Syntax-Enhanced Network with Gated Relational Attention for Aspect Sentiment Triplet Extraction

Bowen Wang
School of Computer Science and Technology
Soochow University
Suzhou, China
20235227144@stu.suda.edu.cn

Xiaoxu Zhu
School of Computer Science and Technology
Soochow University
Suzhou, China
xiaoxzhu@suda.edu.cn

Peifeng Li
School of Computer Science and Technology
Soochow University
Suzhou, China
pfli@suda.edu.cn

***Abstract*—Aspect Sentiment Triplet Extraction (ASTE) has emerged as a critical task in sentiment analysis, specifically aimed at identifying aspect terms, opinion terms, and their corresponding sentiment polarities within textual data. While recent span-based tagging models have achieved promising results, they typically rely on coarse-grained contextual representations, either without explicitly modeling syntactic structure or without mechanisms to selectively control its influence, which limits their ability to effectively distinguish task-relevant information from noise during triplet extraction. To address these limitations, we propose a Syntax-Enhanced Network with Gated Relational Attention (SENGRA), which augments span-based tagging through the selective integration of syntactic structures. SENGRA employs a Gated Relational Graph Attention Network (G-RGAT) to encode typed dependency relations, in which an input-dependent gating mechanism modulates the graph-aggregated token representations. This design effectively reduces noise from irrelevant syntactic information while improving the model's nonlinear expressive capacity. Furthermore, we design a syntax-guided inference strategy to accurately align aspect and opinion terms within sentiment spans. Extensive experiments on four widely adopted ASTE benchmarks show that SENGRA consistently surpasses state-of-the-art methods, confirming its efficacy and robustness in complex sentiment extraction scenarios.**

***Index Terms*—Aspect Sentiment Triplet Extraction, Graph neural networks, Dependency tree, Attention mechanism**

I. Introduction

ASTE is a pivotal task in aspect-based sentiment analysis (ABSA) [1], aiming to extract structured sentiment information from text in the form of triplets consisting of aspect terms, opinion terms, and sentiment polarity. Unlike traditional sentiment classification, which assigns a single sentiment label to an entire sentence or document, ASTE explicitly associates sentiment expressions with the specific entities or attributes they describe. For example, in the sentence "The ambience was nice, but the service wasn't so great", two sentiment triplets can be identified: ("ambience", "nice", Positive) and ("service", "wasn't so great", Negative), where "ambience" and "service" are aspect terms, "nice" and "wasn't so great" are the corresponding opinion terms, and the sentiment polarities are Positive and Negative, respectively.

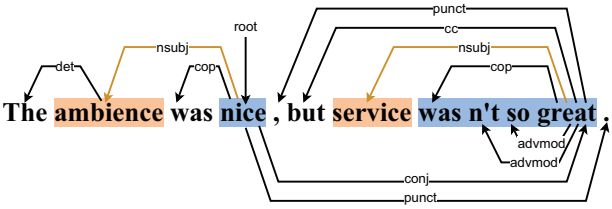
Early ASTE methods followed a pipeline design [2], [3], extracting each component step by step. While intuitive, these approaches suffer from error propagation across stages. To address this limitation, end-to-end models [4]–[6], particularly tagging-based formulations [4], [6], [7], have been proposed, framing ASTE as a structured prediction task to jointly extract triplet components.

However, many tagging-based models struggle with long-range sentiment dependencies. Without explicit modeling of word relationships, they often misidentify aspect-opinion pairs when related terms are distant. Moreover, irrelevant tokens can introduce noise, degrading extraction accuracy.

To mitigate these issues, graph-based approaches [8]–[10], such as Graph Convolutional Networks (GCNs), have been adopted to encode syntactic dependencies. However, syntactic dependencies may contribute unequally to aspect sentiment triplet extraction, and indiscriminate aggregation of graph information may introduce noise and unnecessary complexity, degrading model performance.

For instance, as illustrated in Fig. 1, certain dependency relations provide useful structural cues for associating aspect and opinion expressions, while others offer limited or even misleading information. Treating all syntactic relations uniformly may therefore obscure informative signals and amplify irrelevant ones.

To address this, we propose SENGRA, which selectively incorporates useful syntactic dependencies while reducing the influence of irrelevant ones. Our model introduces a G-RGAT to encode typed syntactic dependencies through relation-aware



Aspect	Opinion	Sentiment
ambience	nice	Positive
service	wasn't so great	Negative

Fig. 1. An example sentence with its dependency tree for the ASTE task. Aspect terms are highlighted in yellow, while opinion terms are highlighted in blue.

