Train Once for All: A Transitional Approach for Efficient Aspect Sentiment Triplet Extraction

Xinmeng Hou[1], **Lingyue Fu**[2], **Chenhao Meng**[2], **Kounianhua Du**[2], **Wuqi Wang**[3], **Hai Hu**[4*]
[1]Columbia University, New York, NY, USA
[2]Shanghai Jiao Tong University, Shanghai, China
[3]Chang'an University, Xi'an, China
[4]City University of Hong Kong, Hong Kong SAR, China
fh2450@tc.columbia.edu, fulingyue@sjtu.edu.cn, chenhaomeng@sjtu.edu.cn,
kounianhuadu@sjtu.edu.cn, wuqiwang@chd.edu.cn, hu.hai@cityu.edu.hk

Abstract

Aspect-Opinion Pair Extraction (AOPE) and Aspect Sentiment Triplet Extraction (ASTE) have drawn growing attention in NLP. However, most existing approaches extract aspects and opinions independently, optionally adding pairwise relations, often leading to error propagation and high time complexity. To address these challenges and being inspired by transition-based dependency parsing, we propose the first transition-based model for AOPE and ASTE that performs aspect and opinion extraction jointly, which also better captures position-aware aspect-opinion relations and mitigates entity-level bias. By integrating contrastive-augmented optimization, our model delivers more accurate action predictions and jointly optimizes separate subtasks in linear time. Extensive experiments on four commonly used ASTE/AOPE datasets show that, our proposed transition-based model outperform previous models on two out of the four datasets when trained on a single dataset. When multiple training sets are used, our proposed method achieves new state-of-the-art results on all datasets. We show that this is partly due to our model's ability to benefit from transition actions learned from multiple datasets and domains. Our code is available at https://github.com/Paparare/trans_aste.

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a fine-grained sentiment analysis task that identifies specific aspects in text and analyzes the sentiments linked to them (Hu and Liu, 2004; Liu, 2012; Wang et al., 2024). As shown in Figure 1, ABSA involves subtasks such as Aspect Extraction (AE) and Opinion Extraction (OE)—identifying mentioned aspects and their related opinions, or the combination—Aspect-Opinion Pair Extraction. Once the aspect and opinion have been extracted, a sentiment is usually computed, and this more complicated task is often referred to as Aspect-Sentiment Triplet Extraction (ASTE). For instance, given the sentence: "*Gourmet food is delicious. Good service, but not so welcoming*", AE identifies **gourmet food** and **service** as aspects, while OE extracts **delicious**, **good**, and **not so welcoming** as opinions. These outputs are then combined to form aspect-opinion pairs, with a separate sentiment tagging system assigning polarities to create triplets (Jiang et al., 2023; Wang et al., 2023; Chakraborty, 2024).

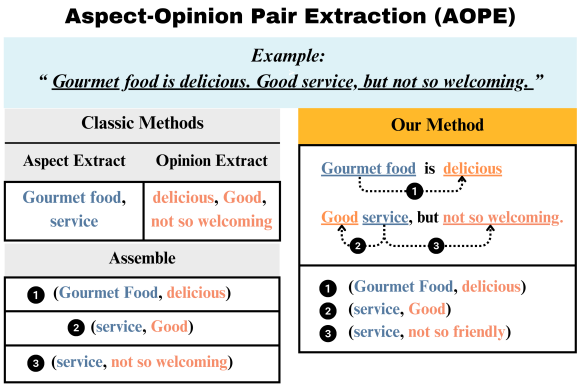


Figure 1: Demonstration of the processing steps in both classic and transitional methods for extracting aspect-opinion pairs. Importantly, our proposed transitional method predicts transition actions, and performs pair extraction after the aspect–opinion relationship has been established, allowing the model to capture contextual relationships more effectively.

ASTE is the most integrated task for aspect-based sentiment analysis, for which diverse models leveraging various methodologies have been developed, including pipeline-based approach (Peng et al., 2020), sequence-to-sequence method (Yan et al., 2021), sequence-tagging method (Wu et al., 2020; Xu et al., 2020), to name just a few. Despite these efforts and growing interests, the accuracy of recent models remains suboptimal, with the best systems scoring 60% or 70% (Sun et al.,

* Corresponding Author: Hai Hu.

Findings of the Association for Computational Linguistics: EMNLP 2025, pages 6706–6719
November 4-9, 2025

2024). There are two key challenges that hinder performance: (1) **Disconnected Aspect-Opinion Extraction**: Opinions are often extracted independently from their corresponding aspects (Liang et al., 2023; Sun et al., 2024). While positional relationships can be added as an auxiliary factor to assist pair extraction (Liu et al., 2022; Wang et al., 2023), this approach loses critical contextual information by treating aspects and opinions as separate entities. This limits the effectiveness of many token-based extraction methods. (2) **High time complexity with longer sequences**: Methods using 2D matrix tagging (Liang et al., 2023; Sun et al., 2024) to capture relationships between tokens face significant increases in time complexity as the length of the token sequence increases. This computational burden restricts their scalability, especially for longer texts in practical applications.

To address these two challenges, we present the first transition-based AOPE system named **Trans-AOPE** that (1) extracts the Aspect and the Opinion at the same time, and (2) has a time complexity of $O(n)$. We also introduce a contrastive-augmented optimization method to enhance model efficiency. We conduct experiments on 4 commonly used ABAS datasets, and compare our system with previous models. Our results show that **Trans-model** achieves state-of-the-art performance on all datasets we tested. We conduct comprehensive ablation studies to evaluate the contribution of optimization components and perform extensive training on various datasets to identify precisely where our model and baselines derive their learning.

Our contributions are: (1) We propose the first transition-based model that extracts aspect-opinion *pairs* based on relational aspects, rather than using relational factors as supplementary references or confirmation, with linear time complexity. (2) We experiment with a contrastive-augmented optimization method and find that balanced weighting yields faster, more stable improvements, emerging as the optimal training configuration. (3) We explore various training strategies and show that our proposed method achieves optimal performance on four datasets when trained on combined training sets, with better cross-dataset generalization.

2 Related Work

Previous methods on ASTE Pipeline-based approaches, such as Peng-Two-stage (Peng et al., 2020), decompose the task into multiple stages for modular refinement. Sequence-to-sequence frameworks like BARTABSA (Yan et al., 2021) employ pretrained transformers to generate triplets flexibly. Sequence-tagging methods, including GTS (Wu et al., 2020) and JET-BERT (Xu et al., 2020), annotate tokens for precise identification of relationships. Machine Reading Comprehension (MRC)-based models, such as COM-MRC (Zhai et al., 2022) and Triple-MRC (Zou et al., 2024), reframe the task as query answering for efficient extraction. Graph-based approaches such as EMC-GCN (Chen et al., 2022), BDTF (Chen et al., 2022), and DGCNAP (Li et al., 2023) use graph structures to capture semantic and syntactic interactions. Tagging schema-based models, exemplified by STAGE-3D (Liang et al., 2023), use hierarchical schemas for multi-level extraction, while lightweight models like MiniConGTS (Sun et al., 2024) focus on efficiency with reduced computational costs.

Table 1 summarizes these baseline methods, along with our proposed model, in terms of their core approaches and time complexities.

Transition-based Methods in NLP Transition-based approaches are widely used in dependency parsing, leveraging shift-reduce and bidirectional arc actions (left-arc, right-arc) for efficient $O(n)$ parsing (Aho and Ullman, 1973; Nivre, 2003; Cer et al., 2010). These parsers maintain stack, buffer, and arc relations to track transitions and then build up dependency relations between tokens.

