Available Online: 19 May 2026 DOI: 10.54254/2977-3903/2026.33752

A dual-graph collaborative model for aspect-based sentiment analysis based on adaptive pruning and key-path enhancement

Xinpei Wang

School of Artificial Intelligence and Computer Science, North China University of Technology, Beijing, China

Wangxinpei0908@163.com

Abstract. Aspect-based sentiment analysis has attracted extensive attention from both academia and industry. Existing approaches often suffer from a large number of noise nodes unrelated to aspect-term sentiment judgment when processing complex syntactic structures, while long-distance dependencies and intricate syntactic-semantic relations are difficult to capture effectively. To address these issues, this paper proposes a dual-graph collaborative model for aspect-based sentiment analysis based on adaptive pruning and key-path enhancement. First, for the syntactic dependency tree, the model introduces adaptive distance pruning and a key-path enhancement strategy, which removes global noise while significantly amplifying the structural signals between aspect terms and opinion words. Second, a semantic similarity dual graph is constructed to capture implicit long-range dependencies. Finally, a multi-source output fusion module is introduced to deeply integrate the refined syntactic features with dense semantic features, thereby enabling efficient interaction and collaborative modeling of multi-source information. Experimental results on several public datasets demonstrate that the proposed model outperforms existing mainstream baseline methods in both accuracy and Macro-F1 score, validating the effectiveness of the pruning-enhancement and dual cross-fusion strategies.

Keywords: aspect-based sentiment analysis, syntactic dependency tree, graph convolutional network, dual graph, pruning strategy

1. Introduction

With the rapid development of social media and e-commerce platforms, user-generated textual data has grown explosively, containing abundant sentiment information. Aspect-Based Sentiment Analysis (ABSA), as a core task in fine-grained sentiment analysis, aims to identify the sentiment polarity toward specific aspect terms in text and has become a major research focus in the field of natural language processing [1]. This task is of great practical significance in application scenarios such as product review analysis, public opinion monitoring, and intelligent customer service. It enables enterprises to accurately understand users' satisfaction with different product attributes and provides data support for decision-making optimization.

2. Related work

Traditional approaches to aspect-based sentiment analysis mainly rely on recurrent neural networks or attention mechanisms to model contextual information. However, these methods struggle to effectively capture the structured dependency relationships between aspect terms and opinion words. In recent years, graph neural network methods based on syntactic dependency trees have achieved remarkable progress. By constructing sentences as graph structures and applying Graph Convolutional Networks (GCNs), these methods can explicitly model syntactic associations among words. Zhang et al. [2] (2019) proposed ASGCN (Aspect-Specific Graph Convolutional Network), which was the first to apply GCNs to aspect-based sentiment classification by using the sentence dependency parsing tree as the graph structure. Bai et al. [3] introduced a type-aware graph attention network that assigns different attention weights to different types of syntactic dependencies. Chen et al. [4] further designed a Typed Graph Convolutional Network (T-GCN) and incorporated a layer integration mechanism to capture features from different hierarchical levels. Zhou et al. [5] proposed a dual-channel GCN that combines position-enhanced word representations with graph convolutional networks to capture syntactic information.

To overcome the limitations of static graphs, researchers have proposed multi-graph fusion and dynamic graph methods. DualGCN [6] simultaneously models syntactic structures and semantic correlations, thereby alleviating the inaccuracies caused by the excessive reliance of traditional GCN methods on a single syntactic parsing result. MulGCN [7] parallelly constructs three GCNs based on syntax, semantics, and context, and integrates these features through a multi-head attention mechanism, demonstrating the effectiveness of multi-source knowledge fusion. By integrating syntactic dependency trees, SenticNet sentiment knowledge, and aspect-aware techniques, this method captures three categories of knowledge simultaneously.

Although graph neural network methods based on syntactic dependency trees have achieved substantial progress, several limitations still remain. First, existing methods usually construct graph structures directly from complete syntactic dependency trees, which contain a large number of noise nodes unrelated to aspect-term sentiment judgment. These noise nodes interfere with information transmission between aspect terms and the actual sentiment-bearing words. Second, standard syntactic dependency trees mainly capture local syntactic dependency relations and are insufficient for modeling long-range semantic associations across clauses as well as implicit semantic similarity relationships. Third, syntactic structural information and semantic representation information are often processed independently or simply concatenated, lacking a deep interactive fusion mechanism, which makes it difficult for the two heterogeneous types of information to achieve collaborative enhancement. To address the above issues, this paper proposes an Adaptive Pruning and Key-Path Enhanced Dual-Graph Collaborative Network for Aspect-Based Sentiment Analysis (AKP-DGCN). The proposed model highlights the core propagation paths from aspect terms to opinion words through adaptive pruning and key-path enhancement strategies on the syntactic graph. In addition, a multi-source output fusion module is designed to achieve deep cross-fusion between syntactic features and semantic features, thereby promoting the collaborative complementarity of multi-source information. The main contributions of this paper are summarized as follows:

(1) An adaptive distance pruning module is proposed to effectively suppress noise interference in syntactic graphs.

(2) A key-path enhancement module is designed to strengthen the semantic dependency between aspect terms and sentiment words.

