



OPEN Graph-enhanced implicit aspect-level sentiment analysis based on multi-prompt fusion

Xu Li¹✉, Xinlong Wang², Chunlong Yao² & Yang Li³

Implicit Aspect-Level Sentiment Analysis aims to identify aspect items and opinion items that do not appear in unstructured text. These items can be analyzed based on the semantics of the text. The analysis also seeks to determine the sentiment tendency of each aspect in the evaluated sentence. The difficulty of semantic comprehension is greatly increased by the fact that the evaluation texts include omitted expressions. Additionally, they often use short texts with few available features, making the analysis more challenging. To this end, this paper proposes a generative model for graph-enhanced implicit aspect-level sentiment analysis based on multi-prompt fusion. The generative pre-training model T5 is used in combination with graph neural networks to capture prompts and semantic information in the context, enabling the understanding of implicit emotions. By employing multi-prompt fusion, the model fully leverages the complementary strengths of multiple prompts, avoiding the incompleteness and instability associated with using a single prompt. In addition, to optimize model performance, this paper designs aspect term and opinion term recognition as index generation tasks. It also proposes a differentiated loss function with varying penalties for different types of errors. Compared to the existing state-of-the-art models, the proposed model improves the F1 value by 1.99% on the standard dataset restaurantACOS. Additionally, it achieves a 1.83% improvement on the LaptopACOS dataset. Ablation experiments demonstrate that each improvement is effective for aspect-level sentiment analysis. These improvements also lead to better results in low-resource settings.

Keywords Implicit aspect-level sentiment analysis, Pre-training model, Graph neural network, Multiple prompts for learning, Differentiated loss.

Early sentiment analysis usually only focuses on the overall sentiment polarity of the text, and classifies the text into simple sentiment categories such as positive, negative or neutral. This method of sentiment analysis is primarily used to determine the overall emotional tendency of a text. However, it is not possible to provide in-depth information about emotional details. Aspect-based sentiment analysis (ABSA) is designed to extract one or more specific emotional elements from a given text or sentence¹. For massive text information, such fine-grained information is difficult to extract manually. Current aspect-level sentiment analysis tasks usually involve four sentiment elements, which are aspect term (A), opinion term (O), aspect category (C), and sentiment polarity (S)².

- Aspect: An entity that indicates the object of an opinion, usually consisting of a word or phrase.
- Category: Indicates the predefined category corresponding to the object of the opinion in the research domain.
- Opinion: A description of a subjective opinion that indicates an aspect, usually consisting of a word or phrase in the text.
- Sentiment: Represents a predetermined sentiment category word (e.g., positive, negative, or neutral).

Traditional aspect-level sentiment analysis typically represents sentiment using discrete categories such as positive, negative, and neutral. In recent years, however, researchers have explored more fine-grained approaches, most notably Dimensional Sentiment Analysis, which characterizes emotions along continuous numerical scales across multiple dimensions, such as Valence, Arousal, and Dominance. As an emerging subtask of ABSA, dimensional aspect-based sentiment analysis (dimABSA) aims to model the emotional tendencies

¹Innovation and entrepreneurship education center, Dalian Polytechnic University, Dalian, Liaoning 116034, China.

²School of Information Science and Engineering, Dalian Polytechnic University, Dalian, Liaoning 116034, China.

³Dalian Cloud Power Technology Co., Dalian, Liaoning 116034, China. ✉email: lixu0319@163.com

associated with each aspect in a multi-dimensional space. For instance, in the sentence “Tuna spread is super delicious”, the aspect “Tuna spread” may have a valence score of 7.5 and an arousal score of 7.25, indicating a highly pleasant and excited emotional state. This fine-grained modeling enables dimABSA to better capture the intensity and type of sentiment expressed toward different aspects. To address this task, recent studies have proposed multi-dimensional relational models, often leveraging multi-label or multi-task learning frameworks to simultaneously learn sentiment across dimensions like valence and arousal, thereby improving prediction accuracy and real-world applicability³. Furthermore, optimizing word embeddings to better reflect sentiment dimensions has become a key strategy in dimABSA. By incorporating information from sentiment dimensions into word representations, these enhanced embeddings can more effectively capture nuanced emotional meanings, thereby improving overall sentiment classification performance⁴. In the context of Chinese sentiment analysis, several dimensional sentiment resources (e.g., Chinese Sentiment Dictionary) have been developed, which not only provide polarity annotations but also assign scores for valence and arousal. Integrating such resources into dimABSA models can significantly improve their ability to handle Chinese language texts with fine-grained sentiment distinctions⁵.

In recent years, several emerging research directions have significantly advanced the performance and generalization of ABSA models. To better capture the structural relationships between aspects and opinion words, syntactic dependency information has been widely incorporated. For example, researchers have proposed models that integrate syntactic dependency graphs with Transformer-based architectures, highlighting the crucial role of syntactic structure in identifying sentiment polarity⁶. Syntactically-guided graph convolutional networks (GCNs) have also been employed to extract structured features from dependency trees, enhancing sentiment classification accuracy⁷. Furthermore, the modeling of clause-level information has shown promising results in ABSA. Clause partitioning strategies help reduce sentiment interference between multiple aspects in a sentence, thereby leading to more precise sentiment predictions⁸. In addition, the application of contrastive learning has attracted attention for its ability to improve representation learning. For instance, some works propose semantic-aware contrastive learning frameworks that construct positive and negative sample pairs to better capture subtle semantic distinctions⁹. When combined with context enhancement techniques, contrastive learning has proven effective in low-resource scenarios by boosting the model’s generalization ability¹⁰.

However, in the actual scene, there are a large number of comment sentences without explicit emotional words, but they can still express emotions clearly¹¹. For example, in the sentence “The service attitude is really bad”, although there is no explicitly mentioned aspect term, it can be inferred based on commonsense reasoning that the implicit aspect is “service”, with a negative sentiment. In another example, “I was attracted by the strong fragrance as soon as I entered the door”, although there is no explicit opinion word, it conveys a positive sentiment about the “environmental smell” through metaphorical expression. These examples illustrate the challenges posed by the implicit expression of aspects and opinions in implicit ABSA tasks, highlighting the importance of semantic reasoning in such scenarios. For a computer, the lack of emotional words will make it more difficult to understand the meaning. Aspect extraction is usually categorized into explicit versus implicit aspects according to the form in which the object of the review is presented. Explicit aspects are the objects of evaluation that are explicitly mentioned in the text of the review. Implicit aspects are usually not directly specified and are the objects of comments implicit in the semantics of the comment text. To distinguish it from explicit aspect-level sentiment analysis, which involves clear emotion words, sentiment analysis for texts containing implicit aspects and ambiguous emotion words is needed. This type of analysis is referred to as implicit aspect-level sentiment analysis. This article uses it and null tags to identify implied aspects or points of view. The quadruples corresponding to the above two statements are “it-food quality-Yum-positive”, “it-restaurant general-null-positive”. As shown in Fig. 1.

Implicit aspect-level sentiment analysis plays an important role in enriching sentiment analysis applications and enhancing business intelligence. It also improves user experience and advances academic research. By capturing the point of view in comments more precisely, it provides users, businesses, and researchers with more comprehensive, accurate, and valuable sentiment analysis results¹². Compared to explicit emotions, however, there are two main difficulties in the study of implicit emotions. One is the lack of explicit emotion words, resulting in semantic features that are not easily recognizable; Second, implicit sentiment analysis usually requires deeper contextual understanding¹³.

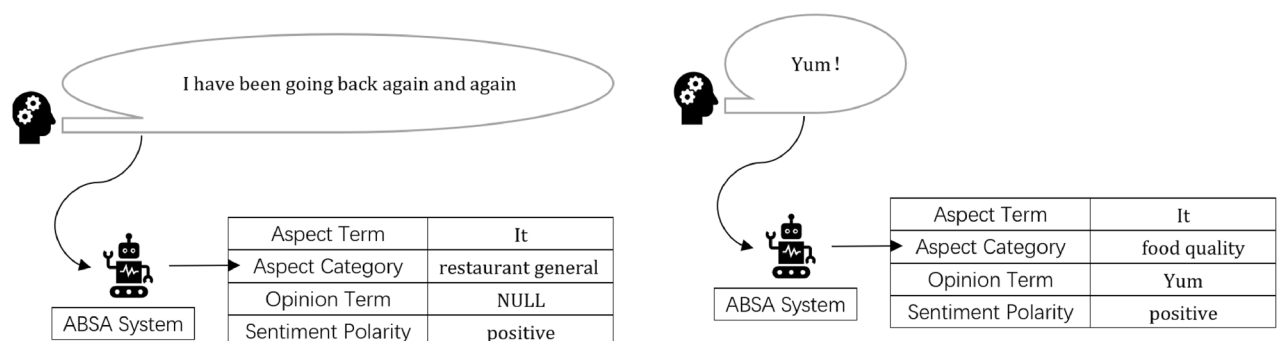


Fig. 1. The four sentiment elements in the implicit aspect-level sentiment analysis task.

To improve on the lack of implicit emotion, the researchers applied the generative model to the ABSA task. Generative models can utilize contextual information to better understand and generate natural language text, bringing it closer to human expression¹⁴. With the rapid development of large language models (LLMs) such as BERT, GPT, and T5(Text-to-Text Transfer Transformer), ABSA research has begun to benefit from their powerful pre-trained language understanding capabilities. Recent studies have fine-tuned LLMs for sentiment extraction or adapted these models to ABSA tasks using cue-based learning. These methods provide a new paradigm for end-to-end aspect and sentiment modeling, especially in zero-shot or few-shot settings¹⁵. In this paper, we use the T5 model developed by Google Research to generate target sequences in the end-to-end process¹⁶. Traditional generative models select the text with the highest probability from the word list based on probabilistic predictions. This requires the model to choose from a large vocabulary, often containing thousands or tens of thousands of words, which increases the complexity of training and reasoning due to the huge search space. In order to reduce the search space and improve accuracy, this paper introduces pointer indexing, transforming aspect and opinion items into indexed generation questions. Index generation typically requires less computational cost, as the model only needs to generate an index of the tags associated with the task rather than a complete sequence of text. This reduces the output space from the size of the vocabulary to the number of task-related tags, this greatly reduces the complexity of the task and increases the efficiency of generation.