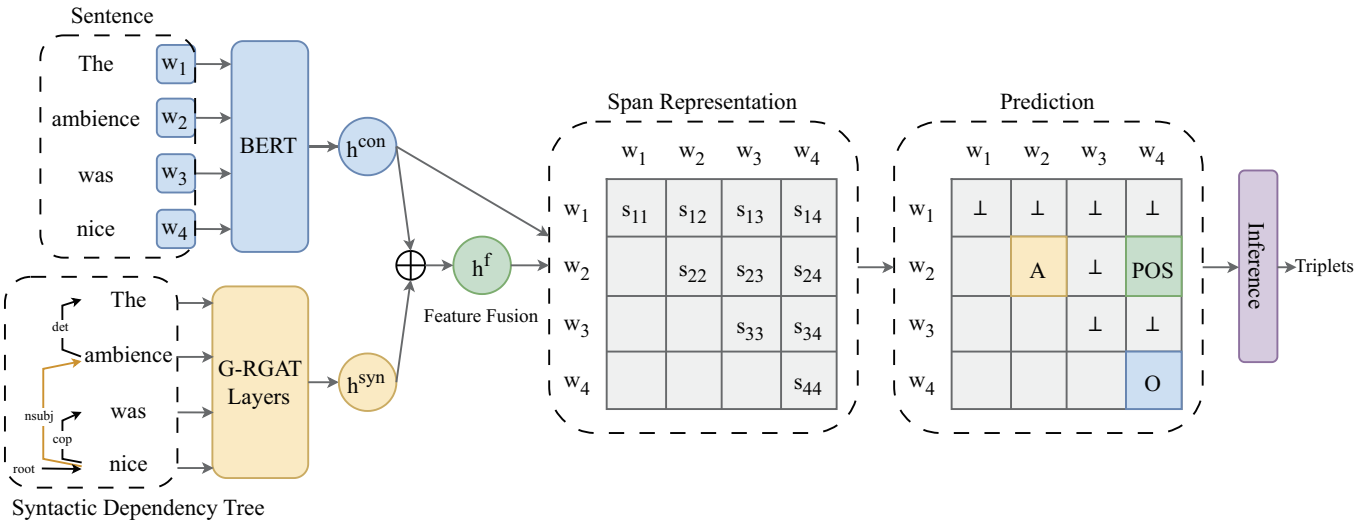


Fig. 2. The overall architecture of SENGRA.

attention. Building on recent advances in gated attention [11], an input-dependent gating mechanism is further incorporated to regulate the information aggregated from syntactic graphs. Relation-aware gated attention enables the model to selectively capture informative syntactic structures while suppressing irrelevant or unreliable dependency signals. In addition, a separate fusion gate integrates the resulting syntactic features with BERT-based contextual embeddings to form enhanced span representations.

Furthermore, existing tagging-based models often lack effective strategies for selecting aspect and opinion spans from overlapping candidates. Prior methods either use exhaustive pairwise scoring [8], which is computationally expensive, or employ greedy selection [6], which may reduce precision. Our method leverages syntactic cues to balance efficiency and accuracy—favoring longer spans when supported by dependencies and otherwise preferring shorter ones—thus improving both speed and extraction quality.

Our main contributions are as follows:

1) We propose SENGRA, a syntax-enhanced network for aspect sentiment triplet extraction that integrates syntactic structure into span-level modeling, enabling more reliable representations for capturing long-range sentiment dependencies.

2) We design a G-RGAT that encodes typed dependency relations via relation-aware gated attention, where an input-dependent gate modulates graph-aggregated syntactic representations, reducing noise from irrelevant or unreliable dependency information.

3) We devise a syntax-guided inference strategy that enhances the efficiency and accuracy of aspect-opinion term selection within sentiment spans, achieving superior performance compared to traditional methods.

To facilitate reproducibility and further research, our code is publicly available at https://github.com/65ABSA/SENGRA.

II. Related Work

ABSA [1], [12], [13] is a fine-grained task that analyzes opinions toward specific aspects in text. Traditional ABSA is typically decomposed into Aspect Term Extraction (ATE) [14], [15], Opinion Term Extraction (OTE) [16], [17], and Aspect Sentiment Classification (ASC) [18], [19]. Recent formulations, such as Aspect-Opinion Pair Extraction (AOPE) [20], [21], jointly extract aspects and opinions but still fail to produce complete sentiment triplets. To address this limitation, the ASTE task was introduced [3], aiming to jointly extract aspect, opinion, and sentiment polarity.

Among ASTE methods, tagging-based approaches have received considerable attention. The earliest model [3] employed a unified tagging scheme but suffered from error propagation and limited dependency modeling. To mitigate these issues, a position-aware tagging scheme [7] incorporates positional encoding and attention mechanisms. The grid tagging scheme [4] further models relationships in a two-dimensional grid, while STAGE [6] formulates ASTE as a multi-class span classification task.

However, tagging-based methods often struggle to capture complex contextual dependencies. To address this limitation, graph-based approaches have been explored. EMC-GCN [8] models multiple types of word relations via a multi-channel GCN with biaffine attention. DRN [22] adopts dual GCN encoders for entity extraction and matching, while DGCNAP [9] integrates affective knowledge and positional information into dual graph networks. These methods leverage syntactic structures to better capture high-order dependencies, alleviating the limitations of tagging-based approaches.

III. Methodology

In this section, we detail the architecture of the proposed SENGRA. As depicted in Fig. 2, the model initiates by

encoding the input sentence using Bidirectional Encoder Representations from Transformers (BERT) [23] to derive contextual representations $\mathbf{h}^{\text{ctx}}$. Concurrently, syntactic dependencies are processed through a G-RGAT to generate structure-aware embeddings $\mathbf{h}^{\text{syn}}$. These features are integrated using a gating mechanism to produce fused representations $\mathbf{h}^{\text{f}}$, which are subsequently employed to construct span representations. Ultimately, a tagging module classifies span roles while a syntax-aware inference component extracts sentiment triplets.

A. Problem Formulation

Given an input sentence $X = \{w_1, w_2, \ldots, w_n\}$, which consists of n words, the objective of the ASTE task is to identify all valid triplets $T = \{(a, o, s)_m\}_{m=1}^{|T|}$. In this context, a represents a specific text span corresponding to an aspect, o denotes a span that expresses an opinion, and s indicates the sentiment polarity associated with a. The sentiment polarity s is drawn from a predefined set $S = \{\text{POS}, \text{NEU}, \text{NEG}\}$, which categorizes sentiment as positive, neutral, or negative.