(3) A semantic-syntactic collaborative interaction mechanism is constructed to realize the deep fusion of dual-graph features.

(4) Experimental results on multiple public datasets demonstrate that the proposed model outperforms existing mainstream baseline methods in terms of both accuracy and F1 score. Ablation studies further verify the effectiveness of each module design.

3. Aspect-based sentiment analysis model based on a pruning-enhanced dual-graph convolutional cross-fusion framework

At present, graph neural network modeling based on syntactic dependency trees for aspect-based sentiment analysis suffers from two major issues: the presence of noise nodes and insufficient fusion between syntactic and semantic information. To address these limitations, this paper introduces three key components: an adaptive distance pruning module, a key-path enhancement module, and a multi-source output fusion module. The adaptive distance pruning module dynamically removes distant and weakly relevant nodes, effectively reducing noise in the syntactic graph. The key-path enhancement module strengthens dependency paths between aspect terms and sentiment words, highlighting critical semantic information. The multi-source output fusion module smoothly integrates syntactic and semantic features, thereby improving the collaborative modeling capability of heterogeneous information sources.

The overall architecture of the proposed AKP-DGCN model is illustrated in Figure 1.

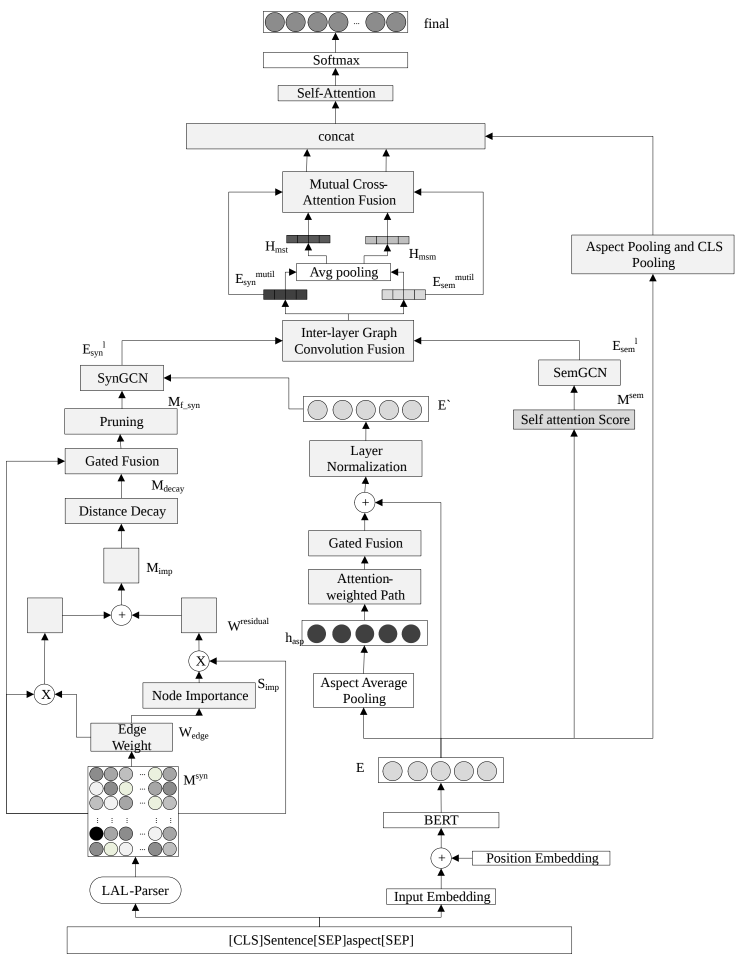

Figure 1. Overall architecture of the AKP-DGCN model

3.1. Syntactic-aware feature extraction module

The goal of syntactic-aware feature extraction is to construct a graph structure based on dependency parsing results and, through a series of optimization operations, enable the model to focus on syntactic paths that are crucial for sentiment expression, thereby producing node representations enriched with structural information.

3.1.1. Dependency graph initialization

Dependency parsing is the first step in constructing the dependency graph, aiming to explicitly represent words in a sentence and their syntactic dependency relations. Common dependency parsers include the Stanford Dependency Parser and spaCy, which are typically built on context-free grammar (CFG) or neural end-to-end models for dependency annotation. In this paper, we adopt the LAL-Parser to extract the syntactic dependency tree $\mathrm{Tree}^{\mathrm{syn}}$ of a sentence and transform it into a dependency probability matrix. Given an input sentence $S=[s_1,s_2,s_3....s_n]$, it is fed into the LAL-Parser to obtain a relation probability matrix (RPM), which is used as the input $\mathrm{M}^{\mathrm{syn}}\in\mathrm{R}^{\mathrm{n}\times\mathrm{n}}$ for subsequent graph convolution operations.

3.1.2. Adaptive distance pruning module

The original dependency tree produced by the LAL-Parser may contain long-distance dependency edges that introduce noise or irrelevant information. To preserve key sentiment-related paths, we design an adaptive distance pruning module to dynamically refine the adjacency matrix.