In aspect-level sentiment analysis, traditional prompt learning methods typically generate sequences of emotion elements in a fixed left-to-right order. This approach ignores the effects of the interdependence of elements in the emotion tuple and the diversity of objects of linguistic expression. For example, Design the prompt template in the order of “a \Rightarrow c \Rightarrow o \Rightarrow s” in the quadruples¹⁷. This single-order generation approach has the following potential drawbacks: (1) Incompleteness: tuple prediction is not a natural text generation task, and the relationships between elements are not linear but interdependent; (2) Instability: significant differences in performance for different target template sequences¹⁸; (3) Accumulation of errors: errors in earlier forecasts can accumulate and affect the accuracy of subsequent forecasts. To solve this problem, this paper uses a multi-sequential prompt learning approach that aggregates sentiment elements predicted in different orders. Inspired by prompt links¹⁹, the method introduces element-order-based prompt learning to control the prediction order of emotional elements. This approach enables the diversity of target expressions. In contrast to fixed-order prompting, multiple-order prompts mitigate the incompleteness and instability of fixed sequences by receiving information from multiple sequential prompts. At the same time, the potential error accumulation of the generation method is alleviated through the arrangement of elements.

In addition, the traditional T5 model mainly handles text information through sequence modeling and relies on the self-attention mechanism of serialization. This approach is difficult for capturing complex relationships in the text and may miss some global information fusion. On the other hand, graph neural networks (GNN) can propagate information between nodes, enabling the model to integrate contextual information throughout the graph. Since the representation of each node is informed by its neighbors, a more comprehensive understanding of the context becomes possible. Therefore, GNN combined with the T5 model can better understand the complex relationships in the text context. It can also effectively capture the complex dependency relationships between words in node sentences. This improves the model's sensitivity and understanding of the context, leading to more accurate and context-aware predictions²⁰.

Most existing sentiment analysis models use cross-entropy loss. However, in the three-class sentiment task, the distance between different sentiment labels varies, so they cannot be treated equally. Implicit sentiment, on the other hand, do not contain obvious emotional words and are mostly objective facts. So, in implicit sentiment categorization, this difference needs to be emphasized. To address this issue, this paper proposes a differentiated cross-entropy loss function, which builds upon the conventional cross-entropy loss by incorporating a discriminative constraint on different prediction outcomes. This enhancement enables the model to more precisely distinguish subtle differences between sentiment categories during training. Experiments on two benchmark datasets, restaurantACOS and LaptopACOS, validate the feasibility and effectiveness of the method proposed in this paper.

The contributions of this paper are as follows:

- (1) In this paper, we frame aspect-level sentiment analysis as an index generation and category prediction problem. We also employ an element-order-based multi-prompt learning approach to improve the accuracy of sentiment tuple prediction.
- (2) Combining graph neural networks with the T5 model effectively captures the complex dependencies between words in a sentence. This integration, in turn, enhances the performance of the model.
- (3) A differentiated loss function is proposed to enhance the model's ability to discriminate between different prediction results. This helps optimize the overall model performance.
- (4) Experiments show that the proposed method improves the F1 values by 1.99% and 1.83% on the benchmark datasets restaurantACOS and LaptopACOS, respectively. It is also highly effective in low-resource settings. The model is highly generalizable and is not only applicable to implicit aspect-level sentiment texts, but also works well for explicit aspect-level sentiment texts.

Related works

Research methodology for aspect level sentiment analysis

In the aspect-level sentiment analysis study, Initially the pipeline approach was taken²¹. First, aspects relevant to sentiment analysis are extracted from the text, and then for each extracted aspect, the sentiment tendency of the text describing that aspect is identified (positive, negative, or neutral)²². This tandem separate treatment is simpler and the individual modules are more flexible. However, this approach neglects the link between the two

tasks of aspect extraction and sentiment categorization. The results extracted from the prior evaluation aspect can affect the results of sentiment categorization, which is prone to error accumulation before and after.

To address the shortcomings of the tandem approach, the researchers proposed a unified labeling approach²³. This method designs a category label that includes both the position of aspect words and sentiment polarity. It transforms the two sequential tasks of aspect extraction and sentiment classification into a joint task based on sequence labeling. Typically, the designed category labels are: { B-P, B-O, B-N, I-P, I-O, I-N, O }. Here, B-P, B-O, and B-N represent the beginning of positive, neutral, and negative evaluation aspects, while I-P, I-O, and I-N denote the middle part of positive, neutral, and negative evaluation aspects, respectively. O is used to mark irrelevant characters. The categorization labels in this method contain both aspect words and sentiment information. Thus, the joint annotation of evaluation aspects and sentiment classification can be achieved by classifying each element of a linear sequence according to its contextual content. Each element is assigned one of the labels from the label set, leading to the sequence prediction results. The unified annotation model handles explicit aspect-level sentiment analysis well. However, in implicit aspect-level sentiment analysis, aspect items are not directly mentioned in the text but are implied through descriptions or modifiers. As a result, this method cannot effectively label the aspect items of these implicit emotions and performs poorly in processing them.

The machine reading comprehension (MRC) framework is a novel approach for improving natural language processing tasks. It can effectively address the shortcomings of the uniform annotation model, particularly its inability to annotate implicit aspect items. The model first generates two questions: one about aspect items in the text (e.g., “What product aspect is mentioned in this text?”), and the other about sentiment tendencies (e.g., “What sentiment about the product aspect is expressed in this text?”). For each generated question, the model searches through the text and generates the corresponding answer. The bi-directional MRC framework not only generates questions to guide the extraction of answers, but also uses the extracted answers to further optimize the question generation²⁴. Although the framework solves the problem that the unified annotation model is unable to annotate implicit aspect items, the model may still struggle to accurately identify sentiment tendencies in certain complex contexts. This is especially true when the implicit sentiment involves multilevel semantic relations. Additionally, the bidirectional MRC framework requires searching and localizing each question in the text, which results in high computational complexity.

With the wide use of sequence-to-sequence Models, Generative Models have gradually become the focus of aspect-level sentiment analysis research. Instead of relying on a preset question and answer format, the generative model generates the target output once. This avoids the complex operation of multiple question-answering and significantly reduces computational overhead. Through pre-training and multi-level semantic modeling, the generative model can better understand complex contexts, generate outputs containing implicit emotional information, and adapt to a wider range of application scenarios. Especially when using large pre-trained models like T5 and GPT, they perform even better at handling complex semantic and multi-level reasoning tasks. Hang et al. used the pre-trained model BART to solve seven subtasks of ABSA, and the framework achieved substantial performance gains²⁵.

T5 is a powerful Natural Language Processing (NLP) model based on the Transformer architecture. It employs a text-to-text training approach that simplifies task definition and processing. The T5 model includes multiple encoder and decoder layers, which makes it very powerful and flexible. T5's pre-trained tasks are text-to-text tasks. When fine-tuned on a specific task, the T5 model adjusts the input and output formats according to the task's requirements, allowing it to seamlessly adapt to multiple tasks without introducing additional task-specific model designs.

To solve the aspect-level sentiment analysis quadruple extraction task, Hongjie et al.²⁶ constructed two benchmark datasets, restaurantACOS and LaptopACOS. The RestaurantACOS dataset primarily contains customer feedback in the restaurant domain, with predefined aspect categories such as food, service, atmosphere, and price. The LaptopACOS dataset focuses on customer feedback in the laptop domain, including aspect categories like battery, design, performance, screen quality, and cost performance. These two datasets provide rich user comments annotated with specific aspect and sentiment labels. They enable researchers to build and evaluate various sentiment analysis models, particularly those that handle implicit aspects and sentiment aspects. Through these datasets, researchers can better understand and parse the sentiment information in the text, and improve the accuracy and practicality of sentiment analysis technology.

Prompt learning

Prompt learning helps to infer implicit sentiment based on text context. By designing appropriate prompt templates, the model can better capture prompts and semantic information in the context. This ensures that the model can effectively capture and process implicit emotions²⁷. With the development of prompt learning, researchers have proposed to use the Chain of Thought (CoT) method to chain reasoning about implicit emotions. This method is based on a pre-trained model, and designs three kinds of prompts to realize three reasoning steps, respectively inferring: (1) the fine-grained aspect of a given goal, (2) the underlying perspective on that aspect, and (3) the ultimate polarity²⁸. Through this progressive reasoning from easy to difficult, the hidden elements of the complete sentiment map can be gradually revealed. This approach makes polarity prediction easier and more accurate, effectively alleviating the prediction difficulty of implicit aspect-level sentiment analysis tasks.

Methodology

Problem definition

The aspect-level sentiment analysis quadruple task is formalized below:

- Input: $X = [x_1, \dots, x_i, \dots, x_n]$, $i \in (1, n)$ where X is an input instance. n is the number of tokens in the instance, x_i denotes the i th token in the input instance.

- Output: $Y = [a_1^b, a_1^e, o_1^b, o_1^e, c_1, s_1, \dots, a_i^b, a_i^e, o_i^b, o_i^e, c_i, s_i, \dots, a_m^b, a_m^e, o_m^b, o_m^e, c_m, s_m]$, $i \in (1, m)$ Among them: a_i^b and a_i^e denote the start index and end index of the i th aspect item respectively, and the start index and end index of the hidden aspect item are denoted by 0. o_i^b and o_i^e denote the starting index and the ending index of the opinion item corresponding to the i th aspect item respectively, and the starting index and ending index of the implied opinion item are also represented by 0. Let c_i denote the aspect category to which the i -th aspect item belongs, $c_i \in C$, C is a predefined category; s_i denotes the sentiment tendency corresponding to the i th aspect item, $s_i \in \{POS, NEG, NEU\}$, where POS, NEU and NEG denote positive, neutral and negative sentiment, respectively.

The goal of aspect-level sentiment analysis model is to automatically learn a mapping function $f: X \rightarrow Y$, so that for any input X on the test data, it can generate a sequence of tuples describing the mention object and expressing the sentiment of X .

