AMR-based Path Aggregation Graph Network for Aspect-based Sentiment Analysis

Anonymous ACL submission

Abstract

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment classification task. Many recent works have used dependency trees to extract the relationship between aspects and contexts and have achieved significant improvements. However, further improvement is limited due to the mismatch between the dependency tree as a syntactic structure and the sentiment classification as a semantic task. To alleviate this gap, we replace the syntactic dependency tree with the semantic structure, Abstract Meaning Representation (AMR) and propose a model called AMR-based Path Aggregation Graph Network (APAGN). Particularly, we design a path aggregation module which collect local information into global information by path to make full use of AMR. APAGN also contains the outer product summary module which transfers the feature from sentence to graph and the relation-enhanced attention mechanism which transfers the feature in the opposite direction. Experimental results on three public datasets demonstrate the effectiveness of APAGN in aspect-based sentiment analysis when compared with baselines.[1]

1 Introduction

Recent years have witnessed growing popularity of the sentiment analysis tasks in natural language processing (Li and Hovy, 2017; Birjali et al., 2021). Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task to recognize the sentiment polarities of specific aspect terms in a given sentence (Jiang et al., 2011; Li et al., 2018; Seoh et al., 2021; Zhang et al., 2022a). For example, here is a restaurant review "All the money went into the interior decoration, none of it went to the chefs" and the sentiment polarity of the two aspects "interior decoration" and "chefs" are positive and negative, respectively. Thus, ABSA can precisely recognize the corresponding sentiment polarity for any aspect, different from allocating a general sentiment polarity to a sentence in sentence-level sentiment analysis.

[1]Our code and data will be open sourced upon acceptance.

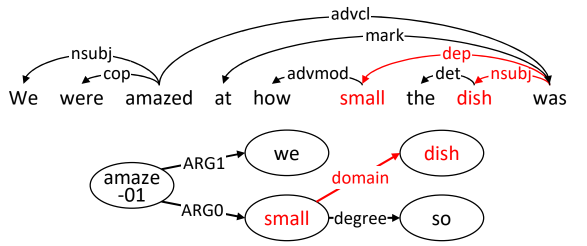

Figure 1: The dependency tree and AMR of the input sentence "we were amazed at how small the **dish** was".

The key challenge for ABSA is to capture the relationship between an aspect and its context, especially the opinion terms. In addition, sentences with multiple aspects and several opinion terms make the problem more complex. To this end, some previous studies (Wang et al., 2016; Chen et al., 2017; Gu et al., 2018; Du et al., 2019) have devoted the main efforts to attention mechanisms. Despite their achievements in aspect-targeted representations and appealing results, this method always suffers noise from the mismatching opinion terms or irrelevant words in contexts.

On the other hand, more recent studies (Zhang et al., 2019; Wang et al., 2020; Tang et al., 2020; Li et al., 2021; Xiao et al., 2021) propose models explicitly exploit dependency trees, the syntactic structure of a sentence, and display significant effectiveness. These models usually employ graph convolutional networks (GCNs) and graph attention networks (GATs) over the syntactic dependencies to identify the interaction between the aspect and the opinion expressions. However, ABSA models utilizing dependency syntax still has the following limitations. First, there is a gap between the **syntactic** dependency structure and the **semantic** sentiment analysis task. Second, nature language parsers including dependency parsers are not absolutely reliable. Without further adjustment, raw

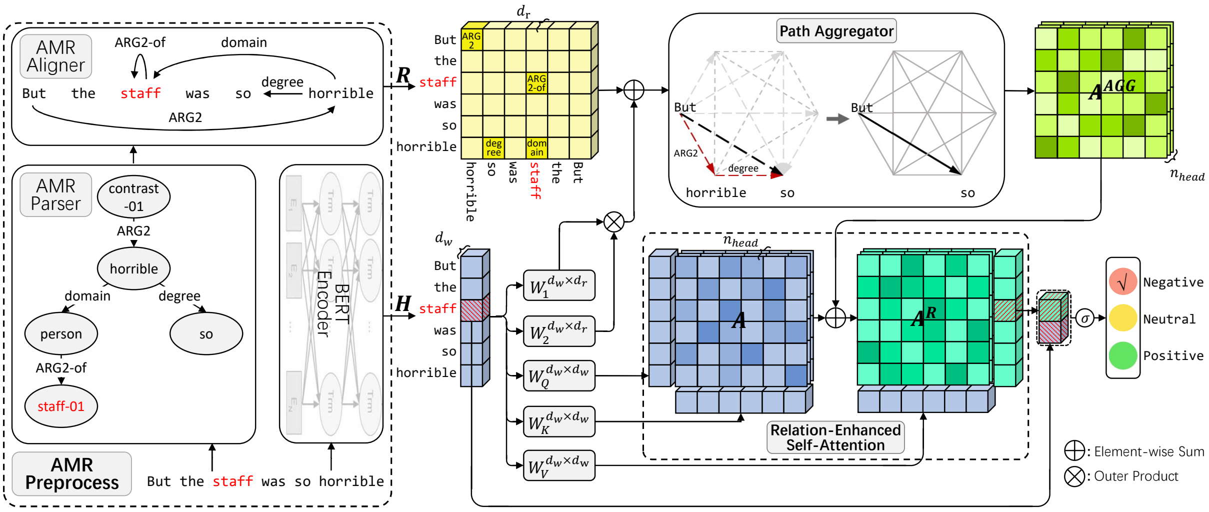

Figure 2: The overall architecture of APAGN.

results of parsers can contain errors and be unsuitable for ABSA task.