First, to capture semantic relevance between node pairs (i,j), we concatenate their hidden states $\mathrm{E_i}$ and $\mathrm{E_j}$, and apply a linear transformation $\mathrm{W_e}\in\mathrm{R}^{2\mathrm{d_e}\times 1}$ to compute the edge weight $\mathrm{W_{edge}}$. A sigmoid function σ() is used to ensure non-negativity:

$$W_{edge}(i,j)=\sigma(W_e\cdot[E_i\bigoplus E_j]) \tag{1}$$

where $\bigoplus$ denotes vector concatenation.

Second, to avoid losing critical hub-node information during pruning, a node importance scoring mechanism is introduced. A node-level linear layer $\mathrm{W_s}\in\mathrm{R}^{2\mathrm{d_e}\times 1}$ is used to compute the importance of each node $\mathrm{I(h_k)}$. The average importance of a node pair (i,j) is defined as a residual protection term $\mathrm{S_{imp}}$. To prevent excessive pruning, a minimum weight is assigned to each existing edge in the original syntactic matrix $\mathrm{M}^{\mathrm{syn}}$($\mathrm{M}^{\mathrm{syn}}[\mathrm{i,j}]\neq 0$), determined jointly by $\mathrm{S_{imp}}$ (i,j) and the hyperparameter λ_{min}. The residual and structural weights are then combined to form the importance-enhanced matrix $\mathrm{M_{imp}}$, as shown in Equations (2)–(5):

$$I(h_k)=\sigma(W_s\cdot h_k) \tag{2}$$

$$S_{imp}(i,j)=\frac{I(h_i)+I(h_j)}{2} \tag{3}$$

$$w_{ij}^{residual}=\lambda_{min}\cdot S_{imp}(i,j)\cdot M^{syn}[i,j] \tag{4}$$

$$M_{imp}[i,j]=W_{edge}(i,j)\cdot M^{syn}[i,j]+w_{ij}^{residual} \tag{5}$$

In this formulation, when two nodes are strongly related, the first term dominates and preserves the structural connection. When semantic similarity is weak but node importance is high ($\mathrm{S_{imp}}$ (i,j) $=1$), the residual term still maintains a weak connection, preventing disruption of key syntactic structures.

In addition, to address the presence of long-distance dependency edges in syntactic trees—which often introduce higher noise—a distance decay mechanism is introduced. It applies an exponential penalty $\mathrm{D}_{\mathrm{dist}}^{(\mathrm{i,j})}$ based on the linear sequence distance |i-j|. The farther the distance, the smaller the decay coefficient, thereby reducing the influence of long-range noisy edges in message passing. This is formulated as:

$$D_{dist}^{(i,j)}=exp(-\alpha\cdot|i-j|) \tag{6}$$

$$M_{decay}[i][j] = M_{imp}[i,j] \cdot D_{dist}^{(i,j)} \tag{7}$$

where α is a hyperparameter controlling the strength of distance-based attenuation.

To balance the original syntactic graph and the learned structural graph, a gated fusion mechanism is introduced. It dynamically computes fusion weights based on node features, enabling adaptive integration of both graphs to obtain the fused matrix M_{merge} :

$$gate_i = \sigma(W_g \cdot E_i) \tag{8}$$

$$M_{merge}[i,j] = gate_i \cdot M^{syn}[i,j] + (1 - gate_i) \cdot M_{decay}[i,j] \tag{9}$$

where $W_g \in R^{d_e \times 1}$ is a learnable gating parameter matrix.

To mitigate gradient vanishing/exploding issues in graph convolution and ensure consistent feature aggregation across nodes, symmetric normalization is applied to the fused adjacency matrix:

$$\widehat{D}_{ii} = (\sum_j M_{merge}[i,j] + \epsilon)^{-\frac{1}{2}} \tag{10}$$

$$M_{norm} = \widehat{D}^{\frac{1}{2}} \cdot M_{merge} \cdot \widehat{D}^{\frac{1}{2}} \tag{11}$$

Here, ϵ is a small constant chosen to prevent division-by-zero errors, and $\widehat{D}$ is the negative of the one-half power of the diagonal elements of the degree matrix.

Although these adjustments improve robustness, M_{norm} may still contain numerous low-weight redundant edges, increasing computational cost and introducing weak noise. Therefore, a final threshold pruning step is applied. Edges with weights below threshold τare set to zero; otherwise, they are retained, producing a sparse matrix M_t :

$$M_t[i][j] = \begin{cases} M_{norm}[i][j], M_{norm}[i][j] \geq \tau \\ 0 other \end{cases} \tag{12}$$

After obtaining the sparsified relation probability matrix M_t, self-loops are introduced into the adjacency matrix to prevent nodes from losing their own feature representations during the message-passing process of graph convolution. Specifically, an identity matrix I is added to the diagonal of M_t, enabling each node to retain its own feature representation while aggregating information from its neighbors. This yields the enhanced syntactic adjacency matrix M_{f_syn}, as shown in Equation (12):

$$M_{f\_syn} = M_t + I \tag{13}$$

3.1.3. Key-path enhancement module

Although adaptive distance pruning effectively suppresses noisy edges in syntactic dependency graphs, there often exist long-range semantic dependencies between aspect terms and opinion words. Conventional key-path enhancement methods strengthen such dependencies by explicitly constructing path-weight matrices and adding them to the adjacency matrix. However, these approaches rely on discrete graph structures and are therefore highly sensitive to both pruning strategies and syntactic parsing errors. To address this limitation, this paper introduces a key-path enhancement mechanism that dynamically models semantic dependencies between aspect terms and global context nodes in a continuous feature space via attention mechanisms, thereby forming a continuous "soft path".