B. Contextual Encoding

We employ BERT to capture contextual dependencies within sentences. BERT utilizes deep Transformer layers to create context-aware representations for each token in the input sequence. For a given sentence $X = \{w_1, w_2, \ldots, w_n\}$, BERT tokenizes each word into subword units using the WordPiece tokenizer. The tokenized sequence is formatted as: $X' = \{[\text{CLS}], t_1, t_2, \ldots, t_m, [\text{SEP}]\}$, where t_i represents a subword token, [CLS] is a classification token, and [SEP] indicates sentence boundaries.

BERT processes this sequence, producing contextualized representations $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_m\}$, where $\mathbf{h}_i \in \mathbf{R}^d$ is the hidden state of token t_i from the final BERT layer, and d is the hidden dimension. Since some words split into multiple subword tokens, we aggregate their representations for word-level embeddings using mean pooling for each original word w_j consisting of subword tokens $\mathcal{T}_j$.

$$\mathbf{h}_j^{\text{ctx}} = \frac{1}{|\mathcal{T}_j|} \sum_{t_i \in \mathcal{T}_j} \mathbf{h}_i, \tag{1}$$

where $\mathbf{h}_j^{\text{ctx}} \in \mathbf{R}^d$ serves as the final representation for word w_j.

C. Syntax-Enhanced Representation

To integrate syntactic information into our model, we construct a syntactic dependency graph using dependency parsing and employ a RGAT to enhance span representations.

1) Graph Construction: The syntactic dependency graph is defined as $G_{\text{syn}} = (V, A, R)$, where $V = \{w_1, ..., w_n\}$ represents the set of words in a sentence. The edge set A is represented as an adjacency matrix $A \in \{0, 1\}^{n \times n}$ where $A_{ij} = 1$ if there is a dependency arc between words. The relation tensor R stores dependency labels, with R_{ij} containing a relation label only when $A_{ij} = 1$. Self-loops are explicitly encoded with $A_{ii} = 1$ for all $i \in \{1, ..., n\}$, where corresponding relations R_{ii} are assigned a special `self` label to preserve lexical semantics. Each non-zero entry $A_{ij} = 1$ is associated with a syntactic relation type (e.g., *nsubj*, *obj*) via the relation matrix.

2) Syntax Learning with G-RGAT: To capture long-range syntactic interactions while suppressing noisy or redundant dependencies, we employ a G-RGAT to encode the syntactic dependency graph. Following previous work [24], for each word w_i, we compute node-aware attention $e_{i,j}^N = f(\mathbf{h}_i^{\text{syn}(l-1)}, \mathbf{h}_j^{\text{syn}(l-1)})$ and relation-aware attention $e_{i,j}^R = f(\mathbf{h}_i^{\text{syn}(l-1)}, \mathbf{r}_{i,j})$ for $j \in \mathcal{N}(i)$, where $f(\cdot)$ is a scoring function. The combined attention weights are obtained through softmax normalization:

$$\alpha_{i,j} = \frac{\exp(e_{i,j}^N + e_{i,j}^R)}{\sum_{j' \in \mathcal{N}(i)} \exp(e_{i,j'}^N + e_{i,j'}^R)} \tag{2}$$

Based on the attention weights, syntactic neighbor information is aggregated to form an intermediate structure-aware context. For each attention head, the aggregation is performed as follows:

$$\mathbf{c}_i^{(l,z)} = \sum_{j \in \mathcal{N}(i)} \alpha_{i,j}^{(l,z)} (\mathbf{W}_V^{(z)} \mathbf{h}_j^{\text{syn}(l-1)} + \mathbf{W}_R^{(z)} \mathbf{r}_{i,j}) \tag{3}$$

While this aggregation captures rich syntactic interactions, it may still introduce noise due to parsing errors or structural ambiguity. To address this issue, we introduce an input-dependent gating mechanism [11] that adaptively regulates the contribution of the aggregated syntactic context.

Specifically, for each attention head z, a gating coefficient $\mathbf{g}_i^{(z)}$ is computed based solely on the current semantic state of the token:

$$\mathbf{g}_i^{(z)} = \sigma(\mathbf{W}_g^{(z)} \mathbf{h}_i^{\text{syn}(l-1)}) \tag{4}$$

The gated context is then projected to produce the output representation of the G-RGAT layer:

$$\mathbf{h}_i^{\text{syn}(l)} = \mathbf{W}_o \left(\|_{z=1}^{Z} \left(\mathbf{g}_i^{(z)} \odot \mathbf{c}_i^{(l,z)} \right) \right) \tag{5}$$

We employ Z independent attention heads with head-specific parameters $\mathbf{W}_V^{(z)}$, $\mathbf{W}_R^{(z)}$, and $\mathbf{W}_g^{(z)}$. The concatenated representations are further transformed by a linear projection $\mathbf{W}_o$. Here, $\|$ denotes the concatenation operation over all attention heads, σ denotes the activation function, and $\odot$ represents element-wise multiplication.

