MASTE: A Multi-Agent Pipeline for Zero-Shot Aspect Sentiment Triplet Extraction

Anonymous ACL submission

Abstract

Aspect Sentiment Triplet Extraction (ASTE) requires jointly identifying *(aspect, opinion, sentiment)* triples from a given review sentence. While large language models (LLMs) achieve strong zero-shot performance on many NLP benchmarks, their effectiveness on ASTE remains limited, as single-pass generation forces the model to commit to span boundaries, opinion grouping, and polarity in one decoding step. Common remedies—few-shot in-context learning and chain-of-thought prompting—offer only marginal improvements and rely heavily on either in-domain demonstrations sampled from labeled training data or carefully engineered reasoning prompts, neither of which is broadly available in zero-shot deployment. Inspired by the classical agent paradigm, we propose **MASTE** (**M**ulti-**A**gent pipeline for zero-shot **A**spect **S**entiment **T**riplet **E**xtraction), a four-stage framework in which specialized agents handle each compositional subtask sequentially with explicit conditioning on prior outputs. This design enables entirely training-free zero-shot ASTE and generalizes across different backbones and datasets. Extensive experiments on four ASTE benchmarks show that MASTE substantially outperforms zero-shot and chain-of-thought LLM baselines under the same backbone, narrowing the gap to fully supervised methods without using any labeled triplets. Our code is available at `https://anonymous.4open.science/r/MASTE-4DE5/`.

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) is a fine-grained sentiment analysis task that identifies the sentiment polarity expressed toward each specific aspect mentioned in a given text (Pontiki et al., 2014). Among its subtasks, **Aspect Sentiment Triplet Extraction (ASTE)** (Peng et al., 2020) is the most integrated formulation.

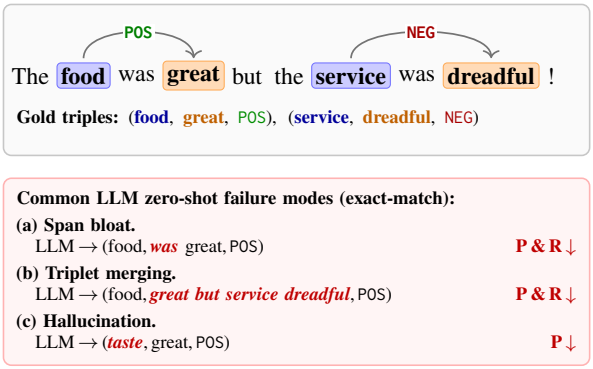

Figure 1: (*Top*) An illustration of the ASTE task: given an input sentence, the model is required to extract the complete set of **`(Aspect, Opinion, Polarity)`** triples. **`Aspects`** (blue) and **`opinions`** (orange) are paired by arcs whose labels denote the sentiment **`polarity`**. (*Bottom*) Three possible systematic failure modes that zero-shot LLM prompting exhibits on the same example; each impairs Precision (**P**), Recall (**R**), or both.

As shown in Figure 1 (*Top*), given a review sentence, ASTE returns the complete set of `(Aspect, Opinion, Polarity)` triples, where the `Aspect` is the term being evaluated, `Opinion` is the expression that describes the aspect, and `Polarity` denotes the corresponding sentiment polarity, including positive, negative, and neutral. For example, given the review sentence *"The food was great but the service was dreadful!"*, an ASTE system is expected to output both (*food*, *great*, `POS`) and (*service*, *dreadful*, `NEG`).

Previous approaches to ASTE and related ABSA subtasks fall into several categories, including pipeline-based methods (Peng et al., 2020), sequence tagging (Xu et al., 2020; Yan et al., 2021), sequence-to-sequence generation (Zhang et al., 2021; Naglik and Lango, 2024), grid and table filling tagging schemes (Wu et al., 2020; Chen et al., 2022; Sun et al., 2024), and multi-domain joint training (Hou et al., 2024). Despite their strong

in-domain performance, these supervised methods share two limitations: (i) they depend on dataset-specific triplet annotations that limit cross-domain transfer, and (ii) they are bound to a fixed annotation convention (e.g., span-boundary style) and require re-training when transferred to new domains or extraction granularities.

Recently, large language models (LLMs) have demonstrated remarkable performance and strong zero/few-shot generalization across a wide range of NLP tasks (Brown et al., 2020; DeepSeek-AI, 2024), including ABSA (Zhang et al., 2024; Scaria et al., 2024; Yang et al., 2024; Fan et al., 2025). However, when applied directly to ASTE, LLMs struggle with this fine-grained structured task: they often fail to capture the complex latent dependencies between aspect and opinion terms, confuse aspect mentions with opinion mentions, and produce systematic span-boundary errors and hallucinated spans. Concretely, even GPT-4o (zero-shot) reaches only 35.4% F1 score on 14RES, leaving a roughly 40-point gap to state-of-the-art supervised methods (Sun et al., 2024). To close this gap, subsequent efforts have explored few-shot in-context learning (Yang et al., 2024), chain-of-thought prompting (Wei et al., 2022), instruction tuning (Scaria et al., 2024), and decomposed pipelines that pair fine-tuned extractors with an LLM-as-a-judge (Bodke et al., 2025); a parallel line of work investigates multi-agent debate for related information extraction tasks (Du et al., 2024; Lu et al., 2025). However, none of these methods explicitly mitigate the systematic single-pass failure modes shown in Figure 1 (*bottom*).

To address the challenges of previous methods, we propose **MASTE**, a training-free **M**ulti-**A**gent pipeline for zero-shot **A**spect **S**entiment **T**riplet **E**xtraction. MASTE decomposes ASTE into four *sequentially-conditioned* agents—Aspect, Opinion, Sentiment, and Consistency—that share one backbone LLM, with each downstream agent receiving the input sentence together with the structured outputs of all upstream agents.