Transition-based methods have also been applied to various NLP tasks, including token segmentation (Zhang et al., 2016), argument mining (Bao et al., 2021), constituency parsing (Yang and Deng, 2020), AMR parsing (Zhou et al., 2021), and sequence labeling (Gómez-Rodríguez et al., 2020), among others. Transition-based methods have been explored in emotion analysis (Fan et al., 2020; Jian et al., 2024). In sentiment analysis, however, transition-based models have not been widely adopted. One exception is their use in generating graph structures for opinion extraction (Fernández-González, 2023), although this design relies on graph embeddings and thus results in a time complexity of $O(N^2)$, with performance that lags behind more recent AOPE and ASTE approaches.

Contrastive-based Optimization Contrastive learning powers state-of-the-art token-independent extraction (MiniconGTS (Sun et al., 2024)), im-

Method	Approach	Time Complexity
Peng-Two-stage (Peng et al., 2020)	Two-Stage Pipeline: entity identification and relation formation	$O(n + k^2)$
BARTABSA (Yan et al., 2021)	Generative-based Aspect-based Sentiment Analysis	$O(m \cdot v)$
GTS (Wu et al., 2020)	Grid Matrix-based Tagging	$O(n^2)$
JET-BERT (Xu et al., 2020)	Position-Aware Sequence Tagging	$O(n)$
COM-MRC (Zhai et al., 2022)	Compositional Machine Reading Comprehension	$O(r \cdot n^2 \cdot h)$
Triple-MRC (Zou et al., 2024)	Multi-turn Machine Reading Comprehension	$O(r \cdot n^2 \cdot h)$
EMC-GCN (Chen et al., 2022)	Multi-channel Graph Convolutional Network	$O(m \cdot n^2 \cdot h)$
DGCNAP (Li et al., 2023)	Graph Convolutional Network w/ Affective Knowledge	$O(m \cdot n^2 \cdot h)$
MiniConGTS (Sun et al., 2024)	Lightweight Grid Matrix-based Tagging System	$O(n^2)$
Trans-model (Ours)	Transition-based Action Prediction for Simulating Relation Formation and Pair Extraction	$O(n)$

Table 1: An overview of previous methods and models (which will serve as baselines in this study), their approaches, and corresponding time complexities. Here, the hidden size for LSTM d is simplified; n is the sequence length; m is the number of graph channels; v is the vocabulary size; k is the number of extracted terms; r is the number of query rounds, and h is the hidden size of the encoder.

proves few-shot prompt learners via view augmentation (Jian et al., 2022), supplies a principled loss for goal-conditioned RL (Eysenbach et al., 2023), and benefits from margin studies that stress positive-sample weighting (Rho et al., 2023). Its versatility prompts us to embed a contrastive loss in our transition-based AOPE and ASTE, sharpening representations and boosting accuracy.

3 The Trans-AOPE/ASTE Model

We recast aspect–opinion extraction as a parsing-guided graph-construction problem in two stages: Trans-AOPE incrementally extracts aspect–opinion pairs from context-rich inputs, and Trans-ASTE tags the recovered pairs. The parser tracks five working structures—stack, buffer, aspect set, opinion set, and pair set—extending the three used in earlier systems and enabling joint recovery of aspects and opinions.

3.1 Transitional Operations and State Change

Phrase relations are modeled as directed edges between two tokens N_1 and N_2. We denote a rightward (aspect-to-opinion) link by $RR : N_1 \xrightarrow{l} N_2$ and a leftward link by $LR : N_1 \xleftarrow{l} N_2$, where $l \in \{l_L, l_R\}$ covers causal (bidirectional) labels. Aspect (A) and opinion (O) spans may contain several tokens (e.g., *gourmet food*, *not bad*), so merge operations are allowed.

For the ASTE task we use seven transition actions that (i) retrieve tokens, (ii) terminate, or (iii) merge spans. Each parser state is the tuple $T = (\sigma, \beta, A, O, R)$ of stack, buffer, current aspect, current opinion and accumulated relations. Default actions are always available, primary actions create or merge spans, and secondary actions add relations once the relevant spans exist. Verbal and symbolic definitions of every action follow.

Default Actions:

1. ***Shift*** (SF) moves a token from the tokenized stack into the buffer for further processing.
2. ***Stop*** (ST) halts the process when only one token remains in the buffer, and the stack is empty.

Primary Actions:

1. ***Merge*** (M) combines multiple tokens in the buffer into a single compound target.
2. ***Left Constituent Removal*** (L_n) removes the left constituent from the buffer.
3. ***Right Constituent Removal*** (R_n) removes the right constituent from the buffer.

Secondary Actions:

1. ***Left-Relation Formation*** (LR) creates a relation from the right aspect constituent to the left opinion constituent.
2. ***Right-Relation Formation*** (RR) creates a relation from the left aspect constituent to the right opinion constituent.

Table 2 provides a symbolic illustration of how the symbolic state is constructed and utilized. Take the sentence "Gourmet food is delicious" as an example. Table 3 demonstrates the process of moving tokens from the buffer to the stack, deciding whether they should be merged into a single entity or removed, and finally evaluating them for relation formation. It is important to note that the set of actions shown in the figure is not the only way to extract the "Gourmet food" and "delicious" aspect-opinion pair. An alternative approach use the stack's capacity of holding multiple tokens, moving "is" to the stack ($\beta_3 \rightarrow \sigma_3$) before merging "Gourmet" and "food" ($[\sigma_1, \sigma_2, \sigma_3] \rightarrow [\sigma_{1\&2}, \sigma_3]$).

Action	Symbolic Expression
Shift (SF)	$(\sigma_0,\ \beta_0 \mid \beta_1, A, O, R) \xrightarrow{SH} (\sigma_0 \mid \sigma_1, \beta_1, A, O, R)$
Stop (ST)	$(\sigma_0, , A, O, R) \xrightarrow{ST} (, , A, O, R)$
Merge (M)	$(\sigma_0 \mid \sigma_1,\ \beta_1 \mid \beta_2, A, O, R) \xrightarrow{M} (\sigma_{0\&1},\ \beta_1 \mid \beta_2, A, O, R)$
Left Constituent Removal (L_n)	$(\sigma_0 \mid \sigma_1,\ \beta_0, A, O, R) \xrightarrow{L_n} (\sigma_1,\ \beta_0, A, O, R)$
Right Constituent Removal (R_n)	$(\sigma_0 \mid \sigma_1,\ \beta_0, A, O, R) \xrightarrow{R_n} (\sigma_0,\ \beta_0, A, O, R)$
Left-Relation Formation (LR)	$(\sigma_0 \mid \sigma_1, \beta_0, A, O, R) \xrightarrow{LR} (\sigma_0 \mid \sigma_1, \beta_0, A \cup \sigma_1, O \cup \sigma_0, R \cup \sigma_0 \leftarrow \sigma_1)$
Right-Relation Formation (RR)	$(\sigma_0 \mid \sigma_1, \beta_0, A, O, R) \xrightarrow{RR} (\sigma_0 \mid \sigma_1, \beta_0, A \cup \sigma_0, O \cup \sigma_1, R \cup \sigma_0 \rightarrow \sigma_1)$

Table 2: Symbolic Expressions for the Proposed Actions. Here, σ represents the stack, β represents the buffer, A denotes the aspect, O denotes the opinion, and R consist of an aspect and an opinion.