Let the index set of aspect terms in a sentence be denoted as Φ_{asp}. The unified semantic representation of aspect terms h_{asp} is obtained via masked average pooling over aspect tokens and is used as the query anchor for subsequent attention modeling, as shown in Equation (13):

$$h_{asp} = \frac{1}{|\Phi_{asp}|} \sum_{i \in \Phi_{asp}} E_i \tag{14}$$

where $|\Phi_{asp}|$ denotes the number of aspect terms.

In contrast to traditional explicit key-path methods, which propagate information through discrete edge sequences in graph structures, the proposed approach reformulates this process as a continuous attention mechanism in feature space. Specifically, the aspect semantic anchor $h_{asp}$is used as the Query, and attention is computed over all node representations E_j, thereby establishing a global semantic shortcut between aspect terms and all words, as shown in Equations (14)–(16):

$$Q = W_Q \cdot h_{asp} \in R^{d_m} \tag{15}$$

$$K_j = W_K \cdot E_j \in R^{d_m} \tag{16}$$

$$V_j = W_V \cdot E_j \in R^{d_m} \tag{17}$$

where $W_Q, W_Q, W_Q \in R^{d_m \times d_m}$are learnable linear projection matrices, and $d_m = d_e/2$ denotes the projected feature dimension.

Subsequently, the attention weight α_j is computed to measure the implicit path strength between the aspect representation and each node j. Based on these attention weights, a weighted aggregation over value vectors is performed to obtain the key-path representation h^{key}, as shown in Equations (17)–(18):

$$\alpha_j = \frac{exp(Q \cdot K_j / \sqrt{d_m})}{\sum_{k=1}^{n} exp(Q \cdot K_k / \sqrt{d_m})} \tag{18}$$

$$h^{key} = \sum_{j=1}^{n} \alpha_j V_j \tag{19}$$

This aggregation process is equivalent to constructing a continuous semantic path originating from the aspect term, where attention weights act as edge strengths over all nodes in the sentence.

To integrate the key-path representation h^{key}into each node embedding, it is broadcast to all positions in the sequence. Meanwhile, to prevent excessive influence from the implicit path that may distort the original syntactic structure, a gating mechanism is introduced to dynamically control the enhancement magnitude g_i. Finally, each node representation is updated via residual connection and layer normalization, producing the enhanced node embedding $\widehat{E_i}$, as shown in Equations (19)–(20):

$$g_i = \sigma(W_g \cdot E_i + b_g) \tag{20}$$

$$\widehat{E_i} = LayerNorm(E_i + g_i \cdot h^{key}) \tag{21}$$

where $W_g \in R^{d_m \times 1}$ is a learnable gating parameter matrix, $\sigma(\cdot)$ is the Sigmoid activation function, and g_i indicates the extent to which node i incorporates key-path enhancement.

3.2. Semantic–syntactic collaborative interaction module

In dual-graph convolutional architectures, the syntactic GCN and semantic GCN encode structural dependency information and contextual semantic associations, respectively. These two types of information are inherently complementary: syntactic paths provide structural constraints but are limited in modeling long-range semantic dependencies, while semantic paths capture global correlations but may lack structural robustness. Therefore, to enable syntactic representations to incorporate contextual semantic relevance and semantic representations to benefit from structural guidance, this paper designs a hierarchical collaborative interaction mechanism. This mechanism performs layer-wise fusion within the graph convolution process and deep fusion at the output stage.

3.2.1. Graph convolution layer fusion module

To enable continuous interaction between the two pathways during feature propagation, this paper introduces a syntactic–semantic fusion module after each layer of GCN updates for information integration. Let the syntactic-path representations at the (l)-th GCN layer be denoted as $E_{syn}^{(l)} \in R^{B\times n\times d_m}$, and the semantic-path representations as $E_{sem}^{(l)} \in R^{B\times n\times d_m}$, where (B) denotes the batch size and $d_m = d_e/2$. First, two learnable matrices $U_1, U_2 \in R^{d_m\times d_m}$ are used to compute attention weights. The resulting attention matrix is then multiplied with the corresponding feature representations to enable information exchange, thereby updating the node representations in each GCN branch. To mitigate gradient vanishing in deep networks and stabilize training, residual connections and layer normalization are applied after each graph convolution layer. The process is formulated as follows:

$$\widehat{E}_a^{(l)} = Softmax(E_a^{(l)} \cdot U_1 \cdot (E_b^{(l)})^T) \cdot E_b^{(l)} \tag{22}$$

$$\widehat{E}_a^{(l)} = LayerNorm(\widehat{E}_a^{(l)} + \widehat{E}_a^{(l-1)}) \tag{23}$$

where $a,b \in \{sem,syn\}$ and $a \neq b$. For the first layer, $\widehat{E}_a^{(0)}$ is obtained by projecting the BERT-encoded initial representations through a linear transformation.