We conduct experiments on four commonly used ASTE benchmarks and compare MASTE against a comprehensive set of previous approaches. Our results show that MASTE achieves state-of-the-art performance among LLM-based methods on all four datasets. We further perform cross-backbone and ablation studies to verify MASTE's generality and assess each component's contribution. Overall, our contributions are summarized as follows:

- **A training-free multi-agent pipeline for ASTE.** To the best of our knowledge, MASTE is the first training-free multi-agent pipeline that requires no task-specific fine-tuning and exposes intermediate structures that can be validated and reused by downstream agents.

- **Span-aware agent design for exact-match ASTE.** Motivated by observed systematic failure modes of zero-shot LLM extraction, we design two mechanisms: minimal-span opinion extraction with one-to-many aspect–opinion mapping and a Consistency Check Agent that performs span grounding, polarity calibration, and duplicate consolidation over the full triplet set.

- **Comprehensive evaluation on ASTE benchmarks.** We conduct extensive experiments on multiple benchmark datasets to validate the effectiveness of MASTE. Results show that the proposed pipeline consistently improves LLM-based ASTE, while ablation and cross-backbone analyses confirm the importance and generality of its stage-wise design.

2 Related Work

2.1 Aspect Sentiment Triplet Extraction

Aspect Sentiment Triplet Extraction (ASTE) was introduced by Peng et al. (2020) to jointly identify aspect terms, opinion expressions, and sentiment polarities. Most prior work relies on supervised learning, including position-aware tagging (Xu et al., 2020), sequence-to-sequence generation (Zhang et al., 2021; Yan et al., 2021), grid-tagging schemes (Wu et al., 2020; Chen et al., 2022; Sun et al., 2024), transformer-based span modeling (Naglik and Lango, 2024), and multi-domain joint training (Hou et al., 2024). Although effective in-domain, these methods require task-specific triplet annotations and typically learn dataset-specific span conventions. MASTE instead studies the training-free setting, where the system must identify exact ASTE triplets without updating model parameters.

Recent work has begun to explore LLMs for ABSA and ASTE. MiniConGTS (Sun et al., 2024) evaluates GPT-3.5-turbo and GPT-4 on ASTE and shows that pretraining–finetuning remains competitive even in the era of LLMs. Other LLM-based ABSA methods use instruction tuning (Scaria et al.,

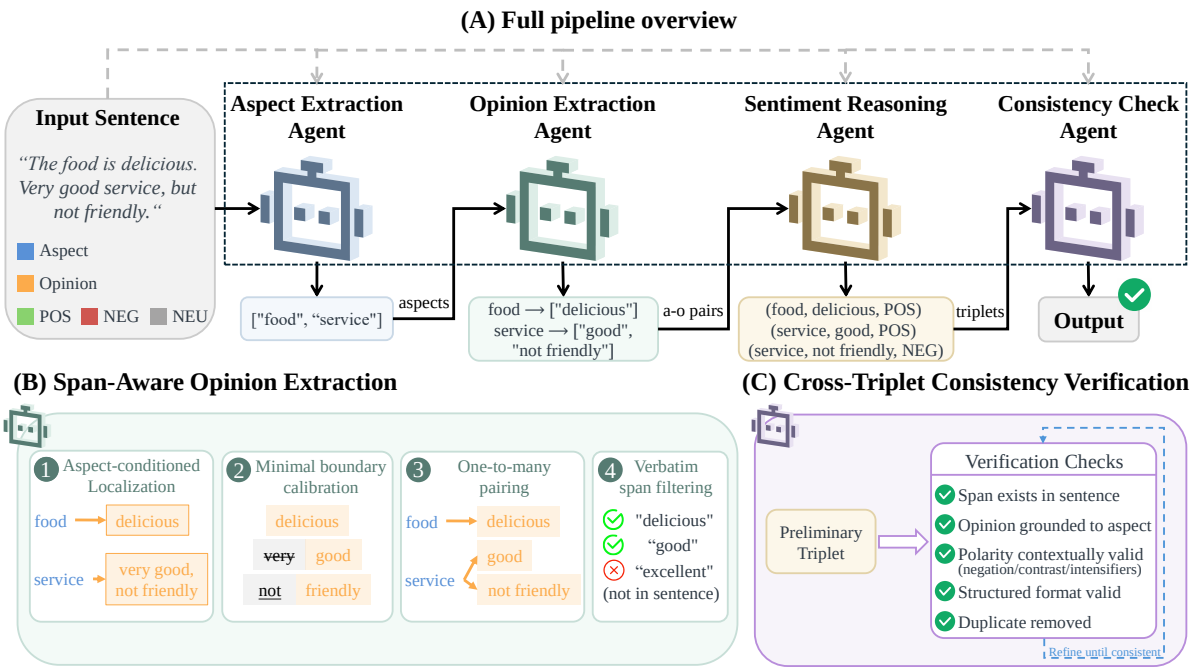

Figure 2: The MASTE pipeline on an example from ASTE-Data-V2. **Solid arrows** carry structured data between stages; **dashed arrows** pass the original sentence as context to each agent. Stage 2 enforces *minimal* opinion spans and supports one-to-many aspect–opinion mapping. Stage 4 validates the full triplet set: hallucination removal, polarity calibration, duplicate consolidation, and output canonicalization.

2024), feature-aware in-context learning (Yang et al., 2024), or syntax–opinion–sentiment reasoning chains (Fan et al., 2025). Closest to our setting, PASTEL (Bodke et al., 2025) decomposes ASTE into subtasks and applies an LLM-as-a-judge, but still depends on finetuned extractors before validation. In contrast, MASTE performs decomposition and span-aware validation entirely with training-free LLM agents.

2.2 Multi-Agent LLMs for Information Extraction

Multi-agent LLM systems were popularized by debate-style methods, where multiple model instances critique and revise one another's outputs (Du et al., 2024). This idea has recently been adapted to structured information extraction: CROSSAGENTIE (Lu et al., 2025) performs cross-type and cross-task debate between Named Entity Recognition (NER) and Relation Extraction (RE) agents; AgentRE (Shi et al., 2024) equips an LLM with memory, retrieval, and reflection modules for low-resource relation extraction; and Agent-Event-Coder (Wang et al., 2025) treats zero-shot event extraction as a multi-agent code-generation and verification process. MASTE differs from these systems by using sequentially conditioned agents over typed ASTE intermediates and by targeting span boundary control for aspect–opinion–sentiment triplets.