For example, for the input instance “I’d highly recommend it for a special occasion, it provides an intimate setting and nice service,” the opinion of “setting” is “intimate” and the emotional tendency expressed is positive. The opinion of “service” is “nice” and the emotional tendency expressed is also positive, but the aspect item to which the opinion item “recommend” belongs is not directly mentioned in the sentence. However, based on the context semantics, it can be inferred that the evaluation object belongs to the aspect category “restaurant miscellaneous,” and the sentiment tendency expressed is positive. The tuples output by the model are shown in Fig. 2.

Model structure

The quadruple task in this paper is further decomposed into index generation and category prediction. Index generation is to generate the aspect and opinion items of the corresponding quadruple in the target utterance in the form of an index. Category prediction means that the aspect categories and sentiment polarity of the quadruple are predicted according to the predefined categories, and the model structure is shown in Fig. 3.

In the figure, the ‘Index Generator’ converts the index value to the corresponding tokens in the original text. If the index of the opinion item is 0, the corresponding token is “It”; if the index of the aspect item is 0, the corresponding token is “NULL.” To obtain better generation performance, this paper uses relative position encoding. The GNN module is introduced at the output of the encoder, and the output features of the T5 encoder are weighted and fused with the output features of the GNN module to obtain the final output. The proposed model consists of two parts: the encoder and the decoder.

Combining encoders for graph neural networks

T5 encoders

The T5 encoder is based on the standard Transformer architecture and its main task is to encode the input sequence into feature vectors. These feature vectors contain the semantic and contextual information of the input. They are fused with the features of the GNN output and passed to the decoder.

First, the words in the input sequence are converted into word embedding representations. The input embeddings are combined with positional encoding to contain the positional information of the words as shown in Eq. (1).

$$E(x) = \text{Embedding}(x) + \text{PositionalEncoding} \tag{1}$$

$\text{Embedding}(x)$ is the word embedding, and $\text{PositionalEncoding}$ is the positional encoding that provides positional information. This encoding helps capture the sequential relationship of the input sequence.

Second, the transformed word vectors pass through a self-attention layer in the encoder. This enables the model to compute, for each position, the correlation between that position and all other positions, capturing the dependencies between the individual words in the sequence. In the encoder, self-attention is usually computed using a multi-head attention mechanism, which performs the attention computation from different subspaces in parallel. The features processed through the self-attention mechanism are then passed through a feed-forward neural network to nonlinearly transform the features. In each sublayer of the encoder (self-attention and feed-forward network), the T5 encoder employs residual concatenation with normalization. This approach makes the model training more stable and helps prevent the gradient from vanishing or exploding.

Finally, the encoder combines the results of all levels of computation to generate the output of the encoder, as shown in Eq. (2).

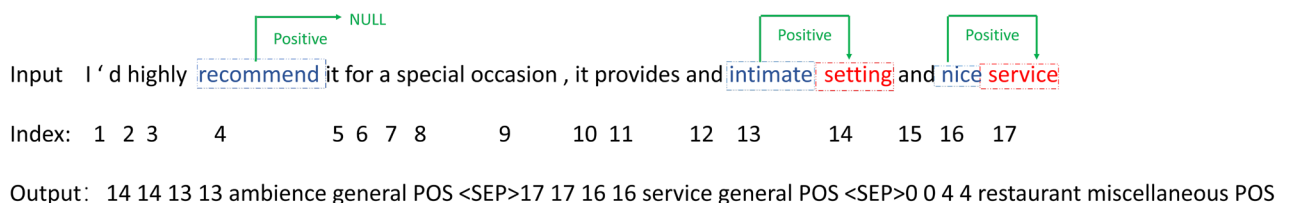


Fig. 2. Sequence of tuples output by the model.

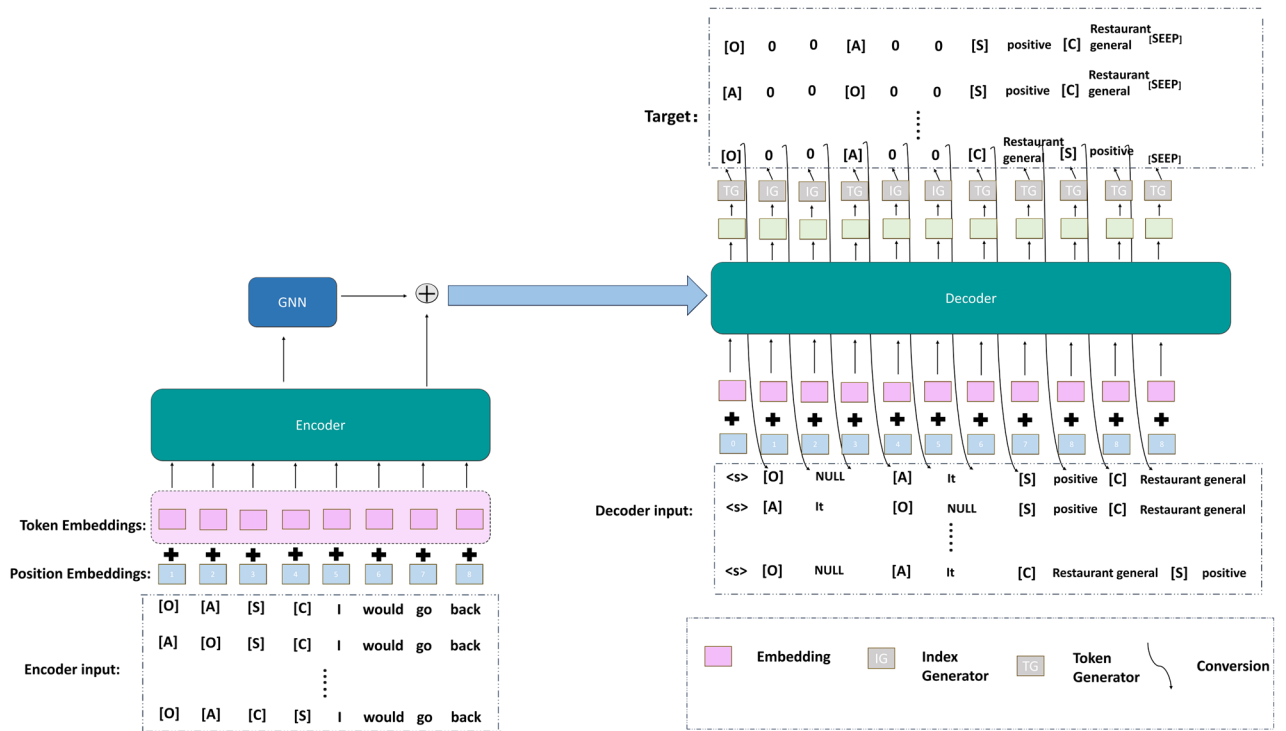


Fig. 3. General architecture of the model.

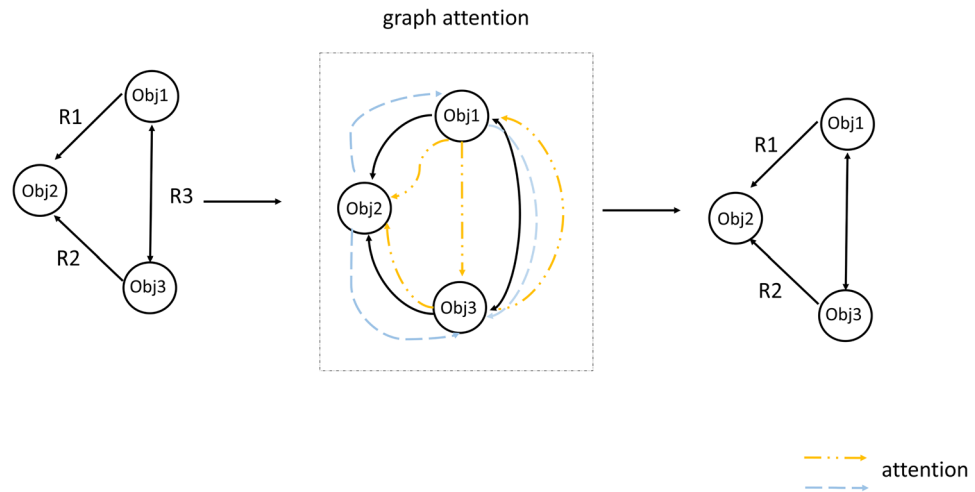


Fig. 4. AttentionGCN model.

$$H^e = Encoder(E(x)) \tag{2}$$

GNN

Graph neural networks are a class of neural network models that are capable of learning on graph-structured data. They can capture the relationships and structural information between nodes in a graph through the propagation mechanism of nodes and edges. In this paper, we design a graph convolutional network (AttentionGCN) that incorporates the attention mechanism, as shown in Fig. 4. Specifically, the outputs of the T5 encoder are regarded as the node features of the graph, and edges between the nodes are constructed based on word-to-word dependencies. AttentionGCN is updated and aggregated by the node features in local neighborhoods through a message-passing mechanism. After several layers of Graph Convolutional Neural Network (GCN) processing, the model obtains node representations that contain rich structural information. These representations are then fused with the output of the T5 encoder and used as input to the T5 decoder.

The AttentionGCN module proposed in this paper consists of two layers of graph convolutional neural network. Additionally, it includes one layer of graph attention network (GAT). Every convolutional layer in the GCN processes only the first-order neighborhood information. Multi-order neighborhood information transfer can be achieved by stacking multiple convolutional layers. The GAT introduces the attention mechanism, which dynamically adjusts the weights based on the similarity between nodes. This effectively enhances the model's expressive power and flexibility.

In the AttentionGCN module, the feature H^e output from the encoder is used to extract preliminary graph structure information. This is achieved through the first layer of graph convolution, as shown in Eq. (3).

$$H^{(1)} = \sigma(\tilde{A}H^eW^{(0)}) \quad (3)$$

where \tilde{A} is the normalized adjacency matrix, $H^e \in \mathbb{R}^{N \times F}$ is the input matrix, N is the number of nodes, F is the feature dimension, $W^{(0)}$ is the weight matrix of the first GCN layer, and σ is the Relu activation function.

Then, the node features are further processed using graph attention network to capture the complex relationships among nodes through the multi-head attention mechanism as shown in Eq. (4).

$$H^{(2)} = \sigma\left(\sum_H^{h=1} a_{ij}^{(h)} W^{(h)} H^{(1)}\right) \quad (4)$$

where $H^{(1)} \in \mathbb{R}^{N \times F'}$ is the output feature matrix of the first GCN layer, F' is the output feature dimension of the first GCN layer, H is the number of attention heads, $W^{(h)}$ is the weight matrix of the ht-h attention head, and $a_{ij}^{(h)}$ is the attention weight of the h-th attention head.