3) Feature Fusion: Following common practice, we adopt a lightweight fusion gate [25] to fuse syntactic and contextual representations dynamically:

$$\mathbf{h}_i^{\text{f}} = g_i^{\text{fuse}} \odot \mathbf{h}_i^{\text{syn}} + (1 - g_i^{\text{fuse}}) \odot \mathbf{h}_i^{\text{ctx}} \tag{6}$$

with gate $g_i^{\text{fuse}} = \sigma(\mathbf{W}_g^{\text{fuse}} [\mathbf{h}_i^{\text{syn}}; \mathbf{h}_i^{\text{ctx}}] + \mathbf{b}_g)$, where $\mathbf{W}_g^{\text{fuse}}$ and $\mathbf{b}_g$ are learnable parameters.

D. Span Representation

To effectively utilize both contextual and syntactic information, we construct the representation of each span by combining boundary-aware and content-aware features. The span representation is defined as:

$$\mathbf{s}_{i,j} = [\mathbf{h}_i^f; \mathbf{h}_j^f; \sum_{k=i}^{j} \mathbf{h}_k^{\text{ctx}}] \tag{7}$$

Here, $\mathbf{h}_i^f$ and $\mathbf{h}_j^f$ are syntactically-enhanced representations of the boundary words of the span, and $\sum_{k=i}^{j} \mathbf{h}_k^{\text{ctx}}$ aggregates the purely contextualized features of the words within the span.

We apply syntax-enhanced representations only to span boundaries, as dependency relations and syntactic roles typically emphasize phrase delimiters over internal content. Internal words mainly support semantic cohesion, which the contextual encoder already captures. Keeping their representation purely contextual avoids redundancy and lets the model focus on structural transitions where they matter most.

E. Tagging Scheme

To systematically capture the role of each text span in ASTE, we adopt a multi-dimensional span tagging scheme inspired by previous work [6]. This scheme allows each span to take multiple roles, reflecting the inherent overlap in aspect-based sentiment analysis. Each span can function as an Aspect (A), an Opinion (O), or carry Sentiment polarity (S), where sentiment is categorized into positive (POS), neutral (NEU), or negative (NEG).

To efficiently represent span roles, we construct an upper-triangular tagging table T, where each entry $T[i][j]$ corresponds to the tag assigned to the span $\mathbf{s}_{i,j}$, which covers words w_i to w_j. Since each span is uniquely determined by its start and end indices ($i \leq j$), the upper-triangular structure avoids redundant representations. Each tag is represented as a three-component label:

$$T[i][j] = (\phi_A, \phi_O, \phi_S) \tag{8}$$

The roles are defined as follows: $\phi_A \in \{A, -\}$ indicates whether the span is an aspect term (A) or not ($-$). $\phi_O \in \{O, -\}$ denotes whether the span is an opinion term (O) or not ($-$). Meanwhile, $\phi_S \in \{-, \text{POS}, \text{NEU}, \text{NEG}\}$ represents the sentiment polarity of the span, which may be positive (POS), neutral (NEU), negative (NEG), or none ($-$). Fig. 3 illustrates the span tagging scheme with an example sentence.

F. Syntax-Guided Inference

To enhance both the accuracy and efficiency of triplet extraction, we propose a syntax-guided inference strategy that leverages dependency structures to constrain span selection. Unlike greedy inference strategies [6] that tend to select the longest candidate span by default, our method explicitly incorporates syntactic evidence to guide aspect–opinion matching.

Specifically, longer spans are retained only when supported by explicit dependency relations; otherwise, the model favors shorter, syntactically coherent spans to prevent overly long spans that conflate aspect and opinion terms. For instance, in Fig. 1, given the sentence "The ambience was nice, but the service wasn't so great.", the candidate opinion related to "service" is not treated as a single span ("service wasn't so great"). Instead, multiple candidate spans are considered, such as "great" and "wasn't so great". By inspecting the dependency structure, we observe a copular (*cop*) relation between "was" and "great", which supports selecting "wasn't so great" as a valid opinion span.

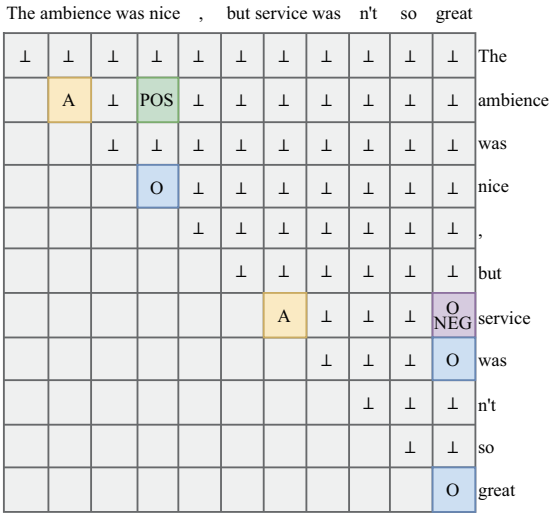


Fig. 3. An illustration of span tagging for triplet extraction, where each cell represents a span annotated with one or multiple labels.

Beyond improving extraction accuracy, this syntax-guided strategy substantially reduces computational complexity. Traditional span-based inference typically requires exhaustive pairwise comparisons between all aspect and opinion spans, yielding $\mathcal{O}(n^4)$ time complexity. In contrast, by restricting candidates to dependency-connected spans, our approach prunes numerous implausible combinations in advance, reducing inference to dependency-aware span matching with $\mathcal{O}(n^2)$ complexity. This theoretical reduction is expected to translate into practical efficiency gains, while detailed runtime and memory measurements are left for future work.