In this way, the syntactic pathway can absorb contextually rich semantic features from the semantic pathway, while the semantic pathway can incorporate structural constraints from the syntactic pathway. The two independent parameter matrices U_1, U_2 enable the model to learn asymmetric information flow patterns, meaning that syntactic-to-semantic guidance can differ from semantic-to-syntactic guidance.

Different depths of GCN layers capture different levels of granularity: shallow layers (e.g., ($l = 1,2$) primarily capture local syntactic patterns and short-range semantic associations, whereas deeper layers (e.g., ($l = L-1, L$) capture long-range dependencies and global structural information. To fully exploit multi-level representations, a multi-scale feature aggregation mechanism is introduced. This mechanism adaptively fuses outputs from different GCN layers for both syntactic and semantic branches, enabling the model to select features from different depths. Learnable parameters $\theta^{syn}, \theta^{sem} \in R^L$ are introduced, and normalized via Softmax to obtain layer-wise importance weights w_l^{syn}, w_l^{sem}. The final multi-scale representations $E_{syn}^{multi}, E_{sem}^{multi}$ are obtained via weighted summation, as shown in Equations (23)–(24):

$$w_l^a = \frac{exp(\theta_l^a)}{\sum_{k=1}^{L} exp(\theta_k^a)} \tag{24}$$

$$E_a^{multi} = \sum_{l=1}^{L} w_l^a \cdot \widehat{E}_a^{(l)} \tag{25}$$

where $a,b \in \{sem,syn\}, a \neq b$.

3.2.2. Multi-source output fusion module

After extracting syntactic and semantic features via the dual-graph convolutional network, how to effectively integrate these heterogeneous representations with the original BERT embeddings becomes a key factor determining sentiment classification performance. Conventional strategies such as simple concatenation or weighted summation often fail to capture higher-order interactions among features. To address this limitation, this paper proposes a multi-source output fusion module, which consists of three subcomponents: aspect-guided pooling, cross-attention fusion, and multi-source feature concatenation.

First, inspired by multi-modal and multi-task attention mechanisms, a pooling-based strategy is adopted to extract representative information from feature vectors. This preserves the main semantic content while reducing noise from redundant information. The pooled representations are further used as guidance signals to facilitate deeper interaction between syntactic and semantic branches. Given the aspect mask $M_{asp} \in \{0,1\}$, aspect-masked average pooling is applied to both syntactic GCN outputs E_{syn} and semantic GCN outputs E_{sem} , yielding intermediate representations H_{syn} and $H_{sem} \in R^{B \times d_m}$. To further enhance high-order interactions between the two pathways, a cross-attention mechanism is introduced. The graph convolution output E_b is used as the query, while the intermediate representation H_a is expanded into a sequence form as keys and values for attention computation, where a,b∈{syn,sem} and a≠b. A residual connection is then applied to mitigate information loss, followed by aspect-guided pooling to obtain the updated representation $\widetilde{E}_b \in R^{B \times d_m}$, as shown in Equations (25)–(29):

$$H_a = \frac{\sum_{i=1}^{n} M_{asp}^{i} \cdot E_a^i}{\sum_{i=1}^{n} M_{asp}^{i} + \epsilon} \tag{26}$$

$$H_b^{cross} = Attention(E_b, unsqueeze(H_a), unsqueeze(H_a)) \tag{27}$$

$$Attention(Q,K,V) = Softmax(\frac{QK^T}{\sqrt{d_m}})V \tag{28}$$

$$H_b^{residual} = H_b^{cross} + E_b \tag{29}$$

$$\widetilde{E}_b = \frac{\sum_{i=1}^{n} M_{asp}^{i} \cdot H_b^{residual}[i]}{\sum_{i=1}^{n} M_{asp}^{i} + \epsilon} \tag{30}$$

where a,b∈{syn,sem}, and ϵ is a small constant used to prevent division by zero.

Through this process, syntactic and semantic features achieve deep interaction under aspect-level guidance: syntactic representations are enhanced with contextual semantic information via semantic pooling vectors, while semantic representations are strengthened with structural constraints derived from syntactic pooling vectors.

To adaptively balance the contributions of the original and enhanced features, a gating fusion mechanism is introduced. Given the original pooled feature $H_a$and the enhanced feature $\widetilde{E}_a$, a learnable gating parameter is used to dynamically compute fusion weights, producing the final gated representation, as shown in Equations (30)–(31):

$$g_a = \sigma(W_g^a \cdot [H_a; \widetilde{E}_a]) \tag{31}$$

$$F_a = g_a \cdot \widetilde{E}_a + (1 - g_a) \cdot H_a \tag{32}$$

where a∈{syn,sem}.

Finally, to mitigate potential loss of aspect-related information caused by dual-path graph convolution and fusion operations, a multi-source concatenation strategy is adopted. Specifically, aspect-masked pooling is applied to the BERT output to obtain E_{seq}^{pooled}. The fused semantic representation F_{sem}, fused syntactic representation F_{syn}, pooled representation E_{seq}^{pooled}, and the BERT [CLS] representation E_{cls} are concatenated and passed through a linear transformation for dimensionality reduction, producing the fused feature F_{fused}. A self-attention mechanism is then applied to obtain the final representation H_{fused}, as shown in Equations (32)–(34):

$$E_{cat} = concat(F_{sem}, F_{syn}, E_{seq}^{pooled}, E_{cls}) \tag{33}$$

$$F_{fused} = MLP(E_{cat}) \tag{34}$$

$$H_{fused} = F_{fused} + Attention(F_{fused}, F_{fused}, F_{fused}) \tag{35}$$

where the MLP typically consists of two linear layers with a ReLU activation function and dropout regularization in between.