2.3 Decomposition and Verification

Decomposition and verification are common strategies for improving LLM reliability. Chain-of-thought prompting (Wei et al., 2022) elicits intermediate reasoning within a single generation, while least-to-most prompting (Zhou et al., 2023) decomposes complex problems into ordered subproblems. Verification methods further check generated solutions using learned verifiers or agentic verification procedures (Cobbe et al., 2021; Anonymous, 2025). MASTE adapts these ideas to fine-grained structured extraction: instead of verifying a free-form final answer, the Consistency Check Agent operates over the whole predicted triplet set, removing hallucinated spans, calibrating sentiment labels, and normalizing opinion boundaries for ASTE's exact-match evaluation.

3 Method

3.1 Task Definition

Given an input sentence $s = (w_1, w_2, \ldots, w_n)$ of n tokens, the goal of ASTE is to extract the complete

set of triplets:

$$\mathcal{T}(s) \;=\; \{(a_i,\, o_i,\, p_i)\}_{i=1}^{m} \qquad (1)$$

where a_i is an aspect term appearing in s, o_i is the corresponding opinion term, and $p_i \in \{\texttt{POS}, \texttt{NEG}, \texttt{NEU}\}$ denotes the sentiment polarity. m is the number of sentiment triplets contained in sentence s. Following standard ASTE evaluation (Xu et al., 2020), a prediction is correct only when the aspect term, opinion term, and sentiment polarity all exactly match a gold triplet.

3.2 Overall Framework

Figure 2 illustrates the overall framework of MASTE. MASTE decomposes ASTE into four sequentially conditioned agents that share the same frozen LLM backbone $\mathcal{M}$ and operate on typed intermediate outputs. Each downstream agent receives the original sentence together with the structured outputs of all upstream agents. The following paragraphs describe each agent in detail.

Aspect Extraction Agent. The Aspect Extraction Agent f_{asp} identifies aspects mentioned in the input sentence. Given s, it returns a set of aspect terms:

$$f_{\text{asp}}(s;\, \mathcal{M}) \;\rightarrow\; \mathcal{A} = \{a_1, \ldots, a_k\} \qquad (2)$$

We constrain the agent to extract only aspect spans that are explicitly mentioned in s, copy them verbatim from the sentence, and avoid paraphrasing, lemmatization, or boundary expansion. Since all downstream stages condition on $\mathcal{A}$, errors at this stage can propagate through the entire pipeline. We therefore apply deterministic span validation after generation: any returned aspect that does not appear as a case-insensitive substring of s is discarded before opinion extraction.

Opinion Extraction Agent. Conditioned on the sentence s and the validated aspect set $\mathcal{A}$, the Opinion Extraction Agent f_{opn} identifies opinion spans that are associated with each extracted aspect:

$$f_{\text{opn}}(s,\, \mathcal{A};\, \mathcal{M}) \;\rightarrow\; \mathcal{P} = \{(a_i, o_i^{(j)})\} \qquad (3)$$

Here $o_i^{(j)}$ denotes the j-th opinion span associated with aspect a_i, allowing the same aspect to appear in multiple pairs when it is described by multiple independent opinions. We further introduce a span-aware extraction procedure consisting of four steps, which is detailed in §3.3.

Sentiment Reasoning Agent. The Sentiment Reasoning Agent f_{sen} takes the sentence s and the aspect–opinion pair set $\mathcal{P}$, and assigns a sentiment polarity to each pair:

$$f_{\text{sen}}(s,\, \mathcal{P};\, \mathcal{M}) \;\rightarrow\; \mathcal{T}^{(3)} = \{(a_i, o_i^{(j)}, p_i^{(j)})\} \quad (4)$$

where $p_i^{(j)} \in \{\texttt{POS}, \texttt{NEG}, \texttt{NEU}\}$ denotes the polarity of opinion $o_i^{(j)}$ toward aspect a_i. By conditioning on the full sentence rather than the opinion span alone, the agent can account for contextual polarity cues such as negation, intensification, and contrast. The resulting set $\mathcal{T}^{(3)}$ serves as the preliminary triplet set for the Consistency Check Agent.

Consistency Check Agent. The Consistency Check Agent f_{con} refines the preliminary triplet set $\mathcal{T}^{(3)}$ in the context of the original sentence s, yielding the final calibrated triplet set:

$$f_{\text{con}}(s,\, \mathcal{T}^{(3)};\, \mathcal{M}) \;\rightarrow\; \mathcal{T}. \qquad (5)$$

Unlike single-pass prompting or per-triplet verification, f_{con} calibrates the output at the triplet-set level, allowing residual errors to be corrected. We detail this procedure in §3.4.

3.3 Span-Aware Opinion Extraction

To address common LLM failure modes in ASTE, especially span bloat and triplet merging (Figure 1, *bottom*), we design f_{opn} as a span-aware extraction stage with the following procedure:

1. **Aspect-conditioned opinion localization.** For each aspect $a_i \in \mathcal{A}$, identify all opinion expressions in s that describe a_i.
2. **Minimal boundary calibration.** Select the shortest contiguous span that preserves the core opinion. For example, degree-only modifiers are removed (*"very clean"*→*"clean"*), whereas polarity-bearing expressions are preserved (*"not bad"*→*"not bad"*).
3. **One-to-many opinion pairing.** For each independent opinion associated with a_i, construct a separate pair $(a_i, o_i^{(j)})$, allowing the same aspect to appear in multiple pairs.
4. **Verbatim span filtering.** Drop pairs whose aspect is not in $\mathcal{A}$, whose opinion span does not appear verbatim in s, or that duplicate an earlier pair.