Phrase	Action	Stack (σ)	Buffer (β)	Aspect	Opinion	Pair
–	–	[]	$[\beta_1, \beta_2, \beta_3, \beta_4]$	–	–	–
1	SF	$[\sigma_1]$	$[\beta_2, \beta_3, \beta_4]$	–	–	–
2	SF	$[\sigma_1, \sigma_2]$	$[\beta_3, \beta_4]$	–	–	–
3	M	$[\sigma_{1\&2}]$	$[\beta_3, \beta_4]$	–	–	–
4	SF	$[\sigma_{1\&2}, \sigma_3]$	$[\beta_4]$	–	–	–
5	R_n	$[\sigma_{1\&2}]$	$[\beta_4]$	–	–	–
6	SF	$[\sigma_{1\&2}, \sigma_4]$	[]	–	–	–
7	RR	$[\sigma_{1\&2}, \sigma_4]$	[]	$[\sigma_{1\&2}]$	$[\sigma_4]$	$(\sigma_{1\&2} \rightarrow \sigma_4)$
9	ST	[]	[]	$[\sigma_{1\&2}]$	$[\sigma_4]$	$(\sigma_{1\&2} \rightarrow \sigma_4)$

Table 3: State changes for "Gourmet food is delicious" using symbolic representation. Here, σ_1 corresponds to "Gourmet", σ_2 to "food", σ_3 to "is", and σ_4 to "delicious". Similarly, β_1, β_2, β_3, and β_4 correspond to tokens in the buffer in sequence.

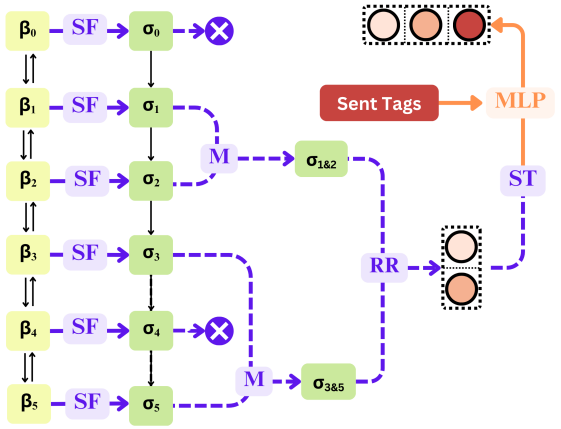


Figure 2: The complete process of the transition-based model is illustrated. Purple highlights represent the transition-based pair extraction actions, while orange indicates the final step of sentiment tagging.

3.2 Trans-AOPE State Representation

The model we propose consists of two core stages: pair extraction with a designed transitional action slot (in purple) and pair-based sentiment tagging (in orange), as illustrated in Figure 2.

In the first stage, the input, denoted as $I_1^n = (t_1, t_2, \ldots, t_n)$, is a sequence of tokens. The output is a sequence of actions, represented as $A_1^m = (a_1, a_2, \ldots, a_m)$. This process can be conceptualized as a search for the optimal action sequence, A^*, given the input sequence I_1^n. At each step n, the model predicts the next action based on the current system state, S, and the sequence of prior actions, A_1^{n-1} The updated system state, S_{n+1} , is determined by the specific action a_t. We define r_n as a symbolic representation for calculating the probability of the action a_n at step n. This probability is computed as follows:

$$p(a_n|r_n) = \frac{\exp(w_{a_n}^\top r_n + b_{a_n})}{\sum_{a' \in \mathcal{A}(S)} \exp(w_{a'}^\top r_n + b_{a'})} \quad (1)$$

Here, w_a is a learnable parameter vector, and b_a is a bias term. The set $\mathcal{A}(S)$ represents the legal actions available given the current parser state. The overall optimization objective for the model is defined as:

$$\begin{aligned}(A^*, S^*) &= \underset{A,S}{\operatorname{argmax}} \prod_n p(a_n, S_{n+1}|A_1^{n-1}, S_n) \\ &= \underset{A,S}{\operatorname{argmax}} \prod_n p(a_n|r_n)\end{aligned} \quad (2)$$

We recast ASTE as a transition-based action prediction problem. At each step the model, given the current state and action history, greedily selects the highest-probability action until parsing terminates. This yields an efficient parser that avoids

information leakage and supports flexible relation construction.

3.3 Transition Implementation with Neural Model

This section introduces a transition-based parsing process. RoBERTa (Liu et al., 2019) encodes the text, while UniLSTM (Hochreiter and Schmidhuber, 1997) and BiLSTM (Graves and Schmidhuber, 2005) capture transitions. The parser state evolves through a sequence of actions, with LSTMs processing each token once. This yields a time complexity of $O(n \cdot d^2)$, typically simplified to $O(n)$ under fixed d. Finally, an MLP classifies the sentiment for each pair or triplet based on the final parser state.

Token representations Consider the process of parsing a text $d_1^n = (p_1, p_2, \ldots, p_n)$, consisting of n phrases. Each phrase $p_i = (w_{i1}, w_{i2}, \ldots, w_{il})$ contains l tokens. A phrase can be represented as a sequence x_i = ([CLS], t_{i1}, …, t_{il}, [SEP]), where [CLS] is a special classification token whose final hidden state serves as the aggregate sequence feature, and [SEP] is a separator token. The hidden representation of each phrase is computed as h_{p_i} = RoBERTa(x_i) $\in \mathbb{R}^{d_b \times |l_i|}$, where d_b is the hidden dimension size, and $|l_i|$ is the length of the sequence x_i. Finally, the entire text d_1^n is represented as a list of tokens: $h_d = [h_{p_1}, h_{p_2}, \ldots, h_{p_n}]$.

State Initialization At the start of the parsing process, the parser's state is initialized as $(\beta = \varnothing, \sigma = [1, 2, \ldots, n], E = \varnothing, C = \varnothing, R = \varnothing)$, where σ is the stack, β is the buffer, and E, C, and R are empty sets representing different outputs. The state evolves through a sequence of actions, progressively consuming elements from the buffer β and constructing the output. This process continues until the parser reaches its terminal state when there is only one token left in buffer, represented as $(\beta = [SEP], \sigma = \varnothing, E, C, R)$.

Step-by-Step Parser State Representation. For the action sequence, each action a is mapped to a distributed representation e_a through a lookup table E_a. An unidirectional LSTM is then utilized to capture the complete history of actions in a left-to-right manner at each step t:

$$\alpha_t = \text{LSTM}_a(a_0, a_1, \ldots, a_{t-1}, a_t) \tag{3}$$

Upon generation of a new action a_t, its corresponding embedding e_{a_t} is integrated into the rightmost position of LSTM$_a$. To further refine the representation of the pair (σ_1, σ_0), their relative positional distance d is also encoded as an embedding e_d from a lookup table E_d. The composite representation of the parser state at step t encompasses these varied features.

The parser state is represented as a triple (β_s, σ_s, A_t), where σ_s denotes the stack sequence $(\sigma_0, \sigma_1, \ldots, \sigma_n)$, β_s represents the buffer sequence $(\beta_0, \beta_1, \ldots, \beta_n)$, and A_t encapsulates the action history $(a_0, a_1, \ldots, a_{t-1}, a_t)$. The stack (σ_n) and buffer (β_n) are encoded using bidirectional LSTMs as follows:

$$[s_t, b_t] = \text{BiLSTM}((\sigma_n, \beta_0), (\sigma_{n-1}, \beta_1), \ldots, (\sigma_0, \beta_n)) \tag{4}$$

Here, s_t and b_t are the output feature representations of the stack and buffer, respectively. Each of these representations consists of forward and backward components: $\sigma_t = (\overrightarrow{\sigma_t}, \overleftarrow{\sigma_t})$ and $\beta_t = (\overrightarrow{\beta_t}, \overleftarrow{\beta_t})$. The forward and backward components are matrices in $\mathbb{R}^{d_l \times |\sigma_t|}$ and $\mathbb{R}^{d_l \times |\beta_t|}$, respectively, where d_l is the hidden dimension size of the LSTM, and $|\sigma_t|$, $|\beta_t|$ are the lengths of the sequences σ_t and β_t.