Finally, the higher-order graph structure information is extracted after the second layer of graph convolution, as shown in Eq. (5).

$$H^{(3)} = \tilde{A}H^{(2)}W^{(1)} \quad (5)$$

where $H^{(2)} \in \mathbb{R}^{N \times (H \cdot F')}$ is the output feature matrix of the attention mechanism layer and $W^{(1)}$ is the weight matrix of the second layer of graph convolution.

The final output of the entire AttentionGCN module is shown in Eq. (6).

$$GNN_H^e = AttentionGCN(H^e) \quad (6)$$

where H^e is the feature vector of the T5 encoder output.

Feature fusion

In this paper, a gated fusion module (GatedFusion) is designed with the aim of effectively fusing the information from the T5 encoder and the graph neural network. Firstly, the outputs of the T5 encoder and AttentionGCN are concatenated to form the combined features. Then, the gating weights are calculated using a linear transformation followed by the Sigmoid activation function, and finally, the combined features are weighted and fused using these gating weights to produce the final output. Thus, the final encoder output can be expressed as follows (7).

$$fused_output = weights * H^e + (1 - weights) * GNN_H^e \quad (7)$$

Decoder

The decoder's task is to evaluate the different candidate outputs during the generation process. It selects the output that best fits the target task. This process involves calculating the probabilities for the decoder outputs. It also includes the scoring mechanism and the final output selection.

The decoder generates at each time step t based on the previously generated word sequence $y_{<t}$ and the feature $fused_output$ passed from the encoder as shown in Eq. (8).

$$P(y_{<t}, y_t, h) = softmax(W \cdot Decoder(y_{<t}, fused_output)) \quad (8)$$

where W is the linear transformation matrix of the decoder and the *softmax* function transforms the feature vectors into probability distributions.

The decoder compares the probabilities of all candidate words. It selects the word with the highest probability as the output of the current time step, as shown in Eq. (9).

$$y_t = argmax P(y|y_{<t}, h) \quad (9)$$

The probability calculation of the decoder is shown in Fig. 5.

The sentence input in the figure is "I would go back." In this sentence, there is neither an explicit aspect item nor an explicit opinion item. Therefore, NULL with the maximum probability is selected as the aspect item and opinion item of the statement, and the corresponding index value is 0. The aspect category with the highest probability is Restaurant general, and the sentiment polarity with the highest probability is positive. The output of the decoder is shown in Eq. (10).

$$h_t^d = Decoder(fused_output; \hat{Y}_{<t}) \quad (10)$$

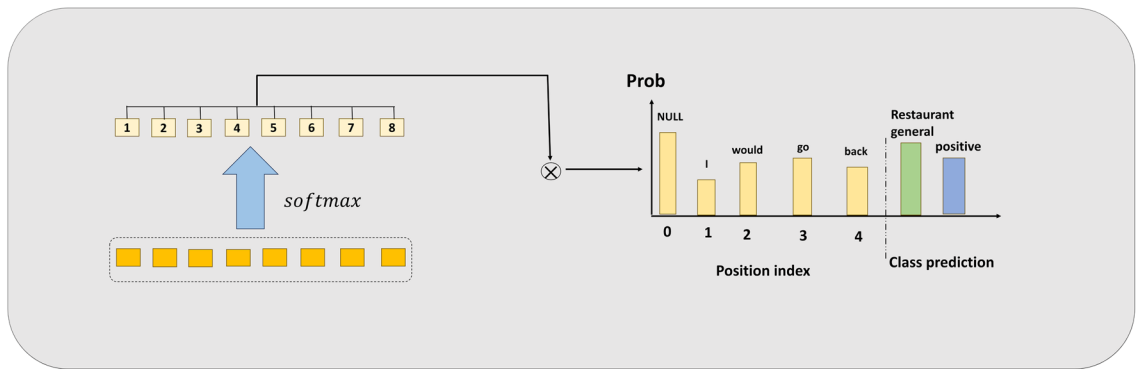


Fig. 5. Probability calculation.

where $h_t^d \in R^d, d$ is the dimension of the hidden layer, an $\hat{Y}_{<t}$ is the output of the decoder for the first t time steps.

The decoding algorithm for the implicit aspect-level sentiment analysis quadruple task in this paper is as follows 1:

Input: $X = \{([A][O][C][S] + x), \dots, ([S][C][A][O] + x)\}$ target sequence: $Y = [y_1, \dots, y_m]$

Output: $L = \{([A] a_1^b, a_1^e, [O] o_1^b, o_1^e, [C] c_1, [S] s_1 [SEP] \dots [A] a_i^b, a_i^e, [O] o_i^b, o_i^e, [C] c_i, [S] s_i), \dots, ([S] s_1, [C] c_1, [A] a_1^b, a_1^e, [O] o_1^b, o_1^e [SEP] \dots [S] s_i, [C] c_i, [A] a_i^b, a_i^e, [O] o_i^b, o_i^e)\}$

- 1: $L = \{ \}, i = 1, j = 1, n = \text{len}(X)$
- 2: **while** $i \leq n$ **do**
- 3: $H^e = \text{T5Encoder}(X[i][j])$
- 4: $\bar{H}^e = \text{GNN}(H^e)$
- 5: $\hat{H}^e = \text{gate_weights} * H^e + (1 - \text{gate_weights}) * \bar{H}^e$
- 6: $h_1^d = \text{T5Decoder}(\hat{H}^e)$
- 7: $e = \{h_1^d\}$
- 8: **while** $j \leq m$ **do**
- 9: $\text{decoder_output_text} = \text{index_to_text}(h_j^d)$
- 10: $H^e = \text{T5Encoder}(X[i][j + 1])$
- 11: $\bar{H}^e = \text{GNN}(H^e)$
- 12: $\hat{H}^e = \text{gate_weights} * H^e + (1 - \text{gate_weights}) * \bar{H}^e$
- 13: $h_{j+1}^d = \text{T5Decoder}(\hat{H}^e ; \text{embedding}(\text{decoder_output_text}))$
- 14: $e.\text{add}(h_{j+1}^d)$
- 15: $j += 1$
- 16: **end while**
- 17: $L.\text{add}(e)$
- 18: $i += 1$
- 19: **end while**
- 20: **return** L

Algorithm 1. Decoding process for quadruple extraction task

Multi-prompt fusion

In order to improve the performance of the model on the aspect-level sentiment analysis quadruple task, in this paper, we use the template order enhancement method²⁹, which uses [A], [C], [O], and [S] as prompt prefixes.

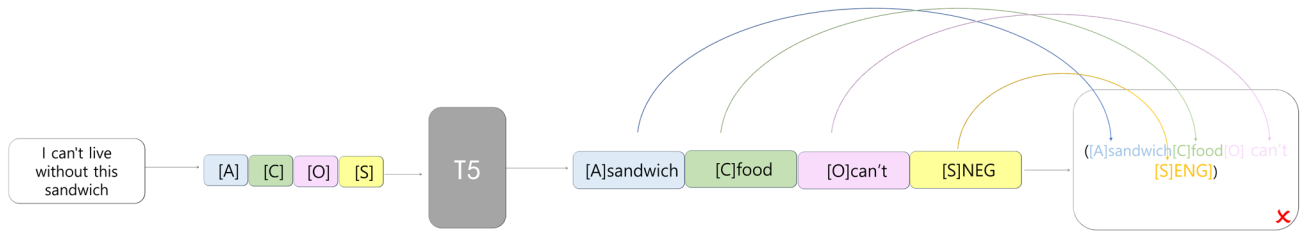


Fig. 6. Fixed- sequential prediction.

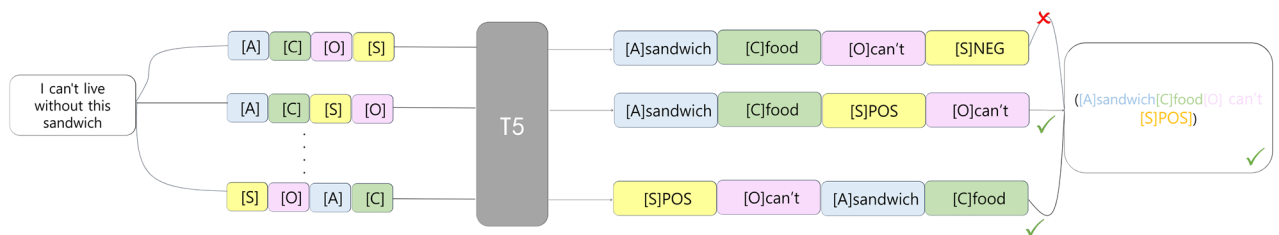


Fig. 7. Multi-sequential prediction.

The traditional method relies on a single fixed order for prediction, which learns the information in only one particular order¹⁷, as shown in Fig. 6.

This article finds that generating new training examples by changing the order of the prompts increases the diversity of the training data without altering the underlying semantics, as shown in Fig. 7. By using prompts in different orders, the model can generate multiple tuples based on the varied prompt sequences.

Since the computational overhead increases linearly with the number of different prompting sequences, this paper requires the selection of an appropriate number of prompting templates¹⁸. (1) First, by arranging and combining the four prompt elements [A], [C], [O] and [S], there are a total of 24 prompt orders, and all possible permutations p_i are used as candidates (2) For each generated prompt order, this paper combines the current input sentence with the prompt order to form the input data for the model. Meanwhile, the corresponding target quadruple of the model is constructed based on the arrangement of the prompt order. These input-output pairs are used to evaluate the performance of the model under different prompting orders. (3) In this paper, the pre-training model T5 is utilized to train the encoded inputs and targets, and the entropy (entropy) of the model is calculated for each prompting order. Entropy is used as a measure of the uncertainty of the model's outputs, and lower entropy values indicate that the model performs better in that prompting order. (4) In order to find the best prompting order globally, the average score of arrangement p_i on the training set D is calculated as shown in Eq. (11):

$$S_{P_i} = \frac{\sum_D P(y_{p_i}|x)}{|D|} \tag{11}$$

Finally, by ranking the cumulative entropy value of each prompt order, the top m permutations with the lowest entropy value are selected as the prompts in this paper. The proposed multi-prompt fusion method not only improves the prediction accuracy of the model, but also significantly reduces the sensitivity to the prompt order, which enables the model to have better generalization ability in different scenarios.