3.4 Cross-Triplet Consistency Verification

The first three agents yield a preliminary triplet set $\mathcal{T}^{(3)}$, but residual errors such as unsupported spans,

Method	14Res			14Lap			15Res			16Res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Pipeline												
CMLA+ (Wang et al., 2017)	39.18	47.13	42.79	30.09	36.92	33.16	34.56	39.84	37.01	41.34	42.10	41.72
RINANTE+ (Dai and Song, 2019)	31.42	39.38	34.95	21.71	18.66	20.07	29.88	30.06	29.97	25.68	22.30	23.87
Two-stage (Peng et al., 2020)	43.24	63.66	51.46	37.38	50.38	42.87	48.07	57.51	52.32	46.96	64.24	54.21
Li-unified-R+PD (Peng et al., 2020)	40.56	44.28	42.34	41.04	**67.35**	51.00	44.72	51.39	47.82	37.33	54.51	44.31
LLM-based												
GPT-4o zero-shot	32.99	38.13	35.37	17.81	22.55	19.90	27.85	37.73	32.05	32.17	43.00	36.80
GPT-4o few-shot	54.11	66.20	59.55	38.23	48.61	42.80	45.57	60.41	51.95	52.90	71.01	60.63
GPT-4o CoT	41.21	53.32	46.49	26.98	37.71	31.46	33.07	50.93	40.10	39.14	58.17	46.79
GPT-4o CoT+few-shot	46.81	59.86	52.54	29.71	40.85	34.40	35.08	53.81	42.47	41.53	61.09	49.45
Ours												
GPT-4o MASTE	**74.64**	**70.12**	**72.31**	**64.78**	49.24	**55.95**	**68.95**	**66.04**	**67.46**	**74.48**	**72.35**	**73.40**

Table 1: Experimental results on ASTE-Data-V2 (Xu et al., 2020). Precision, Recall, and F1 score are reported for each dataset. The best results are highlighted in bold, and the second-best results are underlined.

polarity mismatches, duplicate predictions, and invalid output formats may remain. We therefore use f_{con} as a triplet-set-level verifier, which performs four consistency checks conditioned on s and $\mathcal{T}^{(3)}$:

1. **Span grounding.** Remove triplets whose aspect or opinion span does not appear verbatim in s.
2. **Polarity calibration.** Revise sentiment labels that conflict with contextual cues such as negation or contrast.
3. **Duplicate consolidation.** Merge duplicate or semantically redundant triplets.
4. **Output canonicalization.** Convert the verified result into a canonical triplet format used for evaluation, with normalized sentiment labels and valid structured output.

4 Experiments

4.1 Datasets and Metrics

We conduct experiments on four public datasets, LAP14, RES14, RES15, and RES16, which are derived from the sentiment evaluation benchmarks SemEval 2014 (Pontiki et al., 2014), SemEval 2015 (Pontiki et al., 2015), and SemEval 2016 (Pontiki et al., 2016), respectively. There are two commonly used versions of these datasets in prior ASTE studies. Following Xu et al. (2020), we conduct all experiments on ASTE-Data-V2, with dataset statistics summarized in Appendix A.1.

We evaluate all models using the widely accepted (`Precision`, `Recall`, `F1`) metrics.

4.2 Baselines

We compare MASTE with two groups of baselines: (1) pipeline-based methods, which decompose ASTE into intermediate subtasks or rely on extracted intermediate structures. These methods provide relevant comparisons for evaluating the stage-wise design of MASTE. (2) LLM-based methods with different prompting strategies. Notably, MASTE uses the same backbone (GPT-4o) as the LLM-based baselines, but does not require any in-context demonstrations.

4.3 Main Results

Table 1 summarizes the results comparing MASTE (GPT-4o) with all pipeline-based and LLM-based baselines on ASTE-Data-V2. The best results are highlighted in bold, and the second-best results are underlined. Overall, MASTE achieves the best F1 score among the compared methods on all four datasets.

Comparison with pipeline-based methods. Compared with pipeline-based baselines, MASTE achieves the highest F1 score on all four datasets, but the margin varies across domains. It improves over the strongest pipeline baseline by 20.85 F1 points on 14RES, 4.95 points on 14LAP, 15.14 points on 15RES, and 19.19 points on 16RES. The gain is therefore substantial on the restaurant datasets, while more moderate on 14LAP, where Li-unified-R+PD obtains much higher recall. This pattern indicates that MASTE is not simply more recall-oriented than prior pipeline systems; rather, its advantage mainly comes from producing more precise exact-match triplets. In particular, MASTE achieves the highest precision on every dataset, suggesting that span grounding, aspect-conditioned opinion extraction, and triplet-level consistency checking help reduce unsupported spans and in-

correct aspect–opinion pairings. On 14LAP, the lower recall also reveals a remaining limitation: MASTE may miss valid triplets in domains where pipeline systems recover broader candidate sets.

Comparison with LLM-based methods. MASTE also consistently outperforms all GPT-4o prompting baselines. Compared with zero-shot prompting, MASTE improves F1 by 36.94, 36.05, 35.41, and 36.60 points on 14RES, 14LAP, 15RES, and 16RES, respectively. Compared with few-shot prompting, MASTE still achieves F1 gains of 12.76, 13.15, 15.51, and 12.77 points on the four datasets, without using in-context demonstrations. This suggests that the gains come from the proposed multi-agent decomposition rather than additional demonstration examples.

Precision-oriented improvements. A consistent pattern in Table 1 is that MASTE achieves the highest precision on all four datasets. This is important for ASTE because exact-match evaluation penalizes hallucinated terms, over-extended opinion spans, and incorrect aspect–opinion pairings. The large precision gains suggest that the Aspect Extraction Agent and Consistency Check Agent effectively filter unsupported spans and reduce false positives. Although MASTE does not always obtain the highest recall, especially on 14LAP, it maintains competitive recall while producing more reliable exact-match triplets.

Effect of decomposition. The results also show that chain-of-thought prompting alone is insufficient for ASTE. Although CoT encourages step-by-step reasoning, it still performs the entire extraction process within a single generation. In contrast, MASTE assigns aspect extraction, opinion extraction, sentiment reasoning, and consistency verification to separate agents with structured intermediate outputs. This design reduces the burden on a single decoding pass and better matches the compositional nature of ASTE.