3.4 Optimization Implementation

We compare two optimization strategies: regular optimization using Cross-Entropy Loss and contrastive-based optimization, which aligns predicted and true action embeddings. A weight study will evaluate the impact of the positioning of two components in augmented optimization on model performance. Both action and sentiment classification tasks are optimized using the Cross-Entropy Loss as base optimization, defined as follows:

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{M} y_i^j \log(p_i^j), \tag{5}$$

where N is the number of samples, M is the number of classes (either action or sentiment classes), y_i^j is a binary indicator (0 or 1) indicating whether class j is the correct class for sample i, and p_i^j is the predicted probability for class j for sample i. For AOPE task, total loss is action loss $\mathcal{L}_{\text{action}}$, and for ASTE task, is the sum of the losses for both tasks: $\mathcal{L}_{\text{base}} = \mathcal{L}_{\text{action}} + \mathcal{L}_{\text{sentiment}}$.

Given action and sentiment logits $\mathbf{A}_{\text{logits}}$ and ground-truth labels $\mathbf{A}_{\text{true}}$, the predicted actions are $\mathbf{A}_{\text{pred}} = \arg\max(\text{softmax}(\mathbf{A}_{\text{logits}}))$. Predicted and true actions are embedded as $\mathbf{E}_{\text{pred}} =$

$\text{Embed}(\mathbf{A}_{\text{pred}})$ and $\mathbf{E}_{\text{true}} = \text{Embed}(\mathbf{A}_{\text{true}})$. The cosine-similarity matrix is $S = \cos(\mathbf{E}_{\text{pred}}, \mathbf{E}_{\text{true}}) \in \mathbb{R}^{N \times N}$; its diagonal gives positive pairs. Define $e_{\text{pos}} = \exp(S \odot \mathbf{M}_{\text{pos}})$ and $e_{\text{all}} = \exp(S)$, where $\mathbf{M}_{\text{pos}}$ is the diagonal mask and $\odot$ denotes element-wise multiplication. Contrastive loss is computed as

$$\mathcal{L}_{\text{con}} = -\frac{1}{N} \sum_{i=1}^{N} \log \left(\frac{e_{\text{pos}}^{(i)}}{e_{\text{all}}^{(i)}} \right), \tag{6}$$

and the total loss is the addition of two weighted losses

$$\mathcal{L}_{\text{total}} = \omega_1 \, \mathcal{L}_{\text{base}} + \omega_2 \, \mathcal{L}_{\text{con}}. \tag{7}$$

4 Experimental Setups

4.1 Datasets and preprocessing

We benchmark on four standard ABSA datasets: **14lap** and **14res** (SemEval-2014)(Pontiki et al., 2014), **15res** (SemEval-2015)(Pontiki et al., 2015), and **16res** (SemEval-2016)(Pontiki et al., 2016); statistics are in Table6. **14lap** contains laptop reviews, while the others comprise restaurant reviews, and all are widely used for aspect-based sentiment extraction (Xu et al., 2021). For ASTE we follow the practice of previous studies like Sentic-GCN (Liang et al., 2021) and SK-GCN (Zhou et al., 2020) to construct sentiment-aware dependency graphs: SpaCy supplies syntactic edges (Honnibal et al., 2020), SenticNet provides sentiment weights (Cambria et al., 2017), and the resulting weighted adjacency matrices, paired with tokenized sentences and aspect–opinion–sentiment triplets, feed model training and evaluation.

4.2 Training settings

We experiment with two settings for training data. In this setting, we train with one of the four datasets and test on the test set of the *same* dataset (In-domain training), e.g., train on 14lap and test on 14lap. Since our method depends on the model learning the correct action to pair an aspect with an opinion from the training data, we hypothesize that it will have the advantage of being able to make use of training data from diverse domains to learn various actions. Thus we experiment with training on *two or more training sets combined* (Combined training), to observe whether there is performance gain when more actions are learned. Specifically, we train on several training sets together, and evaluate on a single test set.

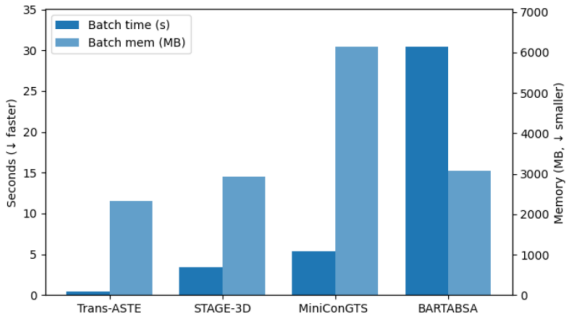


Figure 3: Mean batch time (s) and mean batch memory usage (MB) for Trans-ASTE, STAGE-3D, MiniConGTS, and BARTABSA

4.3 Baselines

We benchmark our model against three representative baselines—STAGE-3D (Liang et al., 2023), MiniConGTS (Sun et al., 2024), and BARTABSA (Yan et al., 2021). STAGE-3D is included through the scores reported in its original paper because no runnable code is available. MiniConGTS, the current state of the art for both ASTE and AOPE with quadratic complexity $O(n^2)$ (fixed constants omitted), is re-trained using the authors' public implementation; we keep all original hyper-parameters and preprocessing steps, modifying only the training data where necessary to incorporate every split (14lap, 14res, 15res, and 16res). BARTABSA, an earlier single-stage AOPE model of linear complexity $O(n)$, is re-trained under the same protocol. After re-training, each baseline is evaluated separately on the four test sets. Results for additional published baselines—CMLA, GTS, Triple-MRC, EMC-GCN, DGCNAP, and others—are presented in Appendix A.2.

5 Results and Analysis

We study wall-clock training time per batch and peak GPU memory per batch because these two metrics jointly determine *scalability* (how fast larger datasets can be processed) and *deployability* (whether the model fits on commodity GPUs). As shown in Figure 3, Trans-ASTE has the smallest footprint on both counts. Then, we present results obtained under various training-data configurations and compare our model with previous methods (Section 5.1). Finally, we assess the impact of adding a contrastive-loss term and identify the optimal loss configuration (Section 5.2), and section 5.3 provides a detailed discussion of these findings.

In-domain training setting	14res			14lap			15res			16res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
$O(n)$ | BARTABSA (Yan et al., 2021) ⋄	-	-	77.68	-	-	66.11	-	-	67.98	-	-	<u>77.38</u>
$O(n^2)$ | MiniConGTS (Sun et al., 2024) ⋆	-	-	<u>79.60</u>	-	-	<u>73.23</u>	-	-	<u>73.87</u>	-	-	76.29
$O(n)$ | Trans-AOPE (**Ours**)	78.89	65.98	71.86	66.31	55.51	60.43	93.20	85.31	89.08	78.16	81.73	79.91
Combined-train setting (Training Sets)												
MiniConGTS (14lap & 14res)	75.72	78.20	76.94	71.05	68.64	69.83	63.30	69.90	66.44	68.13	72.54	70.27
MiniConGTS (14res, 15res, & 16res)	78.68	76.78	77.72	56.86	48.31	52.23	94.54	92.48	93.50	77.65	75.22	76.42
MiniConGTS (14res, 14lap, 15res, & 16res)	78.22	77.11	77.66	76.32	64.19	69.74	93.35	91.99	92.67	76.27	76.79	76.53
BARTABSA (14res, 14lap, 15res & 16res)	75.40	76.76	76.07	72.36	63.40	67.59	93.56	**94.43**	**93.56**	87.06	**87.32**	87.19
Ours												
Trans-AOPE (14lap & 14res)	91.95	83.24	87.38	90.60	79.88	84.91	73.99	73.19	73.59	74.84	69.94	72.31
Trans-AOPE (14res, 15res & 16res)	74.34	62.29	67.78	91.03	78.11	84.07	95.66	88.04	91.69	90.63	80.65	85.35
Trans-AOPE (14res, 14lap, 15res & 16res)	**92.92**	**83.61**	**88.02**	**92.20**	**80.47**	**85.94**	**96.17**	90.94	93.48	**93.75**	84.82	**89.06**