3.3. Classification layer

The fused hidden representation H_{fused} is transformed into sentiment logits and normalized via Softmax for classification:

$$z = W_{cls} \cdot H_{fused} + b_{cls} \tag{36}$$

$$\hat{y}_i = softmax(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{C} e^{z_j}} \tag{37}$$

where $W_{cls} \in R^{C \times d}$ denotes the classification weight matrix, $b_{cls} \in R^{C}$ is the bias vector, and Crepresents the number of sentiment polarity classes.

3.4. Loss function

To improve classification accuracy while simultaneously constraining the semantic adjacency matrix M^{sem} in the dual-graph convolutional network, we encourage it to approximate an orthogonal structure. Specifically, we measure the discrepancy between $M^{sem} \cdot (M^{sem})^T$ and the identity matrix I, thereby promoting sparsity and independence in the learned graph structure. This orthogonality regularization is defined in Equation (37):

$$L_{ortho} = ||M^{sem} \cdot (M^{sem})^T - I|| \tag{38}$$

Although dependency parse trees provide a strong structural inductive bias, semantic relations are typically more complex and flexible than syntactic structures. To ensure that the semantic graph convolution network captures deep semantic dependencies independent of syntactic constraints, a discrepancy loss term L_{diff} is introduced. This term encourages the model to maximize the distance between the semantic adjacency matrix M^{sem} and the syntactic adjacency matrix M^{syn}, thereby enforcing complementary representations across the two graph channels. Specifically, this is achieved by minimizing their similarity, as shown in Equation (38):

$$L_{diff} = \frac{1}{||M^{sem} - M^{syn}|| + \epsilon} \tag{39}$$

where ϵ is a small constant used to prevent division by zero.

To effectively supervise model learning and map the fused high-dimensional representations to the correct sentiment categories, we adopt cross-entropy loss as the primary optimization objective, as shown in Equation (39):

$$L_{ce} = -\sum_{i=1}^{N} y_i log(\hat{y}_i) \tag{40}$$

Finally, the overall loss function L is formulated as a weighted combination of cross-entropy loss, orthogonality regularization, and discrepancy loss, as shown in Equation (40):

$$L = L_{ce} + \alpha \cdot L_{ortho} + \beta \cdot L_{diff} \tag{41}$$

4. Experimental results and analysis

4.1. Experimental settings

The experiments were conducted on an Ubuntu 18.04 server equipped with an Intel Xeon E5-2603 v4 processor and an NVIDIA Tesla V100-PCIE GPU. The optimizer used in all experiments is Adam, with a learning rate of (2 \times 10^{-5}). The initial number of training epochs is set to 20. Early stopping is applied such that the best-performing epoch is saved when the accuracy does not improve for five consecutive epochs. The batch size is set to 16.

4.2. Experimental results and analysis

4.2.1. Results analysis

To validate the effectiveness of the proposed model for aspect-based sentiment analysis, experiments are conducted on three benchmark datasets: Restaurant, Laptop, and Twitter. The following representative baseline models are selected for comparison:

(1) BiGCN: Integrates word co-occurrence and dependency relation types using a hierarchical graph structure for aspect-based sentiment analysis.

(2) KUMA-GCN: Enhances syntactic features by introducing latent graph structures.

(3) InterGCN: Learns aspect representations with syntactic information using GCN over dependency trees.

(4) R-GAT: Proposes an aspect-oriented dependency tree and encodes it using relational graph attention networks.

(5) DGEDT: Employs a dual-channel framework combining sequential and syntactic information to improve semantic understanding.

(6) DGEDT + BERT: Replaces the BiLSTM encoder in DGEDT with a BERT-based pretrained language model.

(7) R-GAT + BERT: Incorporates BERT instead of BiLSTM in R-GAT.

(8) DualGCN + BERT: Enhances DualGCN with BERT as the encoder.

(9) Sentic GCN-BERT: Incorporates commonsense knowledge from SenticNet to enrich syntactic dependency graphs by integrating sentiment intensity into adjacency matrices.

(10) KGAN-BERT: Builds a multi-view representation learning framework that jointly extracts contextual, syntactic, and knowledge graph features through hierarchical fusion.

(11) WordTransABSA [8]: Utilizes a word-transfer language modeling strategy, replacing aspect terms with sentiment-centric tokens via masked language modeling to enhance prediction.

(12) RDGCN-BERT [9]: Introduces a reinforcement learning-based distance weighting function to optimize non-uniform decay in dependency trees.

The experimental results are summarized in Table 1.