5 Analysis

5.1 Ablation Study

We conduct ablation experiments to quantify the contribution of each agent in MASTE. Table 2 reports F1 scores on the four datasets using GPT-4o as the backbone. The full model uses all four agents. The ablated variants remove the Consistency Check Agent, the Opinion Extraction Agent, the Sentiment Reasoning Agent, or both the Opinion Extraction and Consistency Check Agents.

Ablation Setting	14Res	14Lap	15Res	16Res	Avg.
MASTE (full)	**72.31**	**55.95**	**67.46**	**73.40**	**67.28**
w/o Consistency	68.04	54.37	59.86	67.59	62.47
w/o Opinion	60.90	51.69	59.72	64.48	59.20
w/o Sentiment	62.84	50.12	57.45	65.94	59.09
w/o Opinion + Consistency	40.35	36.65	41.84	44.18	40.76

Table 2: Ablation study on F1 scores with GPT-4o where w/o denotes removal of the corresponding agent from MASTE. Full results are provided in Appendix A.3.

Backbone	Method	14Res	14Lap	15Res	16Res	Avg.
GPT-3.5-turbo	Zero-shot	46.18	31.88	39.21	44.34	40.40
	MASTE	57.41	36.45	50.93	56.19	50.25
	Δ	11.23	4.57	11.72	11.85	
Claude Sonnet 4.6	Zero-shot	61.06	38.59	54.09	63.01	54.19
	MASTE	66.41	48.23	56.70	65.83	59.29
	Δ	5.35	9.64	2.61	2.82	
Gemini-3-flash	Zero-shot	54.36	35.85	46.89	56.97	48.52
	MASTE	67.48	51.80	62.98	67.74	62.50
	Δ	13.12	15.95	16.09	10.77	
Deepseek-V3.2	Zero-shot	47.55	31.22	41.86	50.93	42.89
	MASTE	66.38	49.32	57.25	64.39	59.33
	Δ	18.83	18.10	15.39	13.46	
Seed 2.0 pro	Zero-shot	39.02	27.31	32.84	36.96	34.03
	MASTE	67.60	49.16	62.71	65.32	61.20
	Δ	28.58	21.85	29.87	28.36	

Table 3: Cross-backbone analysis on F1 scores. Δ denotes the absolute F1 gain of MASTE over Zero-shot under the same backbone. Full results are provided in Appendix A.4.

The full model achieves the best F1 on all datasets, with an average F1 of 67.28. Removing the Consistency Check Agent reduces the average F1 from 67.28 to 62.47. The drop is observed on all datasets, indicating that final triplet-level verification is consistently useful. This stage helps remove unsupported spans, duplicated triplets, and invalid aspect–opinion–sentiment combinations before evaluation.

Removing the Opinion Extraction Agent causes a larger average drop. This result confirms the importance of aspect-conditioned opinion localization. Without this stage, the model is more likely to miss aspect-specific opinion expressions or generate opinion spans with inaccurate boundaries. Removing the Sentiment Reasoning Agent also leads to a clear degradation, suggesting that explicit sentiment reasoning remains significant even when the aspect and opinion spans are already identified.

The largest degradation appears when the Opinion Extraction and Consistency Check Agents are removed together. In this setting, the average F1

Sentence	Gold Triplet(s)	MASTE Prediction	Error Type
It is fast and easy to use.	(use, easy, POS)	∅	Missing prediction
The baterry is very longer.	(baterry, longer, POS)	(battery, longer, POS)	Span normalization
I am pleased with the fast log on, speedy WiFi connection and the long battery life.	(log on, pleased, POS); (WiFi connection, pleased, POS); (battery life, pleased, POS)	(log on, fast, POS); (WiFi connection, speedy, POS); (battery life, long, POS); ∅ for shared opinion links	Shared-opinion omission
No green beans, no egg, no anchovy dressing, no nicoise olives, no red onion.	(green beans, No, NEU); (egg, no, NEU); ...	(green beans, no, NEG); (egg, no, NEG); ...	Polarity convention
The food is all-around good, with the rolls usually excellent and the sushi/sashimi not quite on the same level.	(food, good, POS); (rolls, excellent, POS)	(food, good, POS); (rolls, excellent, POS); (sushi/sashimi, not quite on the same level, NEG)	Annotation-sensitive FP

Table 4: Representative errors made by the full MASTE pipeline. The analysis focuses on residual failure modes. Green marks gold-only or gold-correct elements; red marks missing, mismatched, or extra MASTE predictions.

drops sharply from 67.28 to 40.76. This result shows that local aspect-conditioned opinion extraction and global consistency verification are complementary. The former constructs higher-quality candidate pairs, while the latter filters remaining invalid triplets. Overall, the ablation results confirm that the four agents contribute jointly to ASTE.

5.2 Cross-Backbone Analysis

We further evaluate whether MASTE generalizes across different LLM backbones. Table 3 compares direct zero-shot prompting with MASTE (which also requires no in-context demonstrations) under multiple backbone models.

Across all backbone settings, MASTE consistently outperforms direct zero-shot prompting on all four datasets. The gains are observed across representative models from different model families, indicating that the effectiveness of MASTE does not depend on a single proprietary model and can generalize across diverse backbones. These results further support our central hypothesis that decomposing ASTE into specialized subtasks provides a more reliable inference process than single-pass triplet generation.

Overall, the cross-backbone results support the generality of MASTE as a training-free and backbone-agnostic framework for zero-shot ASTE.

5.3 Error Analysis

The ablation study quantifies the contribution of each agent to the final performance. We further inspect the residual errors made by the full MASTE pipeline. Table 4 presents representative failure cases. Following the order of the examples, these errors mainly arise from (i) missed event-like aspect spans, (ii) surface-form mismatches under exact-match evaluation, (iii) incomplete recovery of shared opinion links, (iv) polarity disagreement with dataset-specific annotation conventions, and (v) annotation-sensitive extractions that are textually plausible but absent from the gold labels.

The first two cases are span-level failures. In the first case, the gold aspect *use* is predicate-like rather than a concrete noun phrase. Once the Aspect Extraction Agent misses this span, all downstream agents receive no valid target and MASTE returns an empty set. In the second case, MASTE normalizes the misspelled gold aspect *baterry* into *battery*. The prediction is semantically close, but exact-match evaluation requires the surface span to be copied exactly, so the triplet is counted as incorrect.