Since different cue orders do not predict the same results for different sentences, some orders yield the same correct tuple while others are less effective and may produce incorrect results. However, correctly predicted sentiment polarity and aspect categories under the same aspect word typically occur more frequently than incorrect predictions. In this paper, we first summarize the results for all sequences and then use the “majority rule” as the final prediction. Specifically, for an input sentence x , assuming this paper generates m results of different orders, the set of prediction tuples of order p_i is T_{p_i} , which may contain one or more sentiment tuples, allowing us to obtain the final aggregated result T using the following formula 12.

$$T = \left\{ t | t \in \bigcup_{i=1}^m T_{p_i} \text{ and } (\sum_{i=1}^m T'_{p_i}(t) \geq \frac{m}{2}) \right\} \tag{12}$$

The prompt selection and prediction algorithm of multi-prompt fusion is as follows (2):

```

1: elements = ["[A]", "[C]", "[O]", "[S]"]
2: all_orders = permutations(elements) # 24 prompt orders
3: for order in orders do
4:     for sentence in training_set do
5:         model_input = combine_order_and_sentence(each_order, sentence)
6:         target_output = build_target_quad(each_order, sentence)
7:         entropy_value = calculate_entropy(T5_model, model_input, target_output) # Train using the T5 model and calculate
           the entropy values.
8:         save_entropy(each_order, entropy_value)
9:     end for
10: sorted_orders = sort_by_entropy(all_orders)
11: best_orders = sorted_orders[:m] # Select the top m permutations with the lowest entropy values as prompts.
12: end for
13: for sentence in input_data do
14:     predictions = []
15:     for each_order in best_orders do
16:         prediction = predict_with_order(T5_model, sentence, each_order)
17:         predictions.append(prediction)
18:     end for
19:     final_result = majority_vote (predictions) # Aggregate the results using the 'majority voting' principle.
20: end for
21: return final_result

```

Algorithm 2. Prompt selection and prediction for multi-prompt fusion

Differentiated loss function

The cross-entropy loss function is often used to measure the difference between the model's predicted value and the actual target value, represented as a scalar. This function compares the output probability distribution of the model to the true target distribution, with the negative log-likelihood frequently used to quantify this difference, as shown in Eq. (13):

$$LCE(x, y) = - \sum_{t=1}^T \log_p(y_t | x, y < t) \quad (13)$$

where T is the length of the target sequence y and $y < t$ denotes the previously generated token.

In traditional sentiment categorization efforts, the positive, negative, and neutral sentiment categories are usually treated equally. However, for sentiment analysis tasks, the distinction between positive and negative sentiments is often greater than that between positive and neutral or negative and neutral. This means that if the model misclassifies positive text as negative sentiment, the model should give a heavier penalty for misclassifying it as neutral. This phenomenon needs to be especially considered and addressed in the context of implicit emotions. For implicit emotions, sentences often describe objective events without clear emotional words, making emotional characteristics less apparent. This results in a more blurred distinction between positive, negative, and neutral texts, making it more challenging to differentiate between neutral and positive-negative emotions. To this end, this paper proposes a class differentiated loss function, which is calculated as follows (14):

$$L_{diff} = \begin{cases} \sum_{i=0 \& i \neq y}^{C-1} \max(0, m_b - p(\hat{y})[y] + p(\hat{y})[i], \text{if } (|y - i| > 1)) \\ \sum_{i=0 \& i \neq y}^{C-1} \max(0, m_s - p(\hat{y})[y] + p(\hat{y})[i], \text{if } (|y - i| \leq 1)) \end{cases} \quad (14)$$

where, C is the total number of categories, C = 3 for sentiment triple classification, and in the training data of this paper positive category labels are 0, neutral is 1, and negative is 2. $p(\hat{y})$ is the predicted category scores, and y is the true category labels, where $0 \leq y, i \leq C - 1$, $P(\hat{y})[y]$ stands for predicted scores of the category y, and $P(\hat{y})[i]$ represents the predicted scores of the category except m_s and m_b represent two weight values differentiated for different labels, respectively. The aim of this paper is to make positive and negative more distant than positive and neutral. Therefore a large and a small weight value m_s and m_b are used to control the different cases of

$|y - i|$. If $|y - i| > 1$, this paper adopts a larger weight value, resulting in a larger penalty term. If $|y - i| \leq 1$, a smaller weight value is used. The final differentiated cross-entropy loss function in this paper is (15):

$$L = \alpha LCE + \beta Ldiff \quad (15)$$

Experiments

Dataset

This paper evaluates the proposed model using Restaurant and Laptop datasets. This dataset focuses on implicit aspects and opinions that help to fully measure the methodology of this paper. The dataset is labeled with four aspects: aspect item, aspect category, opinion item, and sentiment polarity. The predefined aspect category of Restaurant has 13 categories, including “food prices”, “food quality”, “food general”, etc. The predefined aspect category of Laptop has 121 categories, including “laptop usability”, “shipping quality”, “keyboard general”, etc. The specific details are given in Table 1.

Denotes the number of corresponding elements. EA, EO, IA, and IO denote explicit aspect, explicit opinion, implicit aspect, and implicit opinion, respectively.

Experimental environment and parameter settings

The experimental environment is python3.8, PyTorch2.0.0, Cuda11.8, and the graphics card RTX3090 is used for model training. In this paper, the T5-BASE model from Huggingface Transformers library is used as the pre-trained model³⁰. T5-BASE consists of 12 layers of encoder and 12 layers of decoder stacked, with a hidden layer dimension of 768. The training configuration includes setting the epochs to 20, the maximum sentence length to 200, the batch size to 16, and the initial learning rate to $1e-4$. The results of the first 15 different cue outputs were selected in the multi-cue fusion. The results of the top 15 different prompt outputs were selected for multi-prompt fusion. m_s and m_b in the differentiated loss function are 0.5 and 0.2, and α and β are 0.8 and 0.2, respectively.

Evaluation index

Precision, recall and F1 score are used to measure the performance of the model. A prediction was considered correct only if the predicted ranges of the aspect and opinion items matched exactly with the start and end boundaries. Additionally, the sentiment polarity and aspect category predictions also had to be in perfect agreement. Precision represents the proportion of correctly predicted quadruples among all the quadruples predicted by the model and is calculated as shown in Eq. (16):

$$Precision = \frac{TP}{TP + FP} \quad (16)$$

where, TP is the number of quadruples for which the sample fully conforms to the label; FP is the number of quadruples for which the sample prediction does not conform to the label.

Recall represents the proportion of correctly predicted quadruples among all quadruples and is calculated as shown in Eq. (17):

$$Recall = \frac{TP}{TP + FN} \quad (17)$$

	Restaurant-ACOS	Laptop-ACOS
Categories	13	121
Sentences	2286	4076
Quads	3661	5773
EA & EO	2429 (66.40%)	3269 (56.77%)
IA & EO	530 (14.49%)	910 (15.80%)
EA & IO	350 (9.57%)	1237 (21.48%)
IA & IO	349 (9.54%)	342 (5.94%)
POS	2503	3578
NEU	151	316
NEG	1007	1879
Train	1531	2934
Dev	170	326
Test	585	816
Train (Quads)	2484	4172
Dev (Quads)	261	440
Test (Quads)	916	1161

Table 1. Data statistics for the ACOS-Datase.

	Methods	Precision (%)	Recall (%)	F1 (%)
Extractive methods	DP-ACOS	34.67	15.08	21.04
	Extract-Classify-ACOS	38.54	52.96	44.61
	SGTS	55.20	43.70	48.80
Generative	Paraphrase	N/A	N/A	<u>61.16</u>
	Seq2Path	N/A	N/A	58.41
	Special_Symbols+ UAUL	<u>61.22</u>	59.87	60.53
	DLO + UAUL	61.03	<u>60.55</u>	60.78
	ILO + UAUL	59.46	59.12	59.29
	BART-CRN	50.84	47.10	48.90
	The approach in this paper	63.69(↑2.47)	62.63(↑2.08)	63.15(↑1.99)

Table 2. Experimental results on the Restaurant dataset. Significant values are in bold and underlined.

	Methods	Precision (%)	Recall (%)	F1 (%)
Extractive methods	DP-ACOS	13.0	5.70	8.0
	Extract-Classify-ACOS	45.56	29.48	35.80
	SGTS	41.40	32.30	36.30
Generative	Paraphrase	N/A	N/A	43.51
	Seq2Path	N/A	N/A	42.97
	Special_Symbols+ UAUL	44.38	<u>43.65</u>	<u>44.01</u>
	DLO + UAUL	43.78	43.53	43.65
	ILO + UAUL	43.92	43.46	43.69
	BART-CRN	48.16	31.83	38.32
	The approach in this paper	<u>46.12</u>	44.84(↑1.19)	45.47(↑1.37)

Table 3. Experimental results on the laptop dataset. Significant values are in bold and underlined.

where, FN is the number of quadruples that exist in the label itself but are not predicted.

Sometimes Precision and Recall are contradictory. The usual method used is F1 value, which aims to perform weighted harmonic average of precision value and recall value, and the calculation formula is shown in Eq. (18)

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (18)$$

Comparative experiment

We compare the proposed method with the following two methods on the Restaurant and Laptop datasets:

- (1) Extractive methods: **DP-ACOS**: Dependency parsing is used to capture syntactic relations between words in a sentence to extract aspect category, opinion, and sentiment quadruples²⁶. **Extract-Classify-ACOS**: First jointly Extract aspect-opinion and then predict category-sentiment pairs²⁶. **SGTS**: The grid marking mechanism is used to jointly extract aspect category-opinion-emotion quadruples guided by sentence-level information³¹.
- (2) Generative methods: **Paraphrase**: Design language template and generate final target by filling elements according to template¹⁷. **Seq2Path**: Generates tuples as paths for the tree and then selects valid paths³². **Special_Symbols+ UAUL**: By learning the errors generated in the model's predictions and consciously avoiding repeating these errors in the generation process, the performance of the generative model (UAUL) is improved. Additionally, special symbols are designed to mark specific emotion, aspect, or opinion words to help the model better handle quadruples²⁹. **DLO**: Let the model learn to make correct predictions and let it learn to avoid incorrect predictions¹⁸. **ILO**: Let the model focus on correctly predicting situations that are particularly difficult or error-prone, directly optimizing the correct output for these complex instances¹⁸. **DLO + UAUL**: The combination of UAUL and DLO is adopted²⁹. **ILO + UAUL**: The combination of UAUL and ILO is adopted²⁹. **BART-CRN**: Contrastive learning is used to distinguish similar aspects or opinions, and a retrospective mechanism is used to refine the prediction³³.