G. Training

To effectively train our model, we adopt a span-based classification approach. Each span representation is passed through a fully connected layer followed by a softmax activation function to predict the role of the span. The probability distribution over role tags is computed as:

$$p(s_{i,j}) = \text{softmax}(\mathbf{W}_r s_{i,j} + \mathbf{b}_r) \tag{9}$$

where $\mathbf{W}_r$ and $\mathbf{b}_r$ are learnable parameters of the classifier.

To train the model, we define the loss function as the cross-entropy loss between the predicted probability distribution and

the ground-truth labels:

$$L(\theta) = -\sum_{i=1}^{n}\sum_{j=i}^{n} \text{loss}(p(s_{i,j}), y_{i,j}) \tag{10}$$

where $y_{i,j}$ denotes the gold label for the span $SP_{i,j}$, and $\text{loss}(\cdot,\cdot)$ refers to the standard cross-entropy loss function. The model parameters, denoted by θ, are updated through backpropagation.

IV. Experiment

A. Datasets

We assessed our model using four widely adopted benchmark datasets: Lap14, Res14, Res15 and Res16. These datasets are derived from the SemEval ABSA challenges [1], [26], [27], and are collectively referred to as the V2 version [7], which refines and corrects earlier annotations. Each dataset includes manually annotated aspect-opinion-sentiment triplets and facilitates evaluation in both laptop and restaurant domains.

B. Experiment Settings

We utilized BERT-base-uncased with a hidden size of 768 coupled with a 200-dimensional projection layer. For syntax modeling, a two-layer RGAT with four attention heads per layer and 40-dimensional dependency relation embeddings was employed. The model was trained using the AdamW optimizer, with a learning rate of 2e-5 for BERT and 1e-3 for other components. Training parameters included a dropout rate of 0.5, a batch size of 16, and early stopping applied across 100 epochs.

C. Baselines

To evaluate our model, we compare it with a range of strong baselines, covering both pipeline-based and end-to-end ASTE approaches.

1) Pipeline-based methods:

- **CMLA+** [2]: A two-stage model using attention-enhanced memory networks for extracting aspects/opinions and a binary classifier to filter invalid triplets.
- **RINANTE+** [28]: A BiLSTM-CRF model guided by syntactic dependency rules for structured aspect-opinion extraction.
- **Peng-Two-Stage** [3]: Separately extracts aspect-opinion pairs and sentiment, followed by a classifier to match them into complete triplets.

2) End-to-end methods:

- **JET-BERT** [7]: Introduces a position-aware tagging scheme with factorized features to jointly model triplet components.
- **Span-ASTE** [5]: Enumerates candidate spans and models their interactions, using dual-channel pruning to reduce computation.
- **EMC-GCN** [8]: Applies a multi-channel GCN over ten relation types with biaffine attention for contextualized triplet extraction.
- **SSJE** [29]: Performs span-level enumeration and matching to jointly identify multi-word aspects and opinions via structured decoding.
- **DGCNAP** [9]: Enhances GCN with position-aware attention and affective knowledge to better model sentiment reasoning.
- **SA-Transformer** [10]: Incorporates syntax-aware attention into a transformer, distinguishing dependency edge types for relation modeling.
- **STAGE** [6]: A span-level model allowing each span to assume multiple roles while employing greedy inference to assemble triplets.
- **Biston** [30]: Leverages syntactic distance signals (sequence, constituency, dependency) within a transformer to handle overlapping triplets.
- **DRN** [22]: A dual-relation encoder model with multi-channel GCNs and criss-cross attention for entity extraction and matching.
- **SE-TRDF** [31]: Detects triplet regions enclosed by aspect and opinion words and classifies their sentiment, improving multi-word aspect and opinion extraction with a sentiment detection module.

D. Experimental Results

Table I summarizes the overall results, demonstrating that our proposed SENGRA model achieves the highest F1 scores across all four benchmark datasets. This performance highlights its robust capability in aspect sentiment triplet extraction.

In comparison to existing syntax-aware models such as EMC-GCN, DGCNAP, and Biston, SENGRA consistently delivers significant improvements, confirming the efficacy of our G-RGAT in selectively and dynamically employing syntactic information for more effective structural representations. On the 14lap dataset, SENGRA surpasses the best syntax-aware baseline by 3.71% in F1 score. On the 14res dataset, it achieves a 2.06% improvement over EMC-GCN, which highlights the effectiveness of our approach in leveraging syntactic structures in a more adaptable manner. The improvements on the 15res and 16res datasets are even more pronounced, with gains of 5.15% and 4.74% respectively over their best-performing syntax-aware counterparts.

Additionally, SENGRA exceeds the performance levels of mainstream span-based and joint extraction methods such as Span-ASTE and SSJE. These findings suggest that, beyond merely capturing entity spans, SENGRA's capacity to selectively amplify task-relevant syntactic relations results in the formation of more accurate triplets.

E. Ablation Study

To evaluate the impact of each component in SENGRA, we performed ablation experiments on four benchmark datasets, as presented in Table II.

First, replacing the syntax-guided inference module with a simple greedy strategy (**w/o SynInference**) leads to a slight but consistent performance decline across all datasets.