The third case shows a relation-coverage error. MASTE extracts local attribute opinions such as *fast*, *speedy*, and *long*, but misses the shared opinion *pleased*, which applies to multiple aspects. This suggests that one-to-many aspect–opinion recovery remains difficult when a sentence contains both local modifiers and a global evaluative predicate.

The last two cases are annotation-sensitive. For absence expressions such as *no*, MASTE tends to infer negative sentiment, while ASTE-Data-V2 may annotate the corresponding triplets as neutral. Conversely, MASTE may extract a textually supported negative comparison absent from the gold set. These cases suggest that future work should focus not only on aspect recall, but also on annotation-aware polarity calibration and uncertainty handling for convention-dependent triplets.

6 Conclusion

In this paper, we presented MASTE, a training-free multi-agent framework for zero-shot Aspect Sentiment Triplet Extraction. Rather than requiring a single LLM generation to determine aspect spans, opinion spans, sentiment labels, and their pairings at once, MASTE decomposes ASTE into four sequentially conditioned agents for aspect extraction, aspect-conditioned opinion extraction, sentiment reasoning, and triplet-level consistency verification. Experiments on four benchmark datasets from ASTE-Data-V2 show that MASTE consistently improves over existing pipeline-based and LLM-based methods. Ablation and cross-backbone analyses further confirm the contribution of each stage and the generality of the framework across diverse LLM families. These findings suggest that explicit intermediate structures and verification are effective tools for adapting LLMs to fine-grained structured sentiment extraction without task-specific training.

7 Limitations

MASTE's performance is inherently bounded by the reasoning and span-extraction capabilities of the underlying backbone LLM. In particular, errors made in the Aspect Extraction Agent can propagate to all downstream stages, since opinion extraction and sentiment reasoning both condition on the predicted aspect set. Therefore, the aspect recall remains a major bottleneck.

The multi-agent design introduces additional inference cost relative to single-pass generation. MASTE requires four LLM calls per sentence, which increases both latency and monetary cost in large-scale or real-time applications. Future work may explore model-level routing mechanisms that assign simpler inputs or subtasks to smaller models while reserving stronger LLMs for ambiguous and complex cases. Such a method could improve efficiency while preserving the benefits of staged decomposition.

Finally, our experiments are conducted on English review-domain ASTE benchmarks. Although the cross-backbone results support the robustness of the framework across LLM families, further validation is needed for multilingual settings, longer documents, noisier user-generated content, and domains whose annotation conventions differ from ASTE-Data-V2. We encourage future work to evaluate MASTE in these broader conditions.

8 Ethical considerations

This work is designed for scientific research on aspect-level sentiment analysis. Our experiments are conducted on public benchmark datasets and do not involve new user data collection or human subject experiments. We believe MASTE can contribute positively to the ABSA research community by reducing the dependence on task-specific training data, improving the transparency of intermediate extraction decisions, and providing reusable prompts and code for future research.

At the same time, MASTE inherits potential risks from the underlying LLMs. Its outputs may reflect model bias or unsupported sentiment inferences, and therefore should not be treated as ground-truth judgments in high-stakes decision making. When applying the framework to proprietary, private, or sensitive text, practitioners should ensure that data handling complies with the privacy policies and legal requirements of their deployment context. We will release code and prompts to facilitate reproducibility and to make each extraction stage easier to audit.

References

Anonymous. 2025. VerifiAgent: A unified verification agent with meta- and tool-based verification. In *Findings of EMNLP*.

Mandar Bodke and 1 others. 2025. PASTEL: Polarity-aware sentiment triplet extraction with LLM-as-a-judge. In *Findings of ACL*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Hao Chen, Zheyu Zhai, Siyun Feng, Dandan Li, and Bing Wang. 2022. Enhanced multi-channel graph convolutional network for aspect sentiment triplet extraction. In *Proceedings of ACL*, pages 2974–2985.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. In *arXiv preprint arXiv:2110.14168*.

Hongliang Dai and Yangqiu Song. 2019. 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*, pages 5268–5277.

DeepSeek-AI. 2024. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2024. Improving factuality and reasoning in language models through multiagent debate. In *Proceedings of ICML*.

Rui Fan, Shu Li, Tingting He, and Yu Liu. 2025. Aspect-based sentiment analysis with syntax-opinion-sentiment reasoning chain. In *Proceedings of the 31st international conference on computational linguistics*, pages 3123–3137.

Yufei Hou and 1 others. 2024. Train once for all: A transitional approach for efficient aspect sentiment triplet extraction. *arXiv preprint arXiv:2412.00208*.

Meng Lu, Yuzhang Xie, Zhenyu Bi, Shuxiang Cao, and Xuan Wang. 2025. CROSSAGENTIE: Cross-type and cross-task multi-agent LLM collaboration for zero-shot information extraction. In *Findings of ACL*.

Iwo Naglik and Mateusz Lango. 2024. ASTE-Transformer: Modelling dependencies in aspect-sentiment triplet extraction. In *Findings of EMNLP*.

Haiyun Peng, Li Xu, Lidong Bing, Fuli Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 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*, pages 19–30.

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*, pages 486–495.

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*, pages 27–35.

Kevin Scaria, Himanshu Gupta, Siddharth Goyal, Saurabh Arjun Sawant, Swaroop Mishra, and Chitta Baral. 2024. InstructABSA: Instruction learning for aspect based sentiment analysis. In *Proceedings of NAACL*.

Yuchen Shi, Guochao Jiang, Tian Qiu, and Deqing Yang. 2024. AgentRE: An agent-based framework for navigating complex information landscapes in relation extraction. In *Proceedings of CIKM*.

Qiao Sun, Zhao Wei, Yunfei Long, Qin Lu, and Ruifeng Chen. 2024. MiniConGTS: A near ultimate minimalist contrastive grid tagging scheme for aspect sentiment triplet extraction. In *Proceedings of EMNLP*.