All experimental results in this paper are obtained based on the pre-trained model T5-Base, and the results are shown in Tables 2 and 3.

Tables 3 and 4 shows the experimental results of the proposed model and the baseline model on two datasets. The results highlight the superiority of generative models over extractive models. This advantage is mainly attributed to two factors: (1) the generative model can directly generate quadruples in an end-to-end framework, thereby avoiding the multi-step error propagation problem common in the extraction model; (2) Generative

Method name	Core mechanism/improvement	Advantages	Limitations
Extraction Models (DP-ACOS/Extract-Classify-ACOS/SGTS)	Multi-stage extraction	Intuitive method, stable training, effective for explicit information extraction	Multi-step structure prone to error propagation, relies on explicit annotations, difficult to handle implicit content
Seq2Path	Path sequence generation	Suitable for explicit information, clear structure	High miss rate when path clues are absent, difficult to handle implicit quadruples
BART-CRN	Revisiting mechanism	Can correct some initial wrong predictions	Implicit quadruples lack annotation, revisiting mechanism struggles to perceive them
ILO + UAUL	Hard sample optimization + Unlikelihood learning	Reduces repeated errors, focuses on complex instances	Contextual dependencies not modeled, effectiveness limited to within-sentence context
DLO + UAUL	Positive/Negative likelihood joint optimization + Unlikelihood learning	Better distinguishes correct/incorrect predictions, handles complex sentences well	Still limited in capturing long-distance dependencies
Special_Symbols	Special symbols to mark aspect/opinion/sentiment + Unlikelihood learning	Explicit markers help the model recognize implicit clues	Symbols are insufficient to express deeper meanings in complex contexts
Paraphrase	Text paraphrasing to assist implicit reasoning	Enhances semantic diversity, improves adaptability to varied expressions	Easily introduces noise, may misassociate unmentioned content
Proposed Method	Multi-order prompt fusion + Graph Neural Network + Pointer-index generation mechanism +	Strong implicit reasoning, accurate long-distance dependency capture, complete structural information	Requires high reasoning ability, slightly higher training complexity

Table 4. Comparison of existing methods and our proposed approach.

models can generate relevant implicit information in the context, while extractive models often rely on the labeling of explicit features, so they perform poorly in the face of unclear opinion words and aspect words.

Seq2Path is one of the earlier generation models, which relies on path generation methods and can effectively capture explicit quadruples, and is more suitable for scenarios with clear explicit information. However, there is often no clear path or clue for implicit quadruples to be generated, resulting in a high miss rate. In order to improve the processing of implicit information, subsequent models begin to introduce more mechanisms, such as review mechanisms, impossibility learning mechanisms, special symbol enhancement mechanisms, and paraphrase strategies. The review mechanism of the BART-CRN model allows the model to correct the generated results by checking again after the first prediction. However, the implicit quadruples lack explicit labels, and the review mechanism cannot effectively capture the implicit information, so the experimental results are not ideal. ILO improves the model's ability to handle complex inputs by focusing on difficult instances during training. It combines the ILO model with the impossible learning mechanism to consciously avoid making the same mistakes repeatedly. However, ILO mainly optimizes for a single instance and fails to fully capture the dependencies in the global context. DLO optimizes both positive and negative likelihoods simultaneously, helping the model better distinguish between correct and incorrect predictions. Combined with the impossible learning mechanism, the model's performance on the implicit quadruple task is improved. However, its effectiveness is limited when handling complex context dependencies, particularly with long-distance dependencies. Special_Symbols make it easier for the model to understand opinion expressions by introducing special symbols into the input to mark aspect words, sentiment words, and opinion words. This mechanism can help the model deduce implicit aspects from the context through an inference process. This is particularly useful when the implicit aspects do not explicitly appear in the text. At the same time, the combination with the impossibilistic learning mechanism improves prediction performance in implicit contexts. However, when the context information for implicit aspects is insufficient or overly complex, symbolic labeling may not adequately capture the nuances of implicit quadruples. The paraphrase model generates new expressions by rephrasing the text, thereby enhancing the model's ability to handle diverse inputs. This capability allows it to capture quadruple information in various forms and improves its understanding of implicit quadruples. However, the paraphrase strategy can introduce noise in the implicit quadruple task. This noise may lead to misassociations with content that is not explicitly mentioned in the original sentence during the paraphrasing process.

In this paper, we propose a graph-enhanced implicit aspect-level sentiment analysis generation model based on multi-prompt fusion. This model introduces prompt learning to better focus on implicit aspects and sentiment in the generation process. It aims to avoid the noise introduced by paraphrase models while providing a more flexible implicit reasoning mechanism than explicit symbolic tagging. The defect of DLO and ILO is that they cannot adequately capture dependencies in the global context when dealing with complex context dependencies. The GNN introduced in this paper excels at capturing complex dependencies in long-distance texts. Experiments show that the proposed model outperforms other generative models on both datasets, with an F1 value increase of 1.99% on the Restaurant dataset and 1.37% on the Laptop dataset. The results highlight the superiority of the proposed model in the generation of emotional quadruples. The relevant summary is shown in Table 4.

Despite the significant performance improvement over the baseline model, the overall performance still needs to be improved. This can be attributed to two main factors. Firstly, the complexity of the task itself requires the extraction and exact matching of four different types of emotional elements, which increases the difficulty.

Error type	Error description	Count	Proportion (%)
Aspect Recognition Error	Some implicit aspect terms were not recognized	50	17.5
Sentiment Polarity Misclassification	The sentiment expression of implicit aspects was not correctly understood	34	12.0
Aspect-Opinion Misalignment	Implicit opinion terms were incorrectly aligned to unrelated aspects	57	20.0
Aspect Category Misclassification	Confusion between similar aspect categories (e.g., <i>food quality</i> vs. <i>food general</i>)	49	17.2
Incomplete Aspect Extraction	Complex sentences with multiple implicit aspects were not fully extracted	95	33.3
Total		285	100.0

Table 5. Error analysis on the restaurant dataset. Significant values are in bold.

Methods	P	R	F1
T5+Prompt+DL+index+GNN	63.69	62.63	63.15
T5+Prompt+DL+index	63.36	61.18	62.25
T5+Prompt+index+GNN	63.02	61.60	62.30
T5+Prompt+DL+GNN	62.92	61.14	62.02
T5+DL+GNN+index	61.28	60.73	61.01
T5+Prompt+index	62.32	61.07	61.69
T5+Prompt+GNN	62.90	60.70	61.78
T5+Prompt+DL	64.25	59.06	61.55
T5+Prompt	64.35	58.96	61.54
T5	61.27	59.06	60.14

Table 6. Results of ablation experiments on Restaurant dataset.

Secondly, the presence of many aspect items and opinion items in the language poses challenges for computers to understand, hindering the model's performance.

Error analysis

To further investigate the performance of the model, an in-depth analysis of typical error cases was conducted. The main types of errors are summarized as follows:

- Aspect recognition errors: For instance, in the sentence “While waiting for the meal, the waiter also poured us tea,” the model mistakenly identifies “waiter” as the aspect, while the correct implicit aspect should be “ervice details.”
- Sentiment polarity misclassification: For example, in the sentence “The restaurant is good overall, but a little noisy,” the model fails to correctly capture the negative sentiment associated with “noise” and instead predicts a neutral sentiment.
- Misalignment between aspect and opinion: In the sentence “It looks unremarkable, but tastes amazing,” the model incorrectly associates the opinion word “unremarkable” with the aspect “taste.”
- Incorrect sentiment polarity assignment: In “The weight is not the largest,” the sentiment should be considered neutral, but the model incorrectly classifies it as negative.
- Incomplete aspect extraction: For complex sentences involving negation, contrast, or long-distance dependencies, the model struggles to distinguish multiple aspects and sentiments. For instance, in “The decoration of this store looks like a product of the last century, but the food is amazing,” the model fails to correctly identify and evaluate both “decoration” and “food taste” separately.

These errors indicate that while the proposed model performs well overall on the implicit ABSA task, there is still room for improvement when handling polysemous expressions, unclear syntactic alignments, and metaphorical sentiment expressions. Detailed error statistics are shown in Table 5.

Ablation experiment

Ablation experiments were conducted to evaluate the impact of different combinations of methods on this task. These experiments were performed under the same experimental conditions to ensure consistency. Results of the ablation experiments as shown in Tables 6 and 7.

Where T5 is the use of T5-BASE model only, Prompt is the multi-prompt fusion. DL is the improved differentiated loss function, Index is the formulation of the text generation task for opinion and aspect items as an index generation problem, and GNN is the addition of graph neural network to the T5 model. The experimental results in Tables 5 and 6 show that, when using only the T5-BASE model, the F1 values for the two datasets are 60.14% and 43.32%, respectively. After introducing the multi-prompt fusion approach, the F1 values increase by 1.4% and 0.6%, respectively. These results demonstrate the importance of multi-prompt fusion, as it enables the model to better capture prompts and semantic information in the context, effectively processing implicit emotions. The F1 values after the introduction of differentiated loss and multi-prompt fusion improved by 0.01%

Methods	P	R	F1
T5+Prompt+DL+index+GNN	46.12	44.84	45.47
T5+Prompt+DL+index	45.20	44.15	44.67
T5+Prompt+index+GNN	45.41	44.67	45.04
T5+Prompt+DL+GNN	45.81	44.06	44.92
T5+DL+GNN+index	44.24	43.28	43.75
T5+Prompt+index	45.38	43.45	44.40
T5+Prompt+GNN	44.99	44.41	44.70
T5+Prompt+DL	44.83	43.28	44.04
T5+Prompt	44.78	43.09	43.92
T5	43.90	42.76	43.32

Table 7. Results of ablation experiments on Laptop dataset.