Table 1. Experimental results of AKP-DGCN and baseline models

No.	Model	Restaurant		Laptop		Twitter	
		Accuracy	F1	Accuracy	F1	Accuracy	F1
1	BiGCN	81.97	73.48	74.59	71.84	74.16	73.35
2	kumaGCN	81.43	73.64	76.12	72.42	72.45	70.77
3	InterGCN	82.23	74.01	77.86	74.32	-	-
4	R-GAT	83.30	76.08	77.42	73.76	75.57	73.82
5	DGEDT	83.90	75.10	76.80	72.30	74.80	73.40
6	DGEDT+BERT	86.30	80.00	79.80	75.60	77.90	75.40
7	R-GAT+BERT	86.60	81.35	78.21	74.07	76.15	74.88
8	DualGCN+BERT	87.13	81.16	81.80	78.10	77.40	76.02
9	Sentic GCN-BERT	86.92	81.03	82.12	79.05	-	-
10	KGAN-BERT	87.15	82.05	82.66	78.98	-	-
11	WordTransABSA	87.3	-	79.7	-	77.4	
12	RDGCN-BERT	87.29	81.16	82.12	78.34	78.29	76.02
13	Our	87.31	81.83	83.09	79.70	78.37	76.50

Compared with traditional graph neural network models, the proposed method achieves significant performance gains over BiLSTM-based GCN models such as BiGCN, KUMA-GCN, and InterGCN. On the Laptop dataset, compared with DualGCN, the proposed model improves Accuracy and F1-score by 4.61% and 4.96%, respectively. On the Restaurant dataset, the F1-score improves by approximately 4.28%. Traditional GCN-based methods typically rely directly on raw dependency trees, which inevitably introduce irrelevant contextual nodes and noisy edges, leading to information dilution during feature aggregation. In contrast, the proposed model introduces an adaptive distance pruning mechanism that effectively filters long-range noisy dependencies and constructs a cleaner syntactic subgraph, thereby significantly improving feature extraction quality.

Compared with strong BERT-enhanced baselines such as DGEDT+BERT, R-GAT+BERT, and DualGCN+BERT, the proposed model still maintains consistent superiority. On the Restaurant dataset, it improves Accuracy and F1-score by 0.89% and 1.2% over DualGCN+BERT, respectively, while achieving a 1.6% F1 improvement on the Laptop dataset. Although these BERT-based models benefit from strong contextual representations, they typically treat syntactic and semantic features independently or combine them in a shallow manner, lacking deep interaction. In contrast, the proposed dual-graph convolutional cross-fusion mechanism enables bidirectional feature enhancement between syntactic and semantic representations via cross-attention, allowing more precise modeling of implicit associations between aspect terms and sentiment words in complex sentences.

Compared with knowledge-enhanced and structure-optimized methods, the proposed model still achieves the best performance on the Laptop dataset, surpassing KGAN-BERT by 0.72% in F1-score and outperforming RDGCN-BERT by 1.36%. While KGAN-BERT relies on external knowledge graphs, such external knowledge may introduce domain mismatch noise. In contrast, the proposed model operates purely on internal syntactic structures and enhances feature focus through a key-path enhancement strategy without requiring external resources. Moreover, although RDGCN also optimizes distance weighting, the proposed pruning strategy is more flexible and is further strengthened by the dual-graph interaction module. This demonstrates that simple weight adjustment is less effective than a combined "pruning + interaction" strategy, particularly in handling longer and more complex review sentences in the Laptop dataset, where distant noise dependencies are more prevalent.

4.2.2. Ablation study

To further validate the effectiveness of the proposed model, ablation experiments are conducted. The variants are defined as follows:

(1) w/o DP: Removing the Adaptive Distance Pruning module.(2) w/o KPE: Removing the Key-Path Enhancement module.(3) w/o DP + KPE: Removing both the Adaptive Distance Pruning and Key-Path Enhancement modules.(4) w/o linear-attn: Removing the syntactic–semantic cross fusion module.(5) w/o DP + linear-attn: Removing both the Adaptive Distance Pruning module and the syntactic–semantic cross fusion module.(6) w/o KPE + linear-attn: Removing both the Key-Path Enhancement module and the syntactic–semantic cross fusion module.

The results are reported in Table 2.

Table 2. Ablation results on the Laptop dataset

No.	model	Accuracy	F1-score
1	w/o DP	82.20	79.10
2	w/o KPE	82.40	79.30
3	w/o DP+KPE	82.00	78.90
4	w/o linear-attn	82.61	79.62
5	w/o DP+linear-attn	82.43	79.38
6	w/o KPE+linear-attn	82.60	79.29
7	Our	83.09	79.70

From the results of experiments (1), (2), and (7), it can be observed that both the Adaptive Distance Pruning module and the Key-Path Enhancement module provide clear performance gains. After removing the Adaptive Distance Pruning module, Accuracy and F1-score decrease to 82.20% and 79.10%, respectively. When the Key-Path Enhancement module is removed, the model further drops to 82.40% and 79.30%. This indicates that both modules are beneficial, with the Key-Path Enhancement module playing a more fundamental role in capturing core semantic paths within the text. When both modules are removed simultaneously, performance further declines to 82.00% and 78.90%, demonstrating that the two components are complementary and jointly contribute to more effective feature learning and improved classification performance.