Anonymous Wang and 1 others. 2025. Extracting events like code: A multi-agent programming framework for zero-shot event extraction. In *arXiv preprint arXiv:2511.13118*.

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.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of NeurIPS*.

Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for aspect-oriented fine-grained opinion extraction. In *Findings of EMNLP*, pages 2576–2585.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. In *Proceedings of EMNLP*, pages 2339–2349.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of ACL*, pages 2416–2429.

Songhua Yang, Xinke Jiang, Hanjie Zhao, Wenxuan Zeng, Hongde Liu, and Yuxiang Jia. 2024. Faima: Feature-aware in-context learning for multi-domain aspect-based sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7089–7100.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Pan, and Lidong Bing. 2024. Sentiment analysis in the era of large language models: A reality check. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3881–3906.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021. Towards generative aspect-based sentiment analysis. In *Proceedings of ACL*, pages 504–510.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *Proceedings of ICLR*.

A Appendix

A.1 Implementation Details

For closed-source LLM backbones, we use the official provider APIs, including OpenAI for GPT models, Anthropic for Claude, Google for Gemini, etc. Unless otherwise specified, all LLM calls use temperature 0 and all intermediate outputs are parsed as JSON objects before being passed to downstream agents or the evaluator.

We conduct experiments on ASTE-Data-V2 (Xu et al., 2020), which contains four standard ASTE domains: 14LAP (Lap14), 14RES (Res14), 15RES, and 16RES. These domains are derived from SemEval 2014 (Pontiki et al., 2014), SemEval 2015 (Pontiki et al., 2015), and SemEval 2016 (Pontiki et al., 2016). Table 5 summarizes the datasets.

All systems are evaluated with micro-averaged exact-match precision, recall, and F1 over triplet sets. Before matching, aspect and opinion strings are case-normalized and whitespace-normalized, and sentiment labels are mapped to a canonical uppercase form. A predicted triplet is counted as correct only when its normalized aspect span, normalized opinion span, and sentiment label all match a gold triplet in the same sentence. Duplicate predictions collapse under set matching, so repeated identical triplets do not increase the number of correct predictions.

Dataset	Split	#S	#T	#A	#O	#+	#0	#-	#SW	#MW
14RES	Train	1266	2338	986	844	1692	166	480	1586	752
	Dev	310	577	396	307	404	54	119	388	189
	Test	492	994	579	437	773	66	155	657	337
14LAP	Train	906	1460	733	695	817	126	517	824	636
	Dev	219	346	268	237	169	36	141	190	156
	Test	328	543	400	329	364	63	116	291	252
15RES	Train	605	1013	582	462	783	25	205	678	335
	Dev	148	249	191	183	185	11	53	165	84
	Test	322	485	347	310	317	25	143	297	188
16RES	Train	857	1394	759	623	1015	50	329	918	476
	Dev	210	339	251	221	252	11	76	216	123
	Test	326	514	338	282	407	29	78	344	170

Table 5: Dataset statistics of ASTE-Data-V2. #S, #T, #A, and #O denote the numbers of sentences, triplets, aspects, and opinions, respectively. #+, #0, and #- denote positive, neutral, and negative triplets. #SW and #MW denote single-word and multi-word triplets, respectively.

A.2 Prompt Templates

We report the prompt templates used by both MASTE and the LLM-based baselines. The MASTE prompts follow a staged agent design: each agent receives only the information required for its subtask, returns a structured JSON object, and passes validated intermediate outputs to downstream agents. For the LLM-based baselines, we follow the zero-shot, few-shot, CoT, and CoT+few-shot settings used in (Sun et al., 2024). To ensure that all methods can be evaluated under the same exact-match protocol, we require the baseline prompts to produce the same JSON triplet format as MASTE.

A.2.1 MASTE Prompts

The variables {sentence}, {aspects}, {pairs}, and {triplets} are filled at inference time.

Aspect Extraction Agent

```
System:
You are an expert in aspect-based sentiment
 analysis. Your task is to identify ASPECT
TERMS in a sentence.

Rules:
- Extract only aspects that are explicitly
present in the sentence, do not infer or
paraphrase.
- Copy the aspect span exactly as it
appears in the sentence, including any
misspellings.
- Use the minimal span that identifies the
evaluated target.
- An aspect can be a single word or a multi-
word phrase.
- Return a JSON list of strings only. If no
 aspects are found, return [].

User:
Sentence: {sentence}
Identify all aspect terms. Return a JSON
list of strings. If no aspects are found,
return [].
```

Opinion Extraction Agent

```
System:
You are an expert in aspect-based sentiment
 analysis. Your task is to identify OPINION
 EXPRESSIONS for given aspect terms in a
sentence.

An opinion expression is the SHORTEST word
or phrase copied VERBATIM from the sentence
that captures the reviewer's evaluation of
a specific aspect. Copy the span exactly as
 it appears in the text, even if it
contains misspellings or typos.

Rules:
- Extract the shortest opinion phrase
copied verbatim from the sentence.
```

```
- Do not include degree adverbs unless they
 change meaning.
- If one aspect has multiple independent
opinions, output one entry
  per opinion rather than merging them.
- Omit aspects for which no opinion is
found.
- Return a JSON list: [{"aspect": ..., "
opinion": ...}].

User:
Sentence: {sentence}
Aspects: {aspects}
For each aspect find its opinion expression
(s). Use the minimal span - no degree
adverbs unless they change sentiment
polarity. If one aspect has multiple
opinions, output one entry per opinion.
Omit aspects with no opinion.
Return JSON list:
[{"aspect": ..., "opinion": ...}].
```

Sentiment Reasoning Agent

```
System:
You are an expert sentiment analyst. Your
task is to determine the SENTIMENT POLARITY
 for aspect-opinion pairs.

Sentiment labels:
- POS: positive sentiment.
- NEG: negative sentiment.
- NEU: neutral sentiment.

Rules:
- Consider the context of the full sentence,
 not just the opinion word in isolation.
- Be careful with negation.
- Consider intensifiers and modifiers.

User:
Sentence: {sentence}
Aspect-Opinion pairs: {pairs}
Classify each as POS, NEG, or NEU.
Return JSON list:
[{"aspect": ..., "opinion": ..., "sentiment
": ...}].
```