To solve aforementioned challenges, we propose a novel architecture called AMR-based Path Aggregation Graph Network (APAGN). For the first challenge, we introduce abstract meaning representations (AMRs), a powerful semantic structure. As shown in Figure 1, the connection on dependency tree between the aspect term "fish" and the opinion term "small" is a 2-hop path for the input sentence "we were amazed at how small the dish was", while they are directly connected in the AMR. Besides, we can notice that the AMR is simpler and more information centralized. To make full use of AMR, we also explore an effective and generalizable process including AMR parsing, aligning and embedding. For the second challenge, we construct the path aggregator module and the relation-enhanced self-attention module. The path aggregator integrates the information from AMRs and sentences to obtain optimized relational features. This procedure not only encourages consistency between semantic structures and basic sentences, but also achieves the global feature by broadcasting local information along the path in the graph. Relation-enhanced self-attention module then injects these relational feature back into attention weights of word features. Credited to these modules, APAGN acquires to utilize sentences and AMRs jointly and achieves higher adaptability and generalization.

To summarize, our main contributions are highlighted as follows:

- We introduce semantic structure into the ABSA task in the form of Abstract Meaning Representations. As a semantic structure, the AMR is more suitable for sentiment analysis task than the syntactic structure such as the dependency tree.
- We propose an ABSA model APAGN which integrates information from both original sentences and parsed structures such as AMRs to relieve the unreliability of the parser. APAGN jointly exploits sentences and AMRs by the path aggregator and the relation-enhanced self-attention mechanism.
- We conducted extensive experiments on three public datasets. These experimental results demonstrate the effectiveness of our APAGN model. Further experiments also show that our model outperforms baselines in cross-domain and low-resource situation.

2 Proposed Model

The overall architecture of our proposed model APAGN is illustrated in Figure 2. It consists of 3 parts: AMR preprocessing, path aggregator and relation-enhanced self-attention mechanism. In the ABSA task, a sentence $s = \{w_1, w_2, ..., w_n\}$ and a specific aspect term $a = \{a_1, a_2, ..., a_m\}$ are given to determine the corresponding sentiment polarity class c_a, where a is a sub-sequence of s and $c_a \in \{Positive, Neutral, Negative\}$.

Many existing works use syntactic dependency trees to establish explicit or implicit connections between aspects and contexts. However, we believe that the sentiment analysis task is essentially about the meanings of sentences, so semantic structures like AMRs are more favorable for this task.

In addition, AMRs are more concise than dependency trees, making it easier to extract valuable information in training but more difficult to preprocess before training. We have to conduct a series of complex steps including: AMR parsing, AMR aligning and AMR embedding. Preprocessed AMRs are not flawless, so we design the path aggregator and the relation-enhanced self-attention mechanism to perform joint representation learning and flexible feature fusion on AMRs together with original sentences. This procedure expands the basic information sources and cross-validates important features, thereby improving the adaptability and generalization of the model.

Next, we elaborate on the details of our proposed APAGN model, including AMR preprocessing and embedding, the path aggregator and the relation-enhanced self-attention mechanism.

2.1 AMR Preprocessing and Embedding

Parsing As we determine to employ the semantic structure AMR as an alternative of the syntactic structure dependency tree to better perform the semantic task ABSA, the first step is parsing the AMR from the input sentence. We choose the best off-the-shelf parser named SPRING (Bevilacqua et al., 2021) for high quality AMR outputs.

Aligning As mentioned above, AMRs are simpler and more abstract than dependency trees. For example, the sentence "we were amazed at how small the dish was" in Figure 1 has 9 nodes and 8 edges in its dependency tree and each node is exactly a word in the sentence, while its AMR has only 5 nodes and 4 edges and some refined nodes are not a word in the sentence. In other words, the AMR does not have a natural alignment with the words in the sentence like a dependency tree. Without alignment with the words in the sentence, it is nearly impossible for the AMR and the sentence to be utilized as a whole satisfactorily. So we have to specifically align the AMR by the aligner LEAMR (Blodgett and Schneider, 2021). In the process of aligning, every node in the AMR is mapped to some distinct words in the sentence. Based on the alignments, we manage to rebuild AMR relations between words in the sentence and get the transformed AMR with words as nodes.

Embedding After aligning, we now have transformed AMRs, which can also be called sentences with AMR relations. Then we need to obtain their embeddings for later representation learning by the model. For the nodes in the AMR, also as words in the sentence, we utilize BERT as an encoder to get contextual embeddings $H = \{h_1, h_2, ..., h_n\}$ like lots of previous works. However, there are few existing studies to reference about the embedding of AMR edges. Considering the convenience of later calculation, we represent the edge relations between nodes as an adjacency matrix $R = \{r_{ij} \mid 1 \leq i, j \leq n\}$, where r_{ij} is the embedding of the edge label between word w_i and word w_j. If there is no edge between w_i and w_j in the AMR, we assign a "none" embedding to r_{ij}. When striving for high-quality edge embeddings, we notice that these special tokens are present in the word vocabulary of the AMR parser mentioned above and have been well fine-tuned. Therefore, we skillfully treat them as excellent edge embeddings which are comparable to the word representations from BERT in terms of information contained.