Table 4: Comparison of different models on multiple datasets for AOPE task. Recall and precision values are omitted where they are not reported. The former best scores are underlined, and current best scores are bold. Highlights are used for analysis. ⋄ are retrieved from Yan et al., 2021. • is retrieved from Zhao et al., 2020, and ⋆ are retrieved from Sun et al., 2024

5.1 Main Results

On AOPE In the in-domain setting (upper half of Table 4), Trans-AOPE outperforms the baselines only on the 15res and 16res test sets.[1] Once we switch to the combined-train regime, Trans-AOPE eclipses MiniConGTS and BARTABSA in every train–test combination. Training on all four datasets boosts Trans-AOPE's F1 by roughly 20 points across the board—for example, from 71.86 $\rightarrow$ 88.02 on 14res and from 60.43 $\rightarrow$ 85.94 on 14lap—whereas MiniConGTS gains meaningfully only on 15res and even declines on 14res and 14lap (BARTABSA shows the same pattern). Domain mixing is especially telling on the laptop set: adding restaurant data catapults Trans-AOPE from 50.94 $\rightarrow$ 84.91, but drags MiniConGTS down from 73.23 $\rightarrow$ 69.83. Remarkably, even when trained only on the three restaurant corpora, Trans-AOPE still reaches 84.07 F1 on 14lap—virtually matching its 84.91 when 14lap is included—whereas MiniConGTS collapses to 52.23. Taken together, these results demonstrate that Trans-AOPE transfers knowledge across domains far more robustly than previous models.

On ASTE Table 5 echoes the pattern seen with AOPE. In the strict in-domain setup, Trans-ASTE trails on the 14res and 14lap test sets but outperforms its peers on 15res and 16res. Once the training data are pooled (combined-train), however, it outshines the baselines on every dataset. Simply adding 14lap to the 14res training set propels Trans-ASTE's F1 from 65.92 to 85.20, while MiniConGTS slips a bit (75.59 $\rightarrow$ 73.28) and BARTABSA gains only modestly (65.25 $\rightarrow$ 71.68). With all four corpora in the training mix, Trans-ASTE leads every test set except 15res—often by wide margins—and is the only model to secure a dramatic jump on 14lap (53.36 $\rightarrow$ 81.26). An exception appears when the model is trained on the three restaurant corpora and evaluated on 14lap (highlighted in yellow): Trans-ASTE plunges to 36.58 F1, far below the 84.07 recorded by its AOPE counterpart. We attribute this gap to domain-specific sentiment-polarity labels, whose transfer proves more fragile than the transfer of aspect–opinion spans themselves.

5.2 Study on Contrastive Loss

To investigate the impact of different weight configurations between the base loss and contrastive loss in Equation 7, we tested loss-weight ratios $w_{\text{base}} : w_{\text{con}} \in \{1{:}0, 1{:}1, 1{:}10, 0{:}1, 10{:}1\}$ with a batch size of 4. On the 14lap AOPE benchmark (Fig. 4), the balanced 1:1 setting climbed fastest and reached higher early F_1 values, while contrastive-only training stalled. ASTE showed the same pattern (Fig. 5), and results were consistent on additional datasets. We therefore adopt $w_{\text{base}} : w_{\text{con}} = 1{:}1$ in all remaining experiments.

5.3 When and Why is Trans-model Better?

When: Trans-model is better with combined training sets and in multi-domain generaliza-

[1] Scores for earlier models are taken from their original papers; we assume they were trained solely on the corresponding in-domain data, although most papers do not state this explicitly.

In-domain training setting	14res			14lap			15res			16Res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
$O(n)$ | BARTABSA (Yan et al., 2021) ⋄	65.52	64.99	65.25	61.41	56.19	58.69	59.14	59.38	59.26	66.60	68.68	67.62
$O(n^2)$ | STAGE-3D (Liang et al., 2023) ⋆	78.58	69.58	73.76	71.98	53.86	61.58	73.63	57.90	64.79	76.67	70.12	73.24
$O(n^2)$ | MiniConGTS (Sun et al., 2024) ⋆	76.10	75.08	75.59	66.82	60.68	63.61	66.50	63.86	65.15	75.52	74.14	74.83
$O(n)$ | Trans-ASTE (**Ours**)	72.25	60.61	65.92	61.49	47.13	53.36	91.30	83.05	86.98	77.95	79.90	77.95
Combined-train setting (Training Sets)												
MiniConGTS (14lap & 14res)	72.11	74.48	73.28	62.50	60.38	61.42	56.48	62.38	59.28	64.57	68.75	66.59
MiniConGTS (14res, 15res, & 16res)	74.64	72.84	73.73	52.12	44.28	47.88	92.31	90.29	91.29	74.42	72.10	73.24
MiniConGTS (14res, 14lap, 15res & 16res)	73.89	72.84	73.36	68.26	57.42	62.37	90.64	89.32	89.98	72.73	73.21	72.97
BARTABSA (14res, 14lap, 15res & 16res)	71.05	72.33	71.68	63.66	56.01	59.59	91.30	**92.99**	92.13	83.82	**84.07**	83.95
Ours												
Trans-ASTE (14lap &14res)	88.05	80.51	84.11	**88.11**	74.56	80.77	69.55	67.03	68.27	64.53	62.80	63.65
Trans-ASTE (14res, 15res & 16res)	73.53	61.74	67.12	42.25	32.25	36.58	89.89	86.96	88.40	81.19	77.08	79.08
Trans-ASTE (14res, 14lap, 15res & 16res)	**89.56**	**81.24**	**85.20**	86.87	**76.33**	**81.26**	**94.34**	90.58	**92.42**	**90.03**	83.33	**86.55**

Table 5: Comparison of different models on multiple datasets for ASTE task. Recall and precision values are omitted where they are not reported. Highlights are used for analysis. The former best scores are underlined, and current best scores are bold. ⋆ are retrieved from Sun et al., 2024, and ⋄ is retrieved from Yan et al., 2021.

tion. From the results above, it seems clear that the Trans-model in the two tasks are better when trained on multiple datasets combined, rather than one dataset alone. Our results also suggest that the added training data do not have to be in the same domain: when trained on 14lap & 14res and tested on either 14lap or 14res, the F1 score is at least about 10 percentage points better than trained on only one of the two datasets. It shows that our Trans-model is capable of learning from multiple domains, and seems to be able to transfer its knowledge from the laptop domain to the restaurant domain and vice versa, unlike MiniConGTS, which seems to be more sensitive to the domain of the data. However, further research is need to better understand the discrepancy between Trans-AOPE and Trans-ASTE models in cross-domain transfer, and improve the cross-domain transfer ability of the sentiment tagger.

Why: Trans-model can learn more actions in data from diverse domains. We believe that the proposed trans-models excel when trained on multiple datasets and show considerable generalization ability for two main reasons. First, by predicting actions instead of tokens, it avoids token-level biases and prevents overfitting on token-level patterns specific to restaurant datasets. Additionally, trans-models perform pair extraction after the aspect–opinion relationship is established, which enables the model to capture contextual relationships more effectively. These design decisions work together to significantly enhance the overall effectiveness and robustness of the trans-models.