TABLE I
PERFORMANCE COMPARISON ON BENCHMARK DATASETS

Model	14lap			14res			15res			16res		
	P	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1	*P*	*R*	F_1
CMLA+	30.09	36.92	33.16	39.18	47.16	42.79	34.56	39.84	37.01	41.34	42.10	41.72
RINANTE+	21.71	18.66	20.07	31.42	39.38	34.95	29.88	30.06	29.97	25.68	22.30	23.87
Peng-two-stage	37.38	50.38	42.87	43.24	63.66	51.46	48.07	57.51	52.32	46.96	64.24	54.21
JET-BERT	55.39	47.33	51.04	70.56	55.94	62.40	64.45	51.96	57.53	70.42	58.37	63.83
Span-ASTE	63.44	55.84	59.38	72.89	70.89	71.85	62.18	64.45	63.27	69.40	71.17	70.26
EMC-GCN#	61.46	55.56	58.32	71.85	72.12	71.98	59.89	61.05	60.38	65.08	71.66	68.18
SSJE	67.33	54.71	60.41	73.12	71.43	72.26	63.94	66.17	<u>65.05</u>	70.82	72.00	71.38
DGCNAP#	57.57	62.02	53.79	70.72	72.90	68.69	61.19	62.23	60.21	69.58	69.75	69.44
SA-Transformer#	61.28	48.98	54.44	70.76	65.85	68.22	62.82	58.31	60.48	72.01	62.87	67.13
STAGE-3D	71.98	53.86	<u>61.58</u>	78.58	69.58	<u>73.76</u>	73.63	57.90	64.79	76.67	70.12	<u>73.24</u>
Biston#	65.99	53.59	59.15	70.03	68.41	69.21	65.25	56.91	60.79	66.85	69.84	68.32
DRN#	66.99	52.61	58.94	75.24	64.49	69.45	68.61	55.47	61.34	73.30	64.42	68.57
SE-TRDF	66.89	48.89	56.49	73.03	65.84	69.24	69.47	54.43	61.04	73.02	64.78	68.65
Our SENGRA#	67.30	58.96	**62.86**	77.33	71.03	**74.04**	69.82	62.47	**65.94**	76.07	72.37	**74.18**

The highest F1 scores are indicated in bold, while the second highest are underlined.
The symbol "#" denotes models that are syntax-aware.

TABLE II
F1 SCORES OF ABLATION STUDY

Model	14lap	14res	15res	16res
SENGRA	**62.86**	**74.04**	**65.94**	**74.18**
w/o SynInference	62.66	73.73	65.51	73.77
w/o StrGate	61.68	73.18	65.13	73.19
w/o SynEncoder	61.44	72.82	64.29	72.67

TABLE III
PERFORMANCE COMPARISON OF DIFFERENT DEPENDENCY PARSERS

Parser	14lap	14res	15res	16res
Stanza	**62.86**	**74.04**	**65.94**	**74.18**
en_core_web_trf	62.42	73.77	65.39	73.61
en_core_web_sm	61.22	72.90	64.43	72.52

This suggests that incorporating dependency structures during decoding helps disambiguate triplets and contributes to higher extraction precision.

Further, removing the structure-aware gating mechanism (**w/o StrGate**) leads to a more noticeable performance drop, especially on the 14lap dataset. This indicates that the gate effectively refines the aggregated syntactic features, enabling a more adaptive integration of structural information to better handle the complex dependency patterns.

Finally, the most significant performance degradation occurs when the G-RGAT syntactic encoder is completely removed (**w/o SynEncoder**). This sharp decrease across all benchmarks demonstrates that syntactic information is fundamental for capturing fine-grained relations between aspects and opinions. Without syntactic constraints, the model becomes more susceptible to confusion when handling complex or long-range dependencies, leading to less reliable triplet identification.

F. Impact of Dependency Parsers

To evaluate the impact of external syntactic tools, we compare three parsers: Stanza [32], en_core_web_trf, and en_core_web_sm. Stanza utilizes BiLSTM and biaffine attention for dependency parsing. The latter two are spaCy models, where *trf* employs Transformer-based encoders and *sm* relies on a lightweight feature representation network.

As shown in Table III, SENGRA achieves the best performance when using Stanza, indicating that higher parsing accuracy positively affects downstream extraction. The results for en_core_web_trf are closely competitive with Stanza. In contrast, en_core_web_sm yields relatively lower scores, reflecting its limitations in capturing complex structural dependencies. Overall, while SENGRA's performance is influenced by the quality of external tools, it maintains competitive results across all mainstream parsers, demonstrating the model's robustness to varying levels of syntactic noise.

G. Case Study

To intuitively demonstrate the extraction capabilities of SENGRA, we randomly selected four sentences from the test set for case analysis, focusing on both comparative performance and internal mechanisms.

1) Comparative Analysis: We compared SENGRA with the STAGE model across three typical examples, as shown in Table IV.

As demonstrated, STAGE fails to extract the correct triplet in the first instance as it lacks structural guidance to associate the verb with the correct aspect. In contrast, SENGRA success-

TABLE IV
CASE STUDIES OF STAGE AND SENGRA ON BENCHMARK DATASETS

Sentence	Ground Truth	STAGE	SENGRA (Ours)
The food is all shared so we get to order together and eat together.	(food, shared, Pos)	*NULL*	(food, shared, Pos)
lobster was good, nothing spectacular.	(lobster, good, Neu) (lobster, nothing spectacular, Neu)	(lobster, good, Pos) (lobster, nothing spectacular, Pos)	(lobster, good, Neu) (lobster, nothing spectacular, Neu)
I took one bite from the $24 salmon, and I have never... tasted salmon as fishy, as dry, and as bland as the one in Flatbush Farms.	(salmon, fishy, Neg) (salmon, dry, Neg) (salmon, bland, Neg)	(salmon, fishy, Neg) (salmon, dry, Neg)	(salmon, fishy, Neg) (salmon, dry, Neg) (salmon, bland, Neg)