2.2 Path Aggregator

Path aggregator receives the mix of AMR embeddings $R \in \mathbb{R}^{d_r \times n \times n}$ and sentence embeddings $H \in \mathbb{R}^{d_w \times n}$, where d_r and d_w denote the dimensions of relation and word embeddings, respectively. Path aggregator outputs the relational feature matrix $R^{AGG} = \{r_{ij}^{AGG} \in \mathbb{R}^{d_r} \mid 1 \leq i, j \leq n\}$. This process integrates and condenses information from two different sources, AMRs and sentences, making semantic knowledge more apparent but parsing errors less influential.

Outer Product Sum We first add the outer product of two independent linear transformation of sentence embeddings H to the original AMR embeddings R to obtain sequence-enhanced relation embeddings $R^S \in \mathbb{R}^{d_r \times n \times n}$. On the one hand, as the outer product of H is the representation of word relations from the sentence perspective, its combination with the AMR embeddings R could enlarge the information base of the model to improve the generalization, also cross validate important features to improve the reliability. On the other hand, AMR embeddings R is usually quite sparse. The outer product sum operation ensures the basic density of the feature matrix and facilitate the subsequent representation learning by avoiding the fuzziness and dilution of numerous background "none" relations to the precious effective relations.

Path Aggregation Next, we perform the path aggregation on $R^S = \{r_{ij}^S \mid 1 \leq i, j \leq n\}$ to calculate $R^{AGG} = \{r_{ij}^{AGG} \mid 1 \leq i, j \leq n\}$ as:

$$r'^{S}_{ij} = \text{LayerNorm}(r^{S}_{ij}), \tag{1}$$

$$g_{ij}^{in}, g_{ij}^{out} = \text{sigmoid}(\text{Linear}(r'^{S}_{ij})), \tag{2}$$

$$a_{ij}, b_{ij} = g_{ij}^{in} \odot \text{Linear}(r'^{S}_{ij}), \tag{3}$$

$$r_{ij}^{out} = \text{Linear}(\text{LayerNorm}(\sum_{k} a_{ik} \odot b_{kj})), \tag{4}$$

$$r_{ij}^{AGG} = g_{ij}^{out} \odot r_{ij}^{out}. \tag{5}$$

The path aggregation has distinctive effect on both local and global dissemination of features. From the local view, the path aggregation covers all the 2-hop paths, so that it is very sensitive to neighborhood features, including the features around the aspect term which are really important for the ABSA task. From the global view, information in any long path can be summarized into the representation between the start and the end by several two-in-one operations in enough times of path aggregations. In other words, path aggregations make the features in matrix more inclusive and finally attain global features. In practice, because the ABSA task focuses more on the neighboring information and the BERT encoder with attention mechanisms has made the feature comprehensive enough, a single path aggregation can achieve quite good results.

Additionally, we also introduce a gating mechanism in the path aggregation to alleviate the disturbance of noise from insignificant relations. Finally, the output of path aggregation R^{AGG} is transformed into the relational attention weight matrix $A^{AGG} = \{a_{ij}^{AGG} \mid 1 \leq i, j \leq n\}$ by a linear transformation for subsequent calculation.

2.3 Relation-Enhanced Self-Attention

The classic self-attention (Vaswani et al., 2017) computes the attention weight by the following formula:

$$A = softmax\left(\frac{QW_Q \times (KW_K)^T}{\sqrt{d}}\right), \tag{6}$$

where Q and K are input vectors with d dimensions, while W_Q and W_K are learnable weights with the same size of $\mathbb{R}^{d \times d}$.

In our relation-enhanced self-attention, we added A^{AGG}, the relational attention weight matrix from AMR into the original attention weight, which can be formulated as:

$$A^{R} = softmax\left(\frac{HW_Q \times (HW_K)^T}{\sqrt{d_w}} + A^{AGG}\right), \tag{7}$$

where input vectors W and Q are both replaced by the BERT embeddings H with d_w dimensions. With A^{AGG}, attention outputs are further guided by the semantic information from AMRs, which improves the efficient attention to semantic keywords.

In addition, similar to path aggregator, we also introduced the gating mechanism into the relation-enhanced self-attention as follows:

$$G = sigmoid(HW_G), \tag{8}$$

$$H^{R} = (HW_V)A^{R} \odot G, \tag{9}$$

where W_G and W_V are trainable parameters and G is the gating matrix. Considering the small proportion of effective words in the whole sentence, the gating mechanism is conducive to eliminating background noise, making it easier for the model to focus on the more critical words.

Finally, with all these above calculations including relation-enhanced self-attention and gating mechanism, we obtain the relation-enhanced aspect representation $H_a^R = \{h_{a1}^R, h_{a2}^R, ..., h_{am}^R\}$ for subsequent classification.

2.4 Model Training

The final classification features are concatenated by the original BERT aspect representation $H_a = \{h_{a1}, h_{a2}, ..., h_{am}\}$ and the relation-enhanced aspect representation H_a^R.

$$H_a^{final} = [H_a, H_a^R]. \tag{10}$$

It is passed through a fully connected softmax layer and mapped to probabilities over all different sentiment polarities.

$$p(a) = softmax(W_p H_a^{final} + b_p). \tag{11}$$

We use standard cross-entropy loss as our objective function:

$$L_{CE} = -\sum_{(s,a)\in\mathcal{D}} \sum_{c\in\mathcal{C}} \log p(a), \tag{12}$$

where $\mathcal{D}$ contains all sentence-aspect pairs and $\mathcal{C}$ contains all sentiment polarities.