Consistency Check Agent

```
System:
You are a quality-control expert for aspect-
based sentiment analysis. Review a set of
extracted (aspect, opinion, sentiment)
triplets.

Checks:
1. HALLUCINATION: remove triplets whose
aspect or opinion does not appear verbatim
in the sentence.
2. SPAN TRIM: trim unnecessary degree
adverbs unless they change polarity.
3. SENTIMENT ERROR: fix polarity labels
that conflict with context.
4. MISSING: add clearly expressed aspect-
opinion pairs that are absent.
5. DUPLICATION: collapse duplicate or
redundant triplets.

Rules:
- Default to keeping triplets.
- Do not add speculative triplets
unsupported by the sentence.

User:
Sentence: {sentence}
Proposed triplets: {triplets}
Review for hallucinations, sentiment errors,
 duplicates, and missing pairs. Return only
 valid, corrected triplets as a JSON list:
[{"aspect": ..., "opinion": ..., "sentiment
": ...}].
```

A.2.2 Baseline Prompts

See Table 6.

A.3 Full Ablation Results

See Table 7.

A.4 Full Cross-Backbone Results

See Table 8.

Baseline	Prompt Template
Zero-shot	**Instruction:** Suppose you are an expert in aspect-based sentiment analysis. Given the input sentence, extract all `(aspect, opinion, sentiment)` triplets. The sentiment label must be `POS`, `NEG`, or `NEU`. Aspect and opinion strings must be copied verbatim from the sentence. Return only a JSON list in the format `[{"aspect": ..., "opinion": ..., "sentiment": ...}]`. **Input:** `{sentence}`.
Few-shot	**Instruction:** Use the same direct extraction instruction as the zero-shot baseline. **Demonstrations:** `{examples}`, where each example consists of an input sentence and a JSON list of gold triplets. **Input:** `{sentence}`. **Output:** return only a JSON list with the same triplet schema.
CoT	**Instruction:** Suppose you are an expert in aspect-based sentiment analysis. Analyze the sentence step by step: (1) identify all explicitly mentioned aspects; (2) identify the opinion expression linked to each aspect; (3) assign `POS`, `NEG`, or `NEU`; (4) output one triplet for each aspect–opinion pair, including one-to-many cases; and (5) keep all aspect and opinion spans verbatim. After brief reasoning, return the final answer as a JSON list in the format `[{"aspect": ..., "opinion": ..., "sentiment": ...}]`. **Input:** `{sentence}`.
CoT+few-shot	**Instruction:** Use the same step-by-step CoT instruction as the CoT baseline. **Demonstrations:** `{cot_examples}`, where each example contains an input sentence, intermediate reasoning over aspects, opinions, and sentiment labels, and a final JSON triplet list. **Input:** `{sentence}`. **Output:** after brief reasoning, return the final JSON list with the same triplet schema.

Table 6: Prompt templates for LLM-based baselines. The structure follows the zero-shot, few-shot, CoT, and CoT+few-shot settings used in prior ASTE prompting protocols, while the final output is normalized to the same JSON triplet schema used by MASTE.

Ablation Setting	14Res			14Lap			15Res			16Res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
MASTE (full)	**74.64**	**70.12**	**72.31**	**64.78**	**49.24**	**55.95**	**68.95**	**66.04**	**67.46**	**74.48**	**72.35**	**73.40**
w/o Consistency	71.17	65.17	68.04	61.93	48.45	54.37	64.89	55.55	59.86	70.16	65.20	67.59
w/o Opinion	65.29	57.06	60.90	59.04	45.97	51.69	65.04	55.20	59.72	67.58	61.65	64.48
w/o Sentiment	66.81	59.32	62.84	57.86	44.21	50.12	61.56	53.85	57.45	68.84	63.27	65.94
w/o Opinion + Consistency	41.36	39.39	40.35	35.60	37.76	36.65	48.64	36.71	41.84	43.67	44.70	44.18

Table 7: Full ablation results (precision, recall, F1) with GPT-4o on four ASTE-Data-V2 benchmarks. Rows correspond to the ablation settings in Table 2.

Backbone	Method	14Res			14Lap			15Res			16Res		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
GPT-3.5-turbo	Zero-shot	39.21	56.17	46.18	26.21	40.69	31.88	31.21	52.75	39.21	35.28	59.64	44.34
	MASTE	62.03	53.42	57.41	39.52	33.83	36.45	54.19	48.04	50.93	55.04	57.39	56.19
Claude Sonnet 4.6	Zero-shot	57.82	64.69	61.06	36.87	40.48	38.59	49.24	60.00	54.09	57.79	69.26	63.01
	MASTE	63.00	70.22	66.41	48.59	47.87	48.23	52.95	61.03	56.70	61.34	71.01	65.83
Gemini-3-flash	Zero-shot	50.43	58.95	54.36	33.18	39.00	35.85	41.67	53.61	46.89	50.84	64.79	56.97
	MASTE	61.23	**75.15**	67.48	48.32	**55.82**	**51.80**	54.81	**74.02**	**62.98**	59.97	**77.82**	**67.74**
Deepseek-V3.2	Zero-shot	45.85	49.40	47.55	29.41	33.27	31.22	38.14	46.39	41.86	46.96	55.64	50.93
	MASTE	63.26	69.82	66.38	48.07	50.65	49.32	53.60	61.44	57.25	59.73	69.84	64.39
Seed 2.0 pro	Zero-shot	38.90	39.13	39.02	26.73	27.91	27.31	36.43	29.90	32.84	41.87	33.07	36.96
	MASTE	**68.20**	67.00	**67.60**	**49.00**	49.33	49.16	**62.50**	63.00	62.71	**64.80**	66.00	65.32

Table 8: Full cross-backbone results (precision, recall, F1) across five backbone models on ASTE-Data-V2. Rows correspond to the settings in Table 3. The best results are highlighted in bold, and the second-best results are underlined.