6 Conclusion and Future Work

In this paper, we present an efficient transition pipeline for the extraction of aspects-opinion pairs with linear time complexity $O(n)$, enhanced by a contrastive-based optimization method. This approach obviates the need to directly identify and extract individual tokens, thereby mitigating token-level bias. It can be trained on a combination of diverse datasets that offers the most comprehensive coverage of the actions needed, resulting in significant performance improvements across various datasets. Specifically, training our model on a well-covered fused dataset enables it to learn robust action patterns, leading to superior performance on all datasets. Our model surpasses retrained baseline models on the same fused dataset, establishing new state-of-the-art results for both AOPE and ASTE tasks.

As transition-based methods have remained relatively less explored in sentiment-related tasks, we believe our work shows a promising direction to employ such methods in aspect-based sentiment analysis. Future work can further examine the potential of transition-based models in other sentiment analysis tasks, as well as the generalization ability of these models in situations of multi-domain data. It is also important to better understand the cross-domain and multi-domain generalization ability of transition models, by experimentation on more domains, since only two domains are involved in this work.

Limitation

Although the Trans-model demonstrates robust generalization capability, its reliance on larger datasets to effectively learn action patterns remains a notable limitation of the transition-based pipeline. This issue is evident in our results: while it is not necessary to use the *same* dataset for both training and testing, the model performs better when trained on blended datasets rather than on a single, limited one. Consequently, if the training data lack sufficient action patterns, the model's ability to handle nuanced or previously unseen contexts can be significantly compromised. These findings underscore the importance of training on a combined or broader dataset to enhance the model's overall effectiveness.

Acknowledgments

This project is funded by Shanghai Pujiang Program (22PJC063) awarded to Hai Hu.

References

Alfred V Aho and Jeffrey D Ullman. 1973. *The theory of parsing, translation, and compiling*, volume 1. Prentice-Hall Englewood Cliffs, NJ.

Jianzhu Bao, Chuang Fan, Jipeng Wu, Yixue Dang, Jiachen Du, and Ruifeng Xu. 2021. A neural transition-based model for argumentation mining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6354–6364.

Erik Cambria, Soujanya Poria, Alexander Gelbukh, and Mike Thelwall. 2017. Senticnet 5: Discovering conceptual primitives for sentiment analysis by means of context embeddings. In *Proceedings of AAAI*.

Daniel M Cer, Marie-Catherine De Marneffe, Daniel Jurafsky, and Christopher D Manning. 2010. Parsing to stanford dependencies: Trade-offs between speed and accuracy. In *LREC*. Floriana, Malta.

Abir Chakraborty. 2024. Aspect and opinion term extraction using graph attention network. *Preprint*, arXiv:2404.19260.

Hao Chen, Zepeng Zhai, Fangxiang Feng, Ruifan Li, and Xiaojie Wang. 2022. 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)*, pages 2974–2985.

Benjamin Eysenbach, Tianjun Zhang, Ruslan Salakhutdinov, and Sergey Levine. 2023. Contrastive learning as goal-conditioned reinforcement learning. *Preprint*, arXiv:2206.07568.

Chuang Fan, Chaofa Yuan, Jiachen Du, Lin Gui, Min Yang, and Ruifeng Xu. 2020. Transition-based directed graph construction for emotion-cause pair extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3707–3717.

Daniel Fernández-González. 2023. Structured sentiment analysis as transition-based dependency parsing. *Preprint*, arXiv:2305.05311.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610.

Carlos Gómez-Rodríguez, Michalina Strzyz, and David Vilares. 2020. A unifying theory of transition-based and sequence labeling parsing. *Preprint*, arXiv:2011.00584.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 168–177.

Yiren Jian, Chongyang Gao, and Soroush Vosoughi. 2022. Contrastive learning for prompt-based few-shot language learners. *Preprint*, arXiv:2205.01308.

Zhongquan Jian, Ante Wang, Jinsong Su, Junfeng Yao, Meihong Wang, and Qingqiang Wu. 2024. Emotrans: Emotional transition-based model for emotion recognition in conversation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5723–5733.

Baoxing Jiang, Shehui Liang, Peiyu Liu, Kaifang Dong, and Hongye Li. 2023. A semantically enhanced dual encoder for aspect sentiment triplet extraction. *Preprint*, arXiv:2306.08373.

Yanbo Li, Qing He, and Damin Zhang. 2023. Dual graph convolutional networks integrating affective knowledge and position information for aspect sentiment triplet extraction. *Frontiers in Neurorobotics*, 17:1193011.

Bin Liang, Hang Su, Lin Gui, Erik Cambria, and Ruifeng Xu. 2021. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, page 107643.

Shuo Liang, Wei Wei, Xian-Ling Mao, Yuanyuan Fu, Rui Fang, and Dangyang Chen. 2023. Stage: span tagging and greedy inference scheme for aspect sentiment triplet extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13174–13182.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Morgan & Claypool Publishers.

Yaxin Liu, Yan Zhou, Ziming Li, Dongjun Wei, Wei Zhou, and Songlin Hu. 2022. Mrce: A multi-representation collaborative enhancement model for aspect-opinion pair extraction. In *International Conference on Neural Information Processing*, pages 260–271. Springer.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13543–13551.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the Eighth International Conference on Parsing Technologies*, pages 149–160, Nancy, France.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8600–8607.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30, San Diego, California. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.

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

Daniel Rho, TaeSoo Kim, Sooill Park, Jaehyun Park, and JaeHan Park. 2023. Understanding contrastive learning through the lens of margins. *Preprint*, arXiv:2306.11526.

Qiao Sun, Liujia Yang, Minghao Ma, Nanyang Ye, and Qinying Gu. 2024. MiniConGTS: A near ultimate minimalist contrastive grid tagging scheme for aspect sentiment triplet extraction. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2817–2834, Miami, Florida, USA. Association for Computational Linguistics.

Chengwei Wang, Tao Peng, Yue Zhang, Lin Yue, and Lu Liu. 2023. Aopss: A joint learning framework for aspect-opinion pair extraction as semantic segmentation. In *Web and Big Data*, pages 101–113, Cham. Springer Nature Switzerland.

Pan Wang, Qiang Zhou, Yawen Wu, Tianlong Chen, and Jingtong Hu. 2024. Dlf: Disentangled-language-focused multimodal sentiment analysis. *arXiv preprint arXiv:2412.12225*.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for aspect-oriented fine-grained opinion extraction. *arXiv preprint arXiv:2010.04640*.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. *arXiv preprint arXiv:2010.02609*.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2021. Position-aware tagging for aspect sentiment triplet extraction. *Preprint*, arXiv:2010.02609.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2416–2429, Online. Association for Computational Linguistics.

Kaiyu Yang and Jia Deng. 2020. Strongly incremental constituency parsing with graph neural networks. *Preprint*, arXiv:2010.14568.

Zepeng Zhai, Hao Chen, Fangxiang Feng, Ruifan Li, and Xiaojie Wang. 2022. Com-mrc: A context-masked machine reading comprehension framework for aspect sentiment triplet extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3230–3241.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016. Transition-based neural word segmentation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 421–431.

He Zhao, Longtao Huang, Rong Zhang, Quan Lu, and Hui Xue. 2020. SpanMlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3239–3248, Online. Association for Computational Linguistics.

Jiawei Zhou, Tahira Naseem, Ramón Fernandez Astudillo, Young-Suk Lee, Radu Florian, and Salim Roukos. 2021. Structure-aware fine-tuning of sequence-to-sequence transformers for transition-based amr parsing. *Preprint*, arXiv:2110.15534.