3 Experiments

In this section, we first introduce the relevant settings of the experiments, including the datasets used, implementation details and baseline methods for comparison. Then, we report the experimental results under normal and special settings. Finally, we select several representative examples for model analysis and discussion.

	Models	Restaurant		Laptop		Twitter	
		Accuracy	Macro-F1	Accuracy	Macro-F1	Accuracy	Macro-F1
w/o BERT	RAM (Chen et al., 2017)	80.23	70.80	74.49	71.35	69.36	67.30
	MGAN (Fan et al., 2018)	81.25	71.94	75.39	72.47	72.54	70.81
	TNet (Li et al., 2018)	80.69	71.27	76.54	71.75	74.90	73.60
	ASGCN (Zhang et al., 2019)	80.77	72.02	75.55	71.05	72.15	70.40
	BiGCN (Zhang and Qian, 2020)	81.97	73.48	74.59	71.84	74.16	75.35
	KumaGCN (Chen et al., 2020)	81.34	73.64	76.12	72.42	72.45	70.77
w. BERT	BERT (Devlin et al., 2019)	85.62	78.28	77.58	72.38	75.28	74.11
	R-GAT (Wang et al., 2020)	86.60	81.35	78.21	74.07	76.15	74.88
	DGEDT (Tang et al., 2020)	86.30	80.00	79.80	75.60	77.90	75.40
	T-GCN (Tian et al., 2021)	86.16	79.95	80.88	77.03	76.45	75.25
	DualGCN (Li et al., 2021)	87.13	81.16	81.80	78.10	77.40	76.02
	dotGCN (Chen et al., 2022)	86.16	80.49	81.03	78.10	78.11	77.00
	SSEGCN (Zhang et al., 2022b)	87.31	81.09	81.01	77.96	77.40	76.02
Ours	APAGN	**87.76**	**82.44**	**81.96**	**78.19**	**79.76**	**78.79**

Table 1: Results comparison on three public datasets. Best performed baselines are underlined.

3.1 Datasets

Our experiments are conducted on three commonly used public standard datasets. The Twitter dataset is a collection of tweets built by Dong et al. (2014), while the Restaurant and Laptop dataset come from the SemEval 2014 Task (Pontiki et al., 2014). The data statistics is shown in Appendix A.1.

3.2 Implementation Details

APAGN uses the BERT of bert-base-uncased version as a pre-trained encoder with max length as 100. During training, we use Adam with the learning rate of 2×10^{-5} and hyper-parameters α of 0.9 and β of 0.98. The BERT encoder and other parts of the model use dropout strategies with probability 0.5 and 0.2, respectively. Following Li et al. (2021), each training lasts for 15 epochs and the evaluation is performed every 5 batches. The model with the highest accuracy among all evaluation results is selected as the final result of this training. Reported results are the average of three runs with different random seeds. See Appendix A.2 for more details.

3.3 Baseline Methods

We compare APAGN with a series of baselines and state-of-the-art alternatives, including:
1) **RAM** (Chen et al., 2017) applies multiple attention mechanisms to memory networks.
2) **MGAN** (Fan et al., 2018) designs a multi-scale attention mechanism to mine aspect relations.
3) **TNet** (Li et al., 2018) converts BiLSTM embeddings to aspect-specific embeddings and uses CNN to further obtain final features for classification.
4) **ASGCN** (Zhang et al., 2019) first proposes to learn aspect-specific representations with GCN.
5) **BiGCN** (Zhang and Qian, 2020) uses a hierarchical graph structure to integrate token co-occurrence information and dependency type information.
6) **kumaGCN** (Chen et al., 2020) utilizes an implicit graph structure to provide syntactic features.
7) **BERT** (Devlin et al., 2019) is composed of a general pre-trained BERT model and a classification layer adapted to the ABSA task.
8) **R-GAT** (Wang et al., 2020) proposes a dependency structure adjusted for aspects and uses a relational GAT to encode this structure.
9) **T-GCN** (Tian et al., 2021) proposes an approach to explicitly utilize dependency types for ABSA with type-aware GCNs.
10) **DGEDT** (Tang et al., 2020) proposes a dual transformer structure based on dependency graph augmentation, which can simultaneously fuse representations of sequences and graphs.
11) **DualGCN** (Li et al., 2021) proposes a dual GCN structure and regularization methods to merge features from sentences and dependency trees.
12) **dotGCN** (Chen et al., 2022) proposes an aspect-specific and language-agnostic discrete latent tree as an alternative structure to dependency trees.
13) **SSEGCN** (Zhang et al., 2022b) proposes an aspect-aware attention mechanism to enhance the node representations with GCN.

3.4 Main Results

Table 1 shows the experimental results of our model and the baseline models on three datasets under

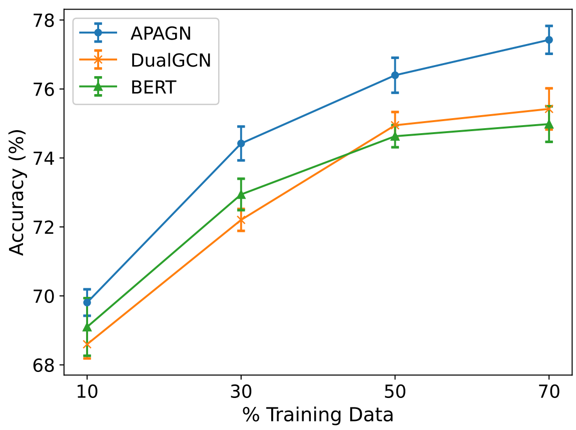

Figure 3: Accuracy on Twitter dataset with partial data.