From experiments (4), (5), (6), and (7), it can be seen that the syntactic–semantic cross fusion module plays a crucial role in integrating multi-source information. Compared with the full model, removing this module leads to a notable drop of 0.48% in Accuracy and 0.08 in F1-score, representing one of the most significant degradations among all single-module ablations. This highlights its central role in fusing syntactic structures with deep semantic representations. Furthermore, when the cross fusion module is removed together with either the pruning module or the key-path enhancement module, performance degradation becomes even more pronounced. This suggests that the cross fusion module does not operate independently; instead, it closely cooperates with the other two modules by leveraging refined features produced by adaptive pruning and enriched semantic paths generated by key-path enhancement, thereby forming a more discriminative representation.

A comparison between experiments (5) and (6) further reveals that the Adaptive Distance Pruning module and the Key-Path Enhancement module contribute differently when interacting with the syntactic–semantic cross fusion module. When "DP + linear-attn" is removed, the model achieves 82.43% Accuracy and 79.38% F1-score; when "KPE + linear-attn" is removed, the results are 82.60% and 79.29%, respectively. From the perspective of Accuracy, the former is 0.17 percentage points lower than the latter; however, in terms of F1-score, the former is slightly higher by 0.09. Although the differences are marginal and not fully consistent across metrics, overall results suggest that the features produced by the Adaptive Distance Pruning module are slightly more critical for the cross fusion module in this task setting, providing higher-quality inputs that may lead to a more stable and slightly stronger overall performance.

5. Conclusion and future work

This paper proposes an adaptive pruning and key-path enhanced dual-graph collaborative model for aspect-based sentiment analysis. The model first employs a BERT-based encoder to obtain deep contextual semantic representations. Then, in the syntactic-aware feature extraction module, an Adaptive Distance Pruning module and a Key-Path Enhancement module are introduced. The former dynamically removes weak long-distance dependencies to construct a cleaner sparse syntactic graph, while the latter enhances information propagation from aspect terms to opinion words via a soft-path attention mechanism. These designs effectively mitigate noise in dependency parsing and alleviate the issue of aspect information loss in deep graph networks. Next, semantic and syntactic feature extraction modules operate in parallel to capture textual information from semantic correlation and syntactic structure perspectives, respectively. The semantic–syntactic collaborative interaction module further integrates these two sources through layer-wise fusion and multi-source output fusion, producing a more comprehensive representation. Finally, the classification layer and loss function are used to predict sentiment polarity. Experimental results demonstrate that the proposed AKP-DGCN model achieves consistent improvements across evaluation metrics.

However, the current model still has limitations in semantic modeling. It primarily relies on local context and dependency structures, which limits its ability to capture implicit semantic relations between aspect terms and opinion words, such as indirect sentiment expressions and cross-sentence sentiment propagation. In addition, correlations between different sentiment categories are not fully exploited. In future work, we plan to explore large language model-based semantic distillation methods, transferring rich commonsense knowledge and deep semantic representations via soft labels, contrastive feature alignment, and aspect–opinion attention distribution transfer. This will enhance the model's ability to handle complex semantic scenarios without increasing inference cost. Furthermore, we will incorporate sentiment category correlation-aware mechanisms to further improve the accuracy and robustness of aspect-level multi-class sentiment classification.

References

[1] Haznitrama, F.G., Choi, H.J. and Chung, C.W. (2025) Methodologies and Their Comparison in Complex Compound Aspect-Based Sentiment Analysis: A Survey. *AI OPEN, 6,* 53-69.

[2] Zhang, C., Li, Q. and Song, D. (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), Association for Computational Linguistics, Hong Kong, 4568-4578.*

[3] Bai, X., Liu, P. and Zhang, Y. (2021) Investigating Typed Syntactic Dependencies for Targeted Sentiment Classification Using Graph Attention Neural Network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 29,* 503-514.

[4] Tian, Y., Chen, G. and Song, Y. (2021) Aspect-Based Sentiment Analysis with Type-Aware Graph Convolutional Networks and Layer Ensemble. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 2910-2922.*

[5] Zhou, J., Huang, J.X., Hu, Q.V. and He, L. (2020) SK-GCN: Modeling Syntax and Knowledge via Graph Convolutional Network for Aspect-Level Sentiment Classification. *Knowledge-Based Systems, 205,* 106292.

[6] Li, R., Chen, H., Feng, F., Ma, Z., Wang, X. and Hovy, E. (2021) Dual Graph Convolutional Networks for Aspect-Based Sentiment Analysis. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP), Online, 6319-6329.*

[7] Phan, H.T., Nguyen, V.D. and Nguyen, N.T. (2025) MulGCN: MultiGraph Convolutional Network for Aspect-Level Sentiment Analysis. *IEEE Access, 13,* 26304-26317.

[8] Jin, W., Zhao, B., Zhang, Y., Huang, J. and Yu, H. (2024) WordTransABSA: Enhancing Aspect-Based Sentiment Analysis with Masked Language Modeling for Affective Token Prediction. *Expert Systems with Applications, 238,* 122289.

[9] Zhao, X., Peng, H., Dai, Q., Bai, X., Peng, H., Liu, Y., Guo, Q. and Yu, P.S. (2024) RDGCN: Reinforced Dependency Graph Convolutional Network for Aspect-Based Sentiment Analysis. *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM '24), Merida, Mexico, 1-9.*