Jie Zhou, Jimmy Xiangji Huang, Qinmin Vivian Hu, and Liang He. 2020. Sk-gcn: Modeling syntax and knowledge via graph convolutional network for aspect-level sentiment classification. *Knowledge-Based Systems*, 205:106292.

Wang Zou, Wubo Zhang, Wenhuan Wu, and Zhuoyan Tian. 2024. A multi-task shared cascade learning for aspect sentiment triplet extraction using bert-mrc. *Cognitive Computation*, pages 1–18.

A Appendix

A.1 Datasets Details

We conduct all experiments on the four benchmark corpora that originate from the SemEval Aspect-Based Sentiment Analysis (ABSA) shared tasks.[2] Table 6 summarises their key statistics. The two restaurant collections, 14res and 16res, are the largest, containing 2068 and 1393 sentences and 3909 and 2247 annotated aspect–sentiment–target triplets, respectively. 14lap covers the laptop domain and is both smaller and more sentiment-balanced: although it includes only 1453 sentences, the proportion of negative triplets (33 %) is comparable to the positive ones, reflecting the more critical tone of consumer-electronics reviews. 15res sits between the two 2014 datasets in size but exhibits the sparsest neutral category, with merely 61 neutral annotations out of 1747 triplets, making it effectively a polar dataset. Across all four corpora, positive opinions dominate (66–72 %), while neutral labels remain scarce; this imbalance motivates the macro-averaged metrics reported in Section 5.

[2] 14RES and 14LAP were released in SemEval-2014 Task 4, whereas 15RES and 16RES come from the 2015 and 2016 restaurant subtasks, respectively.

Datasets	#S	#POS	#NEU	#NEG	#T
14res	2068	2869	286	754	3909
14lap	1453	1350	225	774	2349
15res	1075	1285	61	401	1747
16res	1393	1674	90	483	2247

Table 6: Statistics of four datasets. #S denotes the number of sentence, #POS, #NEU, #NEG the number of positive, neutral and negative sentiment labels, and #T the total number of triplets.

A.2 Other Baselines

To provide a fuller historical context we also re-benchmark a broad set of earlier end-to-end systems—including CMLA (Wang et al., 2017), Peng-Two-stage (Peng et al., 2020), Dual-MRC (Mao et al., 2021), SpanMlt (Zhao et al., 2020), JET-BERT (Xu et al., 2020), COM-MRC (Zhai et al., 2022), Triple-MRC (Zou et al., 2024) under the same four-dataset protocol. Their precision, recall and F_1 scores for the AOPE and ASTE tasks are reported in Tables 7 and 8, respectively, which are placed in the appendix for completeness. These supplementary results serve as additional reference points but are not central to the main narrative of the paper.

A.3 Effect of Contrastive Loss

Figures 4 and 5 show that blending cross-entropy and contrastive loss with equal weight ($1:1$) provides the most effective training signal: the F1 curve climbs steeply from the earliest epochs, surpasses the cross-entropy-only baseline roughly two epochs sooner, and finishes with the highest scores on both aspect–opinion pair and triplet extraction. In contrast, weighting the objectives $10:1$ or $1:10$ slows this ascent and trims the final performance, while relying on contrastive loss alone stalls learning entirely. These results indicate that cross-entropy supplies essential label supervision, contrastive loss sharpens representation learning, and their balanced combination accelerates convergence and yields the best accuracy; consequently, we adopt the $1:1$ setting for all subsequent experiments.

A.4 Error Analysis

Training difficulty on the two 2014 corpora stems from different corpus pathologies. In 14res, the restaurant set, dense annotation leads to structural

Additional AOPE Baselines	14res			14lap			15res			16res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
CMLA$^+$ (Wang et al., 2017)	–	–	48.95	–	–	44.10	–	–	44.60	–	–	50.00
Peng-Two-Stage (Peng et al., 2020)	–	–	56.10	–	–	53.85	–	–	56.23	–	–	60.04
Dual-MRC (Mao et al., 2021)	–	–	74.93	–	–	63.37	–	–	64.97	–	–	75.71
SpanMlt (Zhao et al., 2020)	–	–	75.60	–	–	68.66	–	–	64.68	–	–	71.78

Table 7: Baseline results (%) on the Aspect–Opinion Pair Extraction (AOPE) task. A dash indicates that the corresponding precision or recall was not reported in the source paper.

Additional ASTE Baselines	14res			14lap			15res			16res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Peng-Two-Stage (Peng et al., 2020)	43.24	63.66	51.46	38.87	50.38	42.87	48.07	57.51	52.32	46.96	64.24	54.21
JET-BERT (Xu et al., 2020)	70.56	55.94	62.40	55.39	43.57	51.04	64.45	51.96	57.53	70.42	58.37	63.83
COM-MRC (Zhai et al., 2022)	75.46	68.91	72.01	58.15	60.17	61.17	68.35	61.24	64.53	71.55	71.59	71.57
DGCNAP (Li et al., 2023)	72.90	68.69	70.72	62.02	53.79	57.57	62.23	60.21	61.19	69.75	69.44	69.58
Triple-MRC (Zou et al., 2024)	–	–	72.45	–	–	60.72	–	–	62.86	–	–	68.65

Table 8: Baseline results (%) on the Aspect–Sentiment–Target Extraction (ASTE) task.

Dataset	#Triples	Mean Dist.	Median Dist.	A→O	O→A	Overlap
14res	3 909	3.39	2	2 091	1 796	22
14lap	2 349	3.75	2	1 090	1 257	2
15res	1 747	3.26	2	1 039	707	1
16res	2 247	3.21	2	1 302	944	1

Table 9: Statistics of ASTE triples in the four benchmark datasets. A→O denote aspect appears prior to opinion, and O→A is the other way around; overlap indicate the overlapping between aspect and opinion.

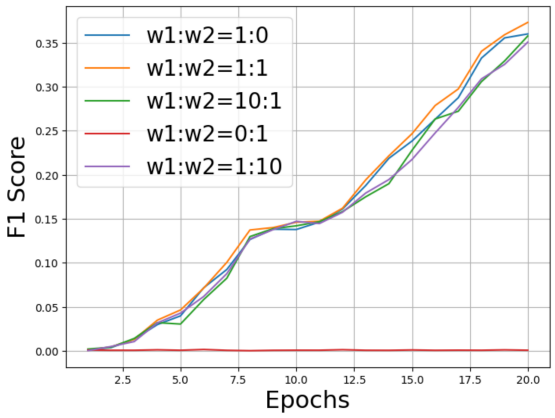


Figure 4: F1 score as a function of training epochs in the combined-train condition for the AOPE task on the 14lap test set, with various loss weight configurations. w1=base loss; w2=contrastive loss.