	Models	R→ L	R→ T	L→ R	L→ T	T→ R	T→ L
	BERT	79.94	**59.88**	81.82	59.11	74.75	72.88
Acc	DualGCN	79.62	59.32	81.84	59.26	73.62	73.45
	APAGN	**80.00**	59.68	**81.99**	**60.80**	**75.17**	**73.77**
	BERT	76.58	57.04	73.48	58.66	64.73	68.09
F1	DualGCN	76.36	58.26	72.82	58.33	65.07	68.63
	APAGN	**76.79**	**58.37**	**73.82**	**59.09**	**67.33**	**68.81**

Table 2: Results on six kinds of cross-domain datasets. R, L, T represent Restaurant, Laptop, Twitter datasets.

the same conventional settings as Li et al. (2021), where the best results are in bold and the second best results are underlined. Our APAGN model exhibits excellent results and achieves the best results on all 6 indicators of 3 datasets, which fully proves the effectiveness of this model.

Comparing the results of different datasets, we can find that the improvement of APAGN on the Twitter dataset is particularly obvious. Compared to the best results, the accuracy rate has increased by 1.65% and the Macro-F1 has increased by 1.79%. The main reason is the similarity of the Twitter dataset to the AMR 3.0 dataset, the training dataset for the AMR parser we used. More than half of the corpus of the AMR 3.0 dataset comes from internet forums and blogs, which are similar to the Twitter dataset as they are both social media. As a result, the AMR parser has better output on the Twitter dataset, which in turn enables the model to extract more valuable features from it and leads to a considerable improvement. This difference among datasets also reflects the effectiveness of semantic information from AMR for the ABSA task.

3.5 Special Situation Study

Low-resource Situation The low-resource scenario is a special scenario that ABSA tasks may actually face. Exploring the performance of the model in this scenario is of great significance to understand the adaptability and application value of the model. We test three models in these experiments, including: the naive pre-training model BERT, the DualGCN model with available code and the best comprehensive performance in existing research, and our APAGN model. These experiments are conducted on the Twitter dataset with the largest total data volume. In these experiments, the input training data is part of all the training data in the dataset and the results are shown in Figure 3.

We can notice that the APAGN model has good adaptability to low-resource scenarios. In multiple experiments with different amounts of input data, APAGN consistently outperforms BERT by 1.6% on average, while the DualGCN model is inferior to BERT in some cases. Considering that AMRs are more compact than dependency trees, it is reasonable that APAGN can efficiently utilize AMRs and perform well with only a small training set.

Cross-domain Situation The cross-domain scenario is another possible special scenario for the ABSA task, which requires the model to have good generalization ability. In cross-domain experiments, the training set of one dataset is used for training and the test set of another dataset is used for testing. Therefore, six cross-domain datasets are formed from three original datasets of Restaurant, Laptop and Twitter. Three models of BERT, DualGCN and APAGN are tested and the performance of each model is shown in Table 2.

We can see that in the cross-domain scenario, the performance of each model is significantly affected by the dataset. BERT and DualGCN compete with each other on different cross-domain datasets, while the APAGN model has the best overall performance because its generalization ability is improved by the joint use of sentences and AMR. APAGN improves less when using the restaurant dataset as a training set because of the specificity of this dataset, which makes the semantic structure information learned from it more difficult to transfer to other datasets.

3.6 Model Analysis

Ablation Study In order to analyze the role of each module, we separately remove four key components of the APAGN model in the ablation studies, and the results are shown in Table 3.

According to the results, each of the four compo-

Models	Restaurant		Laptop		Twitter	
	Acc	F1	Acc	F1	Acc	F1
APAGN	**87.76**	**82.44**	**81.96**	**78.19**	**79.76**	**78.79**
−Outer Product Sum	86.15	80.13	79.43	75.22	76.22	74.75
−Path Aggregation	87.04	81.61	79.11	74.76	76.66	74.9
−Relation in Self-Atteion	87.49	81.82	80.22	76.58	76.81	75.49
−Gate in Self-Attenion	85.61	78.49	79.75	76.14	77.55	76.06

Table 3: Ablation experimental results of our APAGN.

nents contributes significantly to the performance of the APAGN model. Removing Outer Product Sum results in a significant drop in performance, illustrating the importance of promoting consistency of information from sentences and AMRs. Removing Relation in Self-Attention is worse than removing Path Aggregation, indicating that unprocessed AMR information can only interfere with the model instead of being exploited by the model.

Comparing the results in different datasets, we can find that the model depends on information from sentences and AMRs differently on different datasets. On the Restaurant dataset, removing the Relation in Self-Attention component has less impact, while on the Twitter dataset, removing this component has a greater impact. This means the model utilizes sentence information more on the Restaurant dataset and AMR information more on the Twitter dataset. This is also consistent with the analysis of the main results: the AMR of Twitter dataset has higher quality due to the domain relatedness with the training dataset of the AMR parser, which in turn makes the model pay more attention to the information from the AMR on this dataset.