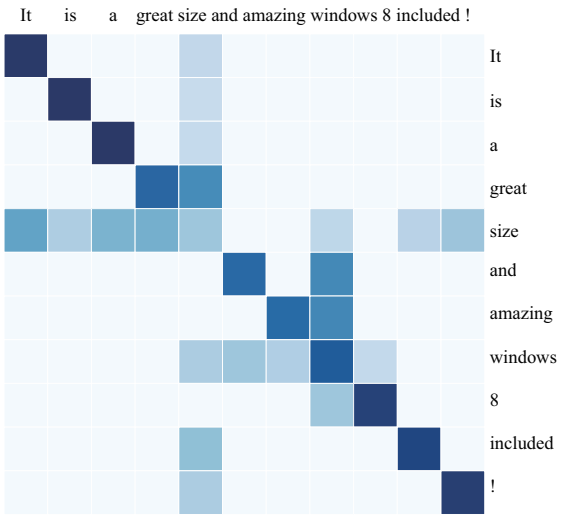


Fig. 4. Visualization of G-RGAT attention matrix.

fully links "shared" with "food" by utilizing syntactic cues. In the second example, SENGRA accurately classifies the sentiment as neutral, whereas STAGE erroneously categorizes it as positive. This indicates that SENGRA's syntax-aware span representation more effectively captures subtle sentiment distinctions. Lastly, in a lengthy and complex sentence involving "salmon", SENGRA extracts all negative triplets, whereas STAGE fails to extract one. This underscores the advantage of SENGRA in managing long-range dependencies using syntactic structures. These examples underscore SENGRA's robustness in aligning sentiment expressions with their corresponding targets through structurally informed representations.

2) Attention Mechanism Analysis: To further explore the internal workings of the model, we visualized the attention distribution of the G-RGAT layer for the sentence: "It is a great size and amazing windows 8 included!" The attention heatmap in Fig. 4 illustrates the model's focus during syntactic relation modeling.

The model effectively captures the structural associations among multiple positive sentiment expressions. In the phrase "great size", a strong attention linkage is observed between the opinion term "great" and the aspect term "size". Meanwhile, in "amazing windows 8 included", the model exhibits precise structural awareness: the opinion "amazing" is strongly associated with the core aspect "windows", while the dependency between "windows" and "8" enables the model to correctly recognize "windows 8" as a complete multi-word aspect.

H. Limitations

Despite the strong empirical performance of SENGRA across multiple benchmarks, several limitations remain. First, the proposed model relies on external dependency parsers to construct syntactic graphs. In informal or grammatically irregular texts, parsing errors may propagate through subsequent modules and accumulate, potentially affecting the overall extraction performance. Although the proposed gating mechanism alleviates this issue to some extent, it does not completely eliminate the dependency on parser quality.

Second, the introduction of the G-RGAT module enhances the model's ability to capture structured dependencies, but also increases computational overhead during training. Specifically, relation-aware attention requires interactions over neighboring nodes and typed dependency relations. When processing long sentences or syntactically dense dependency graphs, both the parameter scale and GPU memory consumption grow accordingly, which may reduce training efficiency and limit practical deployment in resource-constrained scenarios.

V. CONCLUSION

In this paper, we propose SENGRA, a syntax-enhanced span tagging network for aspect sentiment triplet extraction. By incorporating typed syntactic dependencies through a gated relational graph attention mechanism, the proposed model strengthens the modeling of key structural relations while suppressing syntactic noise. In addition, a syntax-guided inference strategy is introduced to improve the efficiency of aspect–opinion matching while maintaining extraction accuracy during triplet construction.

Experimental results on four public benchmark datasets demonstrate that SENGRA consistently outperforms state-of-the-art methods. Further ablation studies verify the effec-

tiveness and complementarity of the syntactic encoding and inference components. Overall, this work provides a new perspective on integrating structural information and semantic representations for aspect sentiment triplet extraction. The core mechanisms of SENGRA are not limited to this task and have the potential to be extended to other span-based structured information extraction scenarios.

Future work will explore more lightweight syntactic integration strategies to improve computational efficiency while maintaining performance advantages. In addition, enhancing the robustness of the model against syntactic parsing errors remains an important direction to further improve its applicability in real-world settings.

REFERENCES

[1] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar, "SemEval-2014 task 4: Aspect based sentiment analysis," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, Ireland, Aug. 2014, pp. 27–35.

[2] X. Li and W. Lam, "Deep multi-task learning for aspect term extraction with memory interaction," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark, Sep. 2017, pp. 2886–2892.

[3] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, and L. Si, "Knowing what, how and why: A near complete solution for aspect-based sentiment analysis," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8600–8607, Apr. 2020.

[4] Z. Wu, C. Ying, F. Zhao, Z. Fan, X. Dai, and R. Xia, "Grid tagging scheme for aspect-oriented fine-grained opinion extraction," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Online, Nov. 2020, pp. 2576–2585.

[5] L. Xu, Y. K. Chia, and L. Bing, "Learning span-level interactions for aspect sentiment triplet extraction," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online, Aug. 2021, pp. 4755–4766.