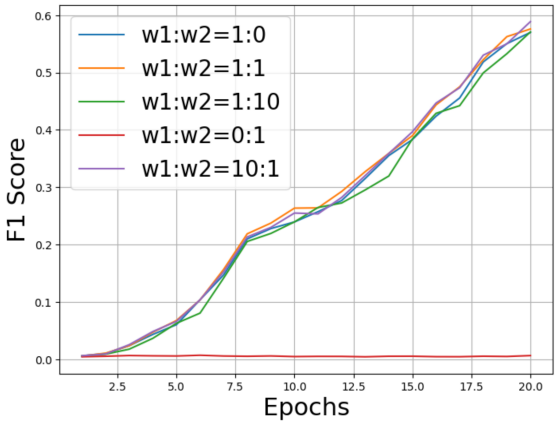
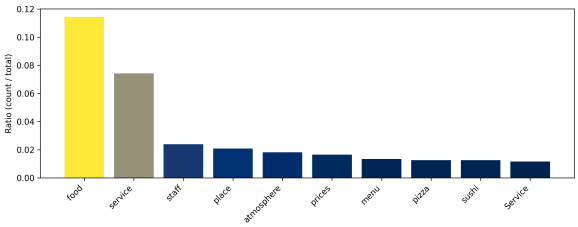


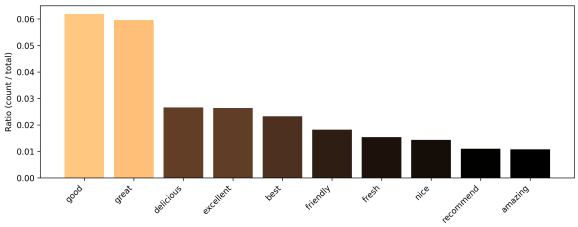
Figure 5: F1 score as a function of training epochs in the combined-train condition for the ASTE task on the 14lap test set, with various loss weight configurations. w1=base loss; w2=contrastive loss.

ambiguity: it contains the most triples and the highest number of bidirectional aspect–opinion links, with 22 explicit overlaps (Table 9). Because most aspect–opinion pairs are separated by only two tokens, the model must assign multiple, often conflicting, roles within very narrow contexts. The extreme lexical skew illustrated in Figure 6 further concentrates gradients on a handful of high-frequency tokens, encouraging overfitting and leaving rare aspects under-represented.

Conversely, 14lap challenges the model through lexical sparsity. Figure 7 shows a much flatter to-

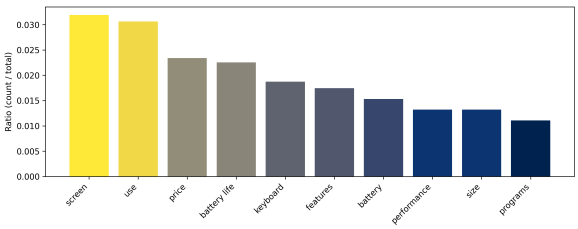


(a) 14res – Aspect token ratios

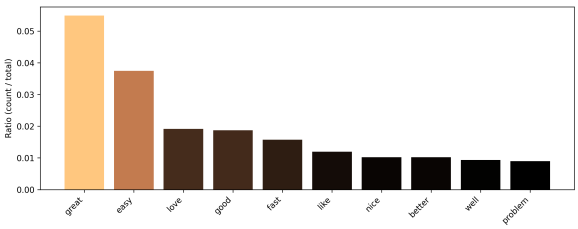


(b) 14res – Opinion token ratios

Figure 6: Token-ratio distributions for the 14res restaurant–review dataset.

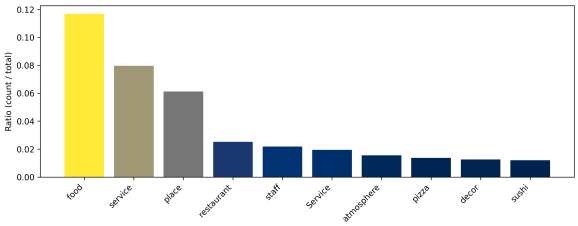


(a) 14lap – Aspect token ratios

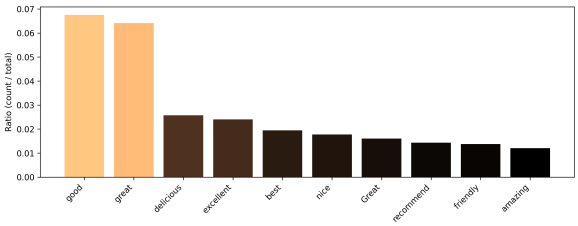


(b) 14lap – Opinion token ratios

Figure 7: Token-ratio distributions for the 14lap laptop-review dataset.

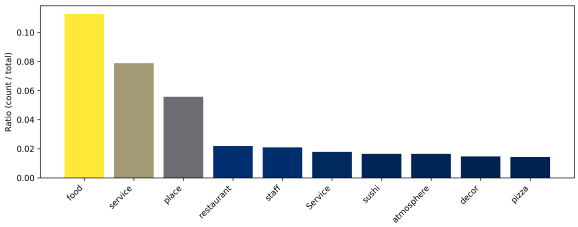


(a) 15res – Aspect token ratios

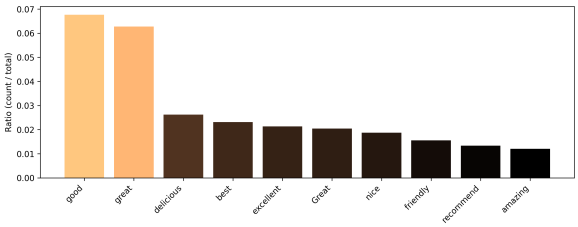


(b) 15res – Opinion token ratios

Figure 8: Token-ratio distributions for the 15res restaurant–review dataset.



(a) 16res – Aspect token ratios



(b) 16res – Opinion token ratios

Figure 9: Token-ratio distributions for the 16res restaurant–review dataset.

ken distribution, indicating a larger type–token ratio and limited repetition for each specialised noun or adjective. Combined with the longest mean aspect–opinion distance in Table 9, this diversity provides too little evidence for reliable embedding updates while simultaneously requiring the encoder to integrate information across wider spans. In short, 14res confounds the model with overlapping, tightly packed signals, whereas 14lap disperses supervision across a broad, domain-specific vocabulary, and both factors are largely absent from the 2015 and 2016 restaurant datasets.

B Ablation Study: Impact of Adding Sentiment Pre-knowledge

We conducted an ablation study to evaluate the contribution of sentiment embedding in our approach. Table 10 compares our model's performance with and without sentiment ground truth against two strong baselines (MiniConGTS and BARTABSA) across four benchmark datasets.

Dataset	Method	F1 (%)
14lap	Ours (w/o Sentiment)	70.67
	Ours (w/ Sentiment)	**81.26**
	MiniConGTS (w/ Sentiment)	62.37
	BARTABSA (w/ Sentiment)	59.59
14res	Ours (w/o Sentiment)	78.85
	Ours (w/ Sentiment)	**85.20**
	MiniConGTS (w/ Sentiment)	73.36
	BARTABSA (w/ Sentiment)	71.68
15res	Ours (w/o Sentiment)	84.73
	Ours (w/ Sentiment)	**92.42**
	MiniConGTS (w/ Sentiment)	89.98
	BARTABSA (w/ Sentiment)	92.13
16res	Ours (w/o Sentiment)	86.06
	Ours (w/ Sentiment)	**86.55**
	MiniConGTS (w/ Sentiment)	72.97
	BARTABSA (w/ Sentiment)	83.95

Table 10: Performance comparison (F1 scores) with and without sentiment ground truth. Bold values indicate the best performance for each dataset.

The results demonstrate that incorporating sentiment ground truth consistently improves performance, with gains ranging from 0.49 to 10.59 percentage points across datasets. The laptop domain (14lap) shows the most substantial improvement (+10.59%), while restaurant datasets show more varied gains (14res: +6.35%, 15res: +7.69%, 16res: +0.49%). Remarkably, even without sentiment ground truth, our model outperforms both baselines on three datasets (14res, 15res, 16res), indicating that our architecture effectively captures sentiment-relevant features from contextual representations alone. With sentiment ground truth, our approach achieves the best performance on three out of four datasets, demonstrating its effectiveness in leveraging explicit sentiment information when available. These findings suggest that while sentiment ground truth provides valuable improvements, our model maintains competitive performance even in its absence, offering flexibility for deployment in scenarios where sentiment annotations may be unavailable or expensive to obtain.