Sentence Length Study Figure 4 compares the accuracy of the APAGN model with and without path aggregator for sentences of different lengths in the Restaurant dataset. According to the figure, we can see that the model achieve higher accuracy on short sentences, while the long sentences is more challenging. In addition, the model with the path aggregator has a larger relative improvement on long sentences, indicating that the path aggregator can effectively help the model to capture long-distance relations with AMR.

Edge Embedding Analysis These experiments investigate the effect of AMRs' edge label embeddings to the result. Three different types of edge label embeddings are tested in the experiments and the results are shown in Table 4.

According to the results, using pretrained edge label embeddings outperforms using randomly ini-

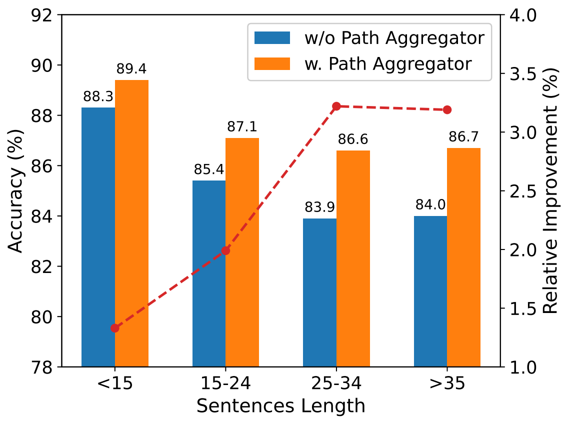

Figure 4: Accuracy of sentences with different length from Restaurant dataset. The red line represents the percentage of accuracy improvement.

Embeddings	Restaurant		Laptop		Twitter	
	Acc	F1	Acc	F1	Acc	F1
Fixed	85.79	79.01	80.22	76.49	76.95	75.49
Random	86.42	80.68	80.70	77.48	77.70	76.12
Pre-trained	**87.76**	**82.44**	**81.96**	**78.19**	**79.76**	**78.79**

Table 4: Results of three different kinds of edge embeddings on all datasets.

tialized edge label embeddings on all datasets, which demonstrates the effectiveness of pretrained edge label embeddings. Also, the use of fixed edge labels is worse than the use of randomly initialized true edge labels, which shows that the edge labels of AMR contain important information and play important roles in the ABSA task.

3.7 Case Study

As shown in Figure 5, we selected three typical cases to visualize the aspect terms' attention to the context before and after adding information from the AMR, respectively.

From the first two examples, we can notice that the model focuses on the copula verb next to the opinion term without the AMR. While with the information from the AMR, the model can capture opinion terms through the attention mechanism more accurately. In the third example, without the AMR, the model pays more attention to words that are closer to the aspect term. With the semantic information from AMR, the model can discover opinion terms farther away from aspect terms.

These cases illustrate that the semantic structure information of AMR plays an important role in making the model focus on the correct opinion words. It also shows that the structure of our

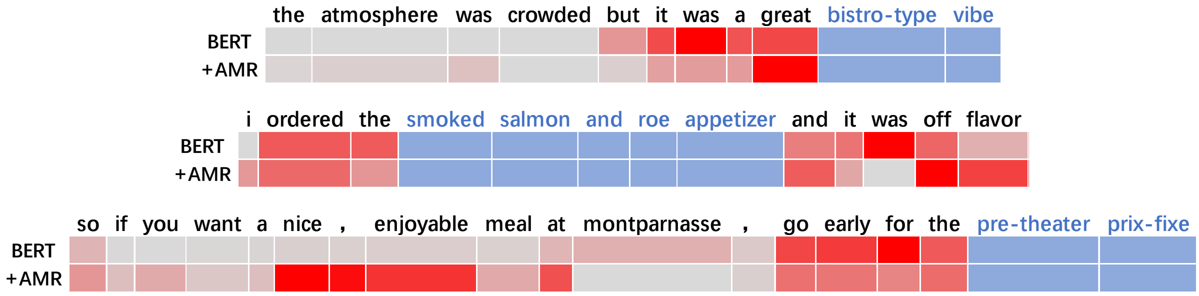

Figure 5: Visualization of aspect terms' attention to the context in three cases. Aspect terms are highlighted in blue.

APAGN model can effectively utilize the semantic structure information in AMR to improve the performance in the ABSA task.

4 Related Work

Aspect-based Sentiment Analysis Traditional sentiment analysis tasks are usually sentence-level or document-level, while the ABSA task is an entity-level and fine-grained sentiment analysis task. Early methods (Jiang et al., 2011; Kiritchenko et al., 2014) are mostly based on artificially constructed features, which are difficult to effectively model the relations between aspect terms and its context. With the development of deep neural networks, many recent works (Wang et al., 2016; Tang et al., 2016; Chen et al., 2017; Fan et al., 2018; Gu et al., 2018; Du et al., 2019) have explored applying attention mechanisms to implicitly model the semantic relations of aspect terms and identify the key opinion terms in the context.

Another trend in ABSA studies is the explicit use of dependency trees. Some works (He et al., 2018; Zhang et al., 2019; Sun et al., 2019; Huang and Carley, 2019; Zhang and Qian, 2020; Chen et al., 2020; Liang et al., 2020; Wang et al., 2020; Tang et al., 2020; Phan and Ogunbona, 2020; Li et al., 2021; Xiao et al., 2021) extend GCN, GAT, and Transformer backbones to process syntactic dependency trees and develop several outstanding models. These models shorten the distance between aspect terms and opinion terms by dependency trees and alleviate the long-term dependency problem.