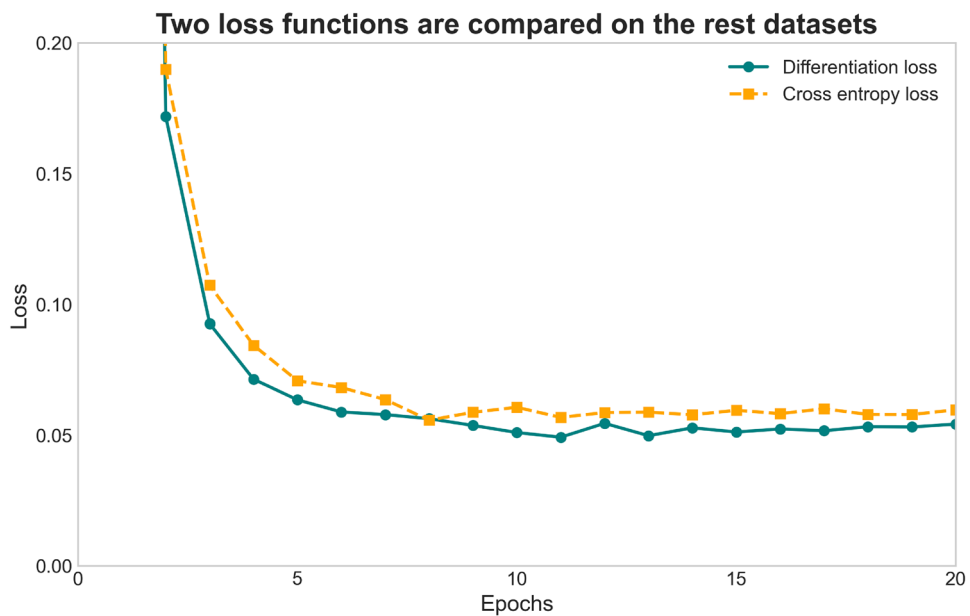


Fig. 8. Comparison of loss.

and 0.12% over the multi-prompt fusion alone in the two datasets, respectively. This indicates that differentiated loss makes the differences between positive, negative, and neutral texts more apparent. Additionally, the F1 values after introducing indexed generation tasks and multi-prompt fusion improved by 0.19% and 0.48% over the multi-prompt fusion alone, suggesting that indexing reduces the output space vocabulary to the number of task-related tags, thereby decreasing task complexity and enhancing model performance. Introducing a GNN model with multi-prompt fusion improves 0.24% and 0.78% over introducing multi-prompt fusion in the two datasets, respectively. The introduction of GNN effectively captures the complex dependencies between words in the node sentences. This enhancement improves the model's sensitivity and understanding of the context. Combining all the modules into one and comparing them with the T5 model results in improvements of 3.01% and 2.15% in the two datasets, respectively. This demonstrates that all the enhanced components of this paper are effective for implicit aspect-level sentiment analysis. This paper also compares the loss values between the differentiated loss and the cross-entropy loss. The comparison is illustrated in Fig. 8.

The results show that the differentiated loss decreases a little faster than the cross-entropy loss in the early stages of training, this means that the differentiated loss adjusts the parameters of the model faster. Throughout the entire training process, the value of the differentiated loss is generally lower than that of the cross-entropy loss function. This indicates that the differentiated loss function enables the model to achieve smaller errors during training, resulting in better performance.

Multi-prompt fusion effect experiment

To further validate the importance of the template order enhancement approach to the model, this paper compares it with no prompt learning introduced, with a single prompt order, and with a multiple prompt order approach in 2 datasets. The experimental results are shown in Tables 8 and 9. And validate the effect of different number of prompts in Restaurant dataset for the model, as shown in Fig. 9.

Method	P	R	F1
T5	61.27	59.06	60.14
Fixed-sequence prompt	61.38	62.07	61.72
Multi-prompt fusion	63.69	62.63	63.15

Table 8. Comparative experiments of different prompting methods on Restaurant dataset.

Method	P	R	F1
T5	43.90	42.76	43.32
Fixed-sequence prompt	44.91	43.19	44.03
Multi-prompt fusion	46.12	44.84	45.47

Table 9. Comparative experiments of different prompting methods on Laptop dataset.

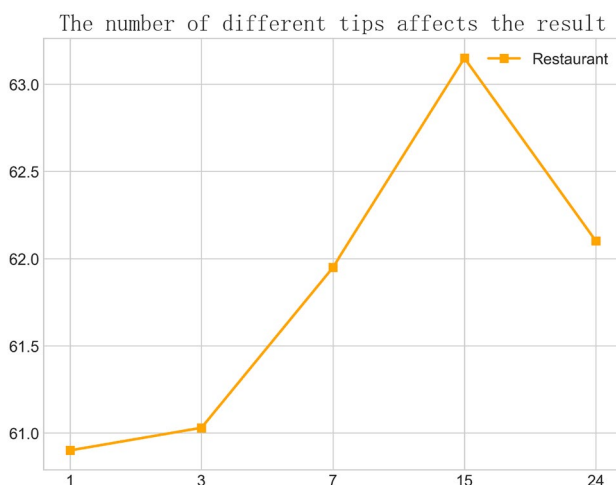


Fig. 9. Experimental results for different number of prompts.

We comparatively analyzed the effectiveness of the T5 model by introducing different prompt learning strategies, namely learning without prompts, fixed-order prompt learning, and multi-prompt fusion learning. The results show that multi-prompt fusion learning has the best effect, while learning without prompting performed the worst. At the same time, we also verified the impact of the number of prompts on the performance of the model, and found that the effect gradually improved when the number of prompts increased from 1 to 15. However, when using 24 prompt orders, the model's performance actually decreased. This indicates that an increase in the number of prompts may lead to complex trade-offs in the model between multiple sequences, making it difficult to efficiently integrate all prompts and, in turn, reducing overall performance.

Results of lack of resources

To further explore the behavior of the method in this paper in a low-resource environment, we utilize five different sizes of training sets: 1%, 2%, 5%, 10%, and 20%. These training sets are applied to two datasets to train the model proposed in this paper. The experimental results are shown in Fig. 10.

It is found that the proposed method achieves good results even when working with only a small number of samples. The comparison with models DLO and Paraphrase on the Restaurant dataset is shown in Fig. 11.

The model in this paper does not perform as well as the Paraphrase and DLO models at 1%, 2%, and 5% resources, but outperforms both models at 5% and above. This is because, first, the introduction of GNN increases the complexity of the model and introduces more parameters. This results in performance degradation when the model does not have access to enough data to efficiently learn its parameters at low resource levels of 5% and below. Second, GNN relies on graph structure information to capture relationships between nodes. With low resources of 5% and below, the model may not be able to obtain enough information to learn the joint features of the graph structure and the language model. In contrast, the Paraphrase model can be efficiently trained with very little data through direct sentence transformation and template-based DLO strategies. When the amount of data exceeds 5%, T5 and GNN can effectively utilize more context and graph structure information. This allows the model to progressively adapt and improve its ability to capture implicit features and graph structure information, thus surpassing the Paraphrase and DLO models.

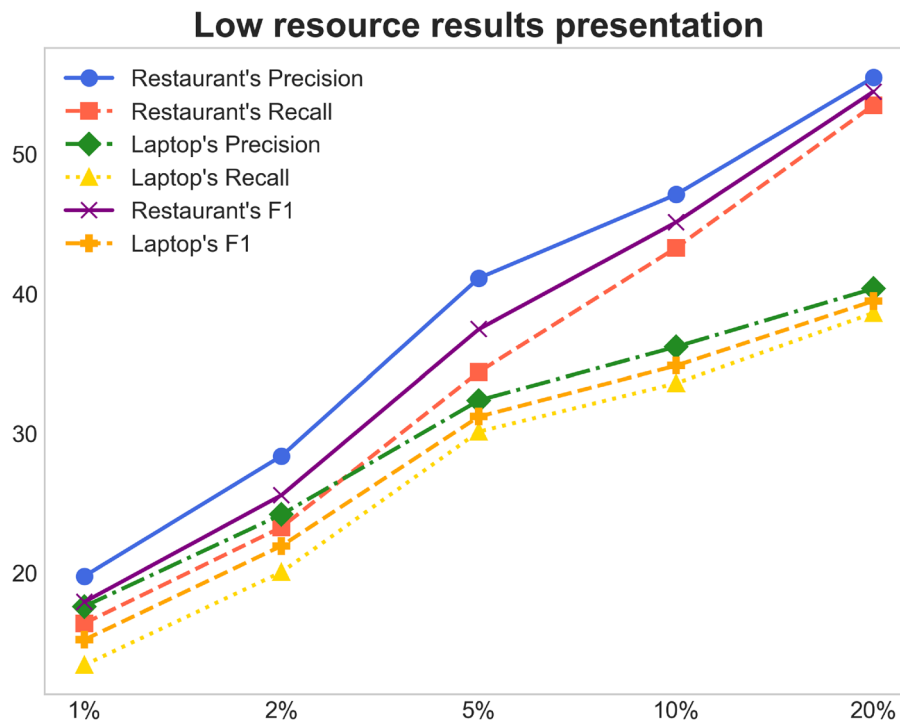


Fig. 10. Results of experiments with low resources.

Comparison of experimental results with low resource models on Rest datasets

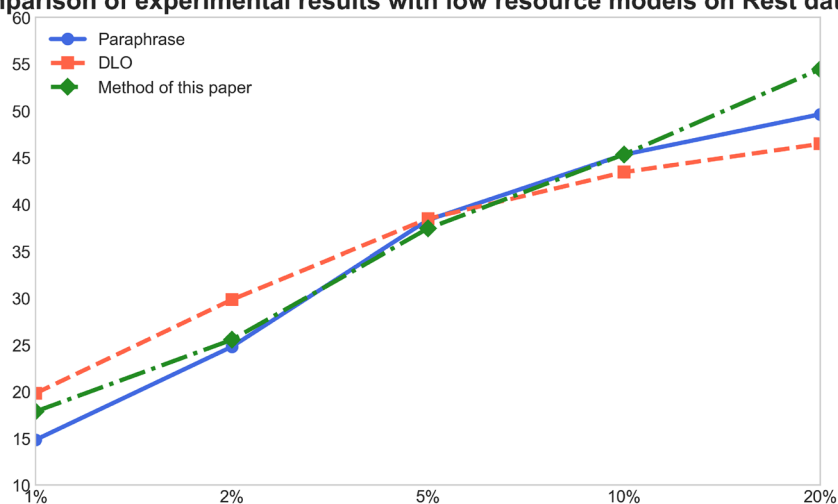


Fig. 11. Comparison of low resource models.

