In *Proceedings of COLING*.

Yan Zhou, Fuqing Zhu, Pu Song, Jizhong Han, Tao Guo, and Songlin Hu. 2021. An adaptive hybrid framework for cross-domain aspect-based sentiment analysis. In *Proceedings of AAAI*.

Yftah Ziser and Roi Reichart. 2019. Task refinement learning for improved accuracy and stability of unsupervised domain adaptation. In *Proceedings of ACL*.