Recent studies have also noticed the limitations of dependency trees in the ABSA task. Chen et al. (2020) propose to combine dependency trees with induced aspect-specific latent maps. Chen et al. (2022) further proposed an aspect-specific and language-independent discrete latent tree model as an alternative structure for dependency trees. Our work is similar in that we also aim at the mismatch between dependency trees and the ABSA task, but different in that we introduce a semantic structure named Abstract Meaning Representation instead of induced trees.

Abstract Meaning Representation AMR is a structured semantic representation that represents the semantics of sentences as a rooted, directed, acyclic graph with labels on nodes and edges. AMR is proposed by Banarescu et al. (2013) to provide a specification for sentence-level comprehensive semantic annotation and analysis tasks. Research on AMR can be divided into two categories, AMR parsing (Cai and Lam, 2020; Zhou et al., 2021; Hoang et al., 2021) and AMR-to-Text (Zhao et al., 2020; Bai et al., 2020; Ribeiro et al., 2021).

AMR has also been applied in many NLP tasks. Kapanipathi et al. (2020) use AMR in question answering system. Lim et al. (2020) employ AMR to improve common sense reasoning. Wang et al. (2021) utilize AMR to add pseudo labels to unlabeled data in low-resource event extraction task. Our model also improves the performance of the ABSA task with AMR.

5 Conclusion

In this paper, we propose APAGN, an AMR-based Path Aggregation Graph Network for the ABSA task. Different from the traditional ABSA model utilizing the syntactic structure like dependency tree, our model employs the semantic structure called Abstract Meaning Representation which is more harmony with the sentiment analysis task. We propose the path aggregator and the relation-enhanced self-attention mechanism to efficiently exploit AMRs and integrate information from AMRs and input sentences. These designs enable our model to achieve better results than existing models, as well as greater adaptability and generalization. Experiments on three public datasets show that APAGN outperforms competing baselines.

Limitations

The high computational complexity is one of the biggest disadvantages of the path aggregation. The time consumption and GPU memory used for multiple operations are expensive. So it is very desirable to use only one time of path aggregation due to attributes of the ABSA task in our APAGN model.

Another limitation of this work is that the performance of the model is still somewhat affected by the quality of the AMR parsing results. The good news is that the research on AMR parsing is continuing to make progress. In the future, APAGN with higher quality AMRs is expected to further improve the level of the ABSA task.

References

Xuefeng Bai, Linfeng Song, and Yue Zhang. 2020. Online back-parsing for AMR-to-text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1206–1219, Online. Association for Computational Linguistics.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Thirty-Fifth AAAI Conference on Artificial Intelligence and Thirty-Third Conference on Innovative Applications of Artificial Intelligence and The Eleventh Symposium on Educational Advances in Artificial Intelligence*, pages 12564–12573, Online. AAAI Press.

Marouane Birjali, Mohammed Kasri, and Abderrahim Beni Hssane. 2021. A comprehensive survey on sentiment analysis: Approaches, challenges and trends. *Knowledge-Based Systems*, 226:107134.

Austin Blodgett and Nathan Schneider. 2021. Probabilistic, structure-aware algorithms for improved variety, accuracy, and coverage of AMR alignments. 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 3310–3321, Online. Association for Computational Linguistics.

Deng Cai and Wai Lam. 2020. AMR parsing via graph-sequence iterative inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1290–1301, Online. Association for Computational Linguistics.

Chenhua Chen, Zhiyang Teng, Zhongqing Wang, and Yue Zhang. 2022. 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, Dublin, Ireland. Association for Computational Linguistics.

Chenhua Chen, Zhiyang Teng, and Yue Zhang. 2020. Inducing target-specific latent structures for aspect sentiment classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5596–5607, Online. Association for Computational Linguistics.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Baltimore, Maryland. Association for Computational Linguistics.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Tong Xu, and Ming Liu. 2019. Capsule network with interactive attention for aspect-level sentiment classification. 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 5489–5498, Hong Kong, China. Association for Computational Linguistics.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442, Brussels, Belgium. Association for Computational Linguistics.

Shuqin Gu, Lipeng Zhang, Yuexian Hou, and Yin Song. 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 774–784, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

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

Thanh Lam Hoang, Gabriele Picco, Yufang Hou, Young-Suk Lee, Lam Nguyen, Dzung Phan, Vanessa Lopez, and Ramon Fernandez Astudillo. 2021. Ensembling graph predictions for amr parsing. In *Advances in Neural Information Processing Systems*, volume 34, pages 8495–8505, Online. Curran Associates, Inc.

Binxuan Huang and Kathleen Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. 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 5469–5477, Hong Kong, China. Association for Computational Linguistics.

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

Pavan Kapanipathi, Ibrahim Abdelaziz, Srinivas Ravishankar, Salim Roukos, Alexander G. Gray, Ramón Fernandez Astudillo, Maria Chang, Cristina Cornelio, Saswati Dana, Achille Fokoue, Dinesh Garg, Alfio Gliozzo, Sairam Gurajada, Hima Karanam, Naweed Khan, Dinesh Khandelwal, Young-Suk Lee, Yunyao Li, Francois P. S. Luus, Ndivhuwo Makondo, Nandana Mihindukulasooriya, Tahira Naseem, Sumit Neelam, Lucian Popa, Revanth Gangi Reddy, Ryan Riegel, Gaetano Rossiello, Udit Sharma, G. P. Shrivatsa Bhargav, and Mo Yu. 2020. Question answering over knowledge bases by leveraging semantic parsing and neuro-symbolic reasoning. *CoRR*, abs/2012.01707.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. NRC-Canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442, Dublin, Ireland. Association for Computational Linguistics.