Conclusion

In order to address the challenges in implicit aspect-level sentiment analysis, this paper proposes a graph-enhanced generative model based on multi-cue fusion. The model incorporates graph neural networks to effectively capture structural cues and contextual semantic information, enabling a better understanding of complex implicit sentiment expressions. Simultaneously, a multi-sequential cue fusion mechanism is introduced to fully leverage the complementary strengths of multiple cues, thereby mitigating the limitations of single cues in terms of incomplete information and generation instability. Additionally, the output format of aspect and opinion terms in the quadruple task is converted into the index form of the original text, which reduces the output space and enhances generation efficiency. A differentiated loss function is also designed to improve the model's capability in identifying and handling various types of prediction errors. Experimental results demonstrate that the proposed method improves the F1 score by 1.99% and 1.37% on the Restaurant and Laptop datasets, respectively, compared to baseline models, thereby validating the effectiveness of the approach. Ablation studies

further reveal that each core component significantly contributes to the model's performance. Moreover, owing to its modular architecture and robust context modeling capability, the proposed model consistently achieves performance gains across different text structures and datasets, showing strong task generalization and cross-scenario adaptability. It also exhibits potential for extension to other sentiment analysis tasks. Nevertheless, the current model still exhibits recognition bias when processing sentences containing explicit aspect or sentiment words, which hinders further performance improvement. Future work may focus on enhancing contextual understanding, incorporating external knowledge to aid inference, and exploring the model's adaptability and application potential across a wider range of scenarios, such as cross-domain, multilingual, and multimodal tasks. These directions are expected to promote the in-depth development and cross-domain integration of implicit aspect-level sentiment analysis.

Data availability

The data utilized in this study were extracted from Cai, Hongjie and Xia, Rui and Yu, Jianfei's research on Aspect-Category-Opinion-Sentiment Quadruple Extraction with Implicit Aspects and Opinions. RestaurantACOS and LaptopACOS datasets used in Cai, Hongjie et al's study is available <https://doi.org/10.18653/v1/2021.acl-long.29>.

Received: 27 October 2024; Accepted: 14 May 2025

Published online: 20 May 2025

References

- Liu, H., Chatterjee, I., Zhou, M., Lu, X. S. & Abusorrah, A. Aspect-based sentiment analysis: A survey of deep learning methods. *IEEE Trans. Comput. Social Syst.* **7**, 1358–1375 (2020).
- Zhang, W., Li, X., Deng, Y., Bing, L. & Lam, W. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Trans. Knowl. Data Eng.* **35**, 11019–11038 (2022).
- Xu, P., Madotto, A., Wu, C.-S., Park, J. H. & Fung, P. Emo2vec: Learning generalized emotion representation by multi-task training. *arXiv preprint*, [arXiv:1809.04505](https://arxiv.org/abs/1809.04505) (2018).
- Chen, Y., Perozzi, B., Al-Rfou, R. & Skiena, S. The expressive power of word embeddings. *arXiv preprint*, [arXiv:1301.3226](https://arxiv.org/abs/1301.3226) (2013).
- Lee, L.-H., Li, J.-H. & Yu, L.-C. Chinese emobank: Building valence-arousal resources for dimensional sentiment analysis. *Trans. Asian Low-Resource Language Inform. Process.* **21**, 1–18 (2022).
- Shi, J., Li, W., Bai, Q., Yang, Y. & Jiang, J. Syntax-enhanced aspect-based sentiment analysis with multi-layer attention. *Neurocomputing* **557**, 126730 (2023).
- Yuan, L., Wang, J., Yu, L.-C. & Zhang, X. Syntactic graph attention network for aspect-level sentiment analysis. *IEEE Trans. Artif. Intell.* **5**, 140–153 (2022).
- Wang, J. et al. Aspect sentiment classification with both word-level and clause-level attention networks. *IJCAI* **2018**, 4439–4445 (2018).
- Chai, H. et al. Aspect-to-scope oriented multi-view contrastive learning for aspect-based sentiment analysis. *Findings Assoc. Comput. Linguistics EMNLP* **2023**, 10902–10913 (2023).
- Wang, J., Yu, L.-C. & Zhang, X. Softmcl: Soft momentum contrastive learning for fine-grained sentiment-aware pre-training. *arXiv preprint*, [arXiv:2405.01827](https://arxiv.org/abs/2405.01827) (2024).
- Tubishat, M., Idris, N. & Abushariah, M. A. Implicit aspect extraction in sentiment analysis: Review, taxonomy, opportunities, and open challenges. *Inform. Process. Manag.* **54**, 545–563 (2018).
- Meiqi, P., Zhiyuan, M., Gaofei, L. & Jiwei, Q. Overview of end-to-end aspect sentiment analysis. *Small Microcomput. Syst.* **45**, 732–746. <https://doi.org/10.20009/j.cnki.21-1106/TP.2023-0342> (2024).
- Yang, C., Han, Y., Chen, Q. & MA, W. An aspect-level implicit sentiment analysis model based on bert and attention mechanism. *J. Nanjing Univ. Inform. Sci. Technol. (Nat. Sci. Edn.)* **15**, 551–560 (2023).
- Lewis, M. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint*, [arXiv:1910.13461](https://arxiv.org/abs/1910.13461) (2019).
- Zheng, G., Wang, J., Yu, L.-C. & Zhang, X. Instruction tuning with retrieval-based examples ranking for aspect-based sentiment analysis. *arXiv preprint*, [arXiv:2405.18035](https://arxiv.org/abs/2405.18035) (2024).
- Raffel, C. et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Machine Learn. Res.* **21**, 1–67 (2020).
- Zhang, W. et al. Aspect sentiment quad prediction as paraphrase generation. *arXiv preprint*, [arXiv:2110.00796](https://arxiv.org/abs/2110.00796) (2021).
- Hu, M., Wu, Y., Gao, H., Bai, Y. & Zhao, S. Improving aspect sentiment quad prediction via template-order data augmentation. *arXiv preprint*, [arXiv:2210.10291](https://arxiv.org/abs/2210.10291) (2022).
- Wang, X. et al. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint*, [arXiv:2203.11171](https://arxiv.org/abs/2203.11171) (2022).
- Zhang, P. et al. Transgnn: Harnessing the collaborative power of transformers and graph neural networks for recommender systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1285–1295 (2024).
- Vanaja, S. & Belwal, M. Aspect-level sentiment analysis on e-commerce data. In *2018 International conference on inventive research in computing applications (ICIRCA)*, 1275–1279 (IEEE, 2018).
- Li, X., Bing, L., Li, P. & Lam, W. A unified model for opinion target extraction and target sentiment prediction. *Proc. AAAI Conf. Artif. Intell.* **33**, 6714–6721 (2019).
- Li, Y., Wang, F. & Zhong, S.-H. A more fine-grained aspect-sentiment-opinion triplet extraction task. *Mathematics* **11**, 3165 (2023).
- Mao, Y., Shen, Y., Yu, C. & Cai, L. A joint training dual-mrc framework for aspect based sentiment analysis. *Proc. AAAI Conf. Artif. Intell.* **35**, 13543–13551 (2021).
- Yan, H., Dai, J., Qiu, X., Zhang, Z. et al. A unified generative framework for aspect-based sentiment analysis. *arXiv preprint*, [arXiv:2106.04300](https://arxiv.org/abs/2106.04300) (2021).
- Cai, H., Xia, R. & Yu, J. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. 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)*, 340–350 (2021).
- Liu, P. et al. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* **55**, 1–35 (2023).
- Fei, H. et al. Reasoning implicit sentiment with chain-of-thought prompting. *arXiv preprint*, [arXiv:2305.11255](https://arxiv.org/abs/2305.11255) (2023).
- Hu, M. et al. Uncertainty-aware unlikelihood learning improves generative aspect sentiment quad prediction. *arXiv preprint*, [arXiv:2306.00418](https://arxiv.org/abs/2306.00418) (2023).
- Wolf, T. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint*, [arXiv:1910.03771](https://arxiv.org/abs/1910.03771) (2019).

31. Zhu, L. et al. Aspect sentiment quadruple extraction based on the sentence-guided grid tagging scheme. *World Wide Web* **26**, 3303–3320 (2023).
32. Mao, Y., Shen, Y., Yang, J., Zhu, X. & Cai, L. Seq2path: Generating sentiment tuples as paths of a tree. *Findings Assoc. Comput. Linguistics ACL* **2022**, 2215–2225 (2022).
33. Xiong, H. et al. Bart-based contrastive and retrospective network for aspect-category-opinion-sentiment quadruple extraction. *Int. J. Machine Learn. Cybern.* **14**, 3243–3255 (2023).

Acknowledgements

First of all, I would like to give my heartfelt thanks to all the people who have ever helped me in this paper. Hereby, I would like to express their gratitude to Cai, Hongjie et al for generously sharing the RestaurantACOS and LaptopACOS Dataset used in this study. The availability of this dataset was instrumental in the completion of our research. I am also extremely grateful to all my friends and classmates who have kindly provided me assistance and companionship in the course of preparing this paper. In addition, many thanks go to my family for their unflinching love and unwavering support.

Author contributions

Xinlong Wang and Xu Li conducted experiments, analyzed data, and drafted the manuscript. Chunlong Yao and Yang Li contributed to the study design and provided intellectual input throughout the research process.

Funding

This work was supported by the Scientific Research Fund for Higher Education Institutions of Liaoning Province, China, under Grant LJ212410152070.

Declarations

Competing interests

The authors declare no competing interests.

Ethical approval

Written informed consent for publication of this paper was obtained from Dalian Polytechnic University and all authors.

Additional information

Correspondence and requests for materials should be addressed to X.L.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025