[6] S. Liang, W. Wei, X.-L. Mao, Y. Fu, R. Fang, and D. Chen, "STAGE: Span Tagging and Greedy Inference Scheme for Aspect Sentiment Triplet Extraction," *arXiv e-prints*, p. arXiv:2211.15003, Nov. 2022.

[7] L. Xu, H. Li, W. Lu, and L. Bing, "Position-aware tagging for aspect sentiment triplet extraction," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, Nov. 2020, pp. 2339–2349.

[8] H. Chen, Z. Zhai, F. Feng, R. Li, and X. Wang, "Enhanced multi-channel graph convolutional network for aspect sentiment triplet extraction," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland, May 2022, pp. 2974–2985.

[9] Y. Li, Q. He, and D. Zhang, "Dual graph convolutional networks integrating affective knowledge and position information for aspect sentiment triplet extraction," *Frontiers in Neurorobotics*, vol. 17, 2023.

[10] L. Yuan, J. Wang, L.-C. Yu, and X. Zhang, "Encoding syntactic information into transformers for aspect-based sentiment triplet extraction," *IEEE Transactions on Affective Computing*, vol. 15, no. 2, pp. 722–735, 2024.

[11] Z. Qiu, Z. Wang, B. Zheng, Z. Huang, K. Wen, S. Yang, R. Men, L. Yu, F. Huang, S. Huang, D. Liu, J. Zhou, and J. Lin, "Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free," in *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[12] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam, "A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges," *arXiv e-prints*, p. arXiv:2203.01054, Mar. 2022.

[13] M. Wankhade, C. Kulkarni, and A. C. S. Rao, "A survey on aspect base sentiment analysis methods and challenges," *Applied Soft Computing*, vol. 167, p. 112249, 2024.

[14] Y. Yin, F. Wei, L. Dong, K. Xu, M. Zhang, and M. Zhou, "Unsupervised Word and Dependency Path Embeddings for Aspect Term Extraction," *arXiv e-prints*, p. arXiv:1605.07843, May 2016.

[15] Z. Chen and T. Qian, "Enhancing aspect term extraction with soft prototypes," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Online, Nov. 2020, pp. 2107–2117.

[16] B. Yang and C. Cardie, "Extracting opinion expressions with semi-Markov conditional random fields," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Jeju Island, Korea, Jul. 2012, pp. 1335–1345.

[17] Z. Wu, C. Ying, F. Zhao, Z. Fan, X. Dai, and R. Xia, "Grid Tagging Scheme for Aspect-oriented Fine-grained Opinion Extraction," *arXiv e-prints*, p. arXiv:2010.04640, Oct. 2020.

[18] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, Nov. 2016, pp. 606–615.

[19] Z. Li, Y. Wei, Y. Zhang, X. Zhang, and X. Li, "Exploiting coarse-to-fine task transfer for aspect-level sentiment classification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 4253–4260, Jul. 2019.

[20] S. Wu, H. Fei, Y. Ren, D. Ji, and J. Li, "Learn from Syntax: Improving Pair-wise Aspect and Opinion Terms Extractionwith Rich Syntactic Knowledge," *arXiv e-prints*, p. arXiv:2105.02520, May 2021.

[21] L. Gao, Y. Wang, T. Liu, J. Wang, L. Zhang, and J. Liao, "Question-driven span labeling model for aspect–opinion pair extraction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 14, pp. 12 875–12 883, May 2021.

[22] T. Xia, X. Sun, Y. Yang, Y. Long, and R. Sutcliffe, "A dual relation-encoder network for aspect sentiment triplet extraction," *Neurocomputing*, vol. 597, p. 128064, 2024.

[23] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, Jun. 2019, pp. 4171–4186.

[24] X. Bai, P. Liu, and Y. Zhang, "Investigating Typed Syntactic Dependencies for Targeted Sentiment Classification Using Graph Attention Neural Network," *arXiv e-prints*, p. arXiv:2002.09685, Feb. 2020.

[25] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1724–1734.

[26] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. Loukachevitch, E. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, and G. Eryiğit, "SemEval-2016 task 5: Aspect based sentiment analysis," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, San Diego, California, Jun. 2016, pp. 19–30.

[27] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "SemEval-2015 task 12: Aspect based sentiment analysis," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, Denver, Colorado, Jun. 2015, pp. 486–495.

[28] H. Dai and Y. Song, "Neural aspect and opinion term extraction with mined rules as weak supervision," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, Jul. 2019, pp. 5268–5277.

[29] Y. Li, Y. Lin, Y. Lin, L. Chang, and H. Zhang, "A span-sharing joint extraction framework for harvesting aspect sentiment triplets," *Knowledge-Based Systems*, vol. 242, p. 108366, 2022.

[30] S. Hao, Y. Zhou, P. Liu, and S. Xu, "Bi-syntax guided transformer network for aspect sentiment triplet extraction," *Neurocomputing*, vol. 594, p. 127880, 2024.

[31] X. Liu, J. Chen, C. Xu, Z. Du, W. Chen, and H. Li, "Sentiment-enhanced triplet region detection framework for aspect sentiment triplet extraction," *Neurocomputing*, vol. 644, p. 130392, 05 2025.

[32] P. Qi, Y. Zhang, Y. Zhang, J. Bolton, and C. D. Manning, "Stanza: A python natural language processing toolkit for many human languages," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, A. Celikyilmaz and T.-H. Wen, Eds. Association for Computational Linguistics, Jul. 2020, pp. 101–108.