Jiwei Li and Eduard Hovy. 2017. *Reflections on Sentiment/Opinion Analysis*, pages 41–59. Springer International Publishing, Cham.

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, Online. Association for Computational Linguistics.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018. Transformation networks for target-oriented sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 946–956, Melbourne, Australia. Association for Computational Linguistics.

Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 150–161, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jungwoo Lim, Dongsuk Oh, Yoonna Jang, Kisu Yang, and Heuiseok Lim. 2020. I know what you asked: Graph path learning using AMR for commonsense reasoning. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2459–2471, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Minh Hieu Phan and Philip O. Ogunbona. 2020. Modelling context and syntactical features for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3211–3220, Online. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Jonas Pfeiffer, Yue Zhang, and Iryna Gurevych. 2021. Smelting gold and silver for improved multilingual AMR-to-Text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 742–750, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ronald Seoh, Ian Birle, Mrinal Tak, Haw-Shiuan Chang, Brian Pinette, and Alfred Hough. 2021. Open aspect target sentiment classification with natural language prompts. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6311–6322, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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, Hong Kong, China. Association for Computational Linguistics.

Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224, Austin, Texas. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008, Long Beach, CA, USA. Curran Associates, Inc.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238, Online. Association for Computational Linguistics.

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, Austin, Texas. Association for Computational Linguistics.

Ziqi Wang, Xiaozhi Wang, Xu Han, Yankai Lin, Lei Hou, Zhiyuan Liu, Peng Li, Juanzi Li, and Jie Zhou. 2021. CLEVE: Contrastive Pre-training for Event Extraction. 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 6283–6297, Online. Association for Computational Linguistics.

Zeguan Xiao, Jiarun Wu, Qingliang Chen, and Congjian Deng. 2021. BERT4GCN: Using BERT intermediate layers to augment GCN for aspect-based sentiment classification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9193–9200, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. 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 4568–4578, Hong Kong, China. Association for Computational Linguistics.

Mi Zhang and Tieyun Qian. 2020. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3540–3549, Online. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2022a. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *arXiv preprint arXiv:2203.01054*.

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, Seattle, United States. Association for Computational Linguistics.

Yanbin Zhao, Lu Chen, Zhi Chen, Ruisheng Cao, Su Zhu, and Kai Yu. 2020. Line graph enhanced AMR-to-text generation with mix-order graph attention networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 732–741, Online. Association for Computational Linguistics.

Jiawei Zhou, Tahira Naseem, Ramón Fernandez Astudillo, Young-Suk Lee, Radu Florian, and Salim Roukos. 2021. Structure-aware fine-tuning of sequence-to-sequence transformers for transition-based AMR parsing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6279–6290, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Appendix

A.1 Datasets

The statistics for the Restaurant dataset, Laptop dataset and Twitter dataset are shown in Table 5. Each sentence in these datasets is annotated with aspect terms and corresponding polarities. Following Li et al. (2021), we remove instances with the "conflict" label. So all datasets have three sentiment polarities: positive, negative and neutral. Throughout the research, we follow the Creative Commons Attribution 4.0 International Licence of the datasets.

Dataset	Positive		Neutral		Negative	
	Train	Test	Train	Test	Train	Test
Restaurant	2164	728	807	196	637	196
Laptop	994	341	870	128	464	169
Twitter	1561	173	3127	346	1560	173

Table 5: Statistics of the three ABSA datasets

A.2 Implementation Details

In data preprocessing, we use SPRING (Bevilacqua et al., 2021) as the parser to obtain the AMRs of input sentences and use LEAMR (Blodgett and Schneider, 2021) as the AMR aligner to establish the correspondence between the AMRs and sentences. The maximum length of the input sentence is set to 100, the shortage is made up with the special word "PAD" and the excess is truncated.

Some edge labels are treated specially when mapping the edges of AMR to the relations between words. Edge labels suffixed with "-of" are used to avoid loops in AMR, so we swap their start and end points and remove the "-of" suffix, eg: the ":ARG0-of" relation from $token_i$ to $token_j$ is changed to the ":ARG0" relation from $token_j$ to $token_i$. Edge labels prefixed with ":prep-" are used because there is no suitable preposition label in the AMR specification. We changed them to original prepositions, for example, ":prep-against" is changed to "against".

APAGN uses the BERT of bert-base-uncased version as a pre-trained encoder. The dimension of its output is 768, which is also used as the dimension of token representation in the path aggregator. The dimension of the AMR edge label embedding derived from the SPRING model is 1024. Due to computational efficiency and memory usage, this dimension is reduced to 376 through a linear layer as the dimension of the relational matrix features in the path aggregator. For the relation-enhanced self-attention mechanism, its gated multi-head attention mechanism uses 8 attention heads with the latent dimension size of 64. The total parameter size of APAGN is about 130M and it takes about 8 minutes to train each epoch on a single RTX 3090 GPU.