DRIFT: Dependency Reasoning with Retrieval-Augmented Instruction Fine-Tuning for Aspect–Sentiment Quadruple Prediction

Anonymous ACL submission

Abstract

Aspect Sentiment Quadruple Prediction (ASQP) extracting [aspect, category, opinion, sentiment] tuples remains challenging due to the limited use of syntactic structure and the weakness of exact-match evaluation under span-boundary variation. We present DRIFT, a novel unified framework that combines syntax-aware instruction tuning with retrieval-augmented inference. DRIFT injects linearized dependency trees into the model input to provide explicit syntactic cues during reasoning, and retrieves semantically similar demonstrations to construct informative contexts at inference time. To mitigate evaluation instability caused by minor boundary disagreements, we further introduce an LLM-based adjudicator that compare predicted and reference spans, yielding judgments that better align with human annotation. Experiments on Rest15 and Rest16 show that DRIFT achieves state-of-the-art F1 score. Moreover, our ablation analysis indicates that syntactic guidance contributes most under fine-tuning, whereas retrieval augmentation is especially beneficial in in-context learning (ICL). The adjudicator improves the robustness of evaluation by reducing sensitivity to boundary-level mismatches[1].

1 Introduction

Aspect Sentiment Quadruple Prediction extracts four elements from reviews (Zhang et al., 2021a): aspect term, aspect category, sentiment polarity, and opinion term. For example, "Can't wait to go back" yields [Null, restaurant general, positive, go back]. Unlike pipeline methods that handle each element separately (Wan et al., 2020), ASQP requires modeling all components together (Mao et al., 2021). Current models struggle with three challenges: finding implicit aspects (Cai et al., 2021), matching distant aspects and opinions (Zhang et al., 2024), and understanding complex sentence structures (Liang et al., 2022). Prior approaches often rely on surface level text proximity, leading to erroneous attachments when the intended sentiment target is syntactically connected but explicity distant (Figure 1).

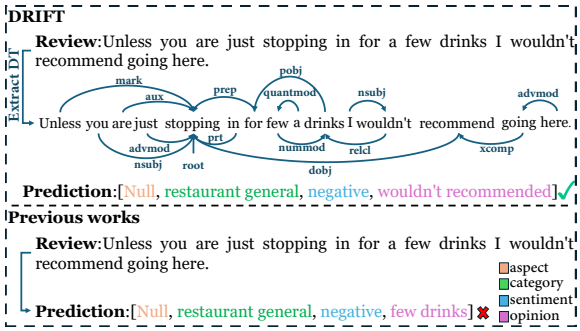


Figure 1: Our method uses the dependency structure to correctly link opinions with their true, even implicit, aspects. Earlier approaches rely only on surface text and often attach sentiments to incorrect words. As a result, our predictions are more accurate and match the ground truth.

Recent advances in Large Language Models (LLMs) have demonstrated strong capabilities in knowledge integration, structured reasoning, and entity extraction (Wei et al., 2022; Brown et al., 2020). These capabilities enable effective instruction tuning for ASQP tasks (Scaria et al., 2024), where models learn task-specific patterns through carefully designed prompts. Building on this, retrieval-augmented generation (RAG) approaches enhance in-context learning by selecting relevant few-shot examples (Zheng et al., 2024). More recently, SimRP (Jian et al., 2025) incorporates syntactic similarity to improve example retrieval, selecting demonstrations based on structural patterns rather than semantic similarity. However, despite using syntax for retrieval, no existing method embeds these syntactic relations directly into the prompt to explicitly guide the LLM's reasoning.

[1] https://github.com/repanonymous/drift

This gap motivates our approach to integrate dependency structures directly into prompts, enabling LLMs to leverage syntactic knowledge explicitly during quadruple extraction.

Recent work has shown that LLMs are not only effective extractors but also reliable evaluators capable of evaluating semantic equivalence between model predictions and reference annotations (Zheng et al., 2023; Liu et al., 2023). In contrast, standard ASQP evaluation methods rely on exact matching, which often underrates model performance because of small differences in span boundaries (Zhang et al., 2021a; Hua et al., 2025). For example, predictions such as "delicious" and "delicious food " are semantically identical yet fail under exact matching. These differences usually come from natural linguistic variation rather than true semantic mistakes, which leads to unfairly lower F1 scores that underestimate the model's real capability (Yang et al., 2025; Zhang et al., 2023). This limitation shows the need for LLM-based semantic evaluation methods that can judge the quality of predicted quadruples more accurately than surface-level token matching.

In this work, we address both the structural reasoning limitations of LLMs and the evaluation challenges in ASQP. First, unlike prior work that uses syntactic information only for retrieval, as shown in Figure 1, we directly incorporate dependency structures into prompts as explicit input to guide the model's reasoning process. By providing grammatical relationships between words, the model can better identify aspect-opinion alignments, especially in complex sentences with implicit aspects, long-distance dependencies or multiple quadruples. Second, to address span boundary variations between model predictions and gold annotations, we introduce an LLM-based judgment module that evaluates semantic equivalence rather than relying solely on exact string matching. This adjudicator selects semantically correct quadruples even when span boundaries differ from reference annotations. By combining syntax-aware prompting with LLM-based evaluation, our framework DRIFT enhances both extraction quality and evaluation accuracy, achieving substantial performance gains across ASQP benchmarks.

Our contributions are as follows:

- We propose a syntax-augmented fine-tuning framework that enriches each training instance with its dependency tree, and a retrieval-augmented inference strategy that injects semantically relevant few-shot examples into prompts to enhance syntactically grounded reasoning.
- We introduce an LLM-based adjudication mechanism that guided by the task definition, query, and corresponding dependency tree, resolves prediction-reference mismatches and span disagreements to produce more consistent quadruples.
- Extensive experiments on the SemEval Rest15 and Rest16 benchmarks demonstrate that our approach consistently surpasses both traditional ABSA systems and strong LLM baselines.

2 Related Work

Graph-based ABSA encodes sentences as dependency or bipartite graphs to propagate sentiment from opinion tokens to their target aspects, yielding richer aspect–opinion representations (e.g., dependency GCNs) (Zhang et al., 2019); attention-augmented variants further weight salient syntactic neighbors (Huang and Carley, 2019). Despite gains, graph construction and edge labeling are parse-sensitive, long-range or implicit sentiments remain difficult and interpretability is limited.

Large pretrained language models have substantially improved ABSA because contextual encoders fine-tuned for downstream tasks (e.g., BERT) typically outperform earlier methods (Devlin et al., 2019; Sun et al., 2019). Instruction tuning recasts ABSA subtasks as natural-language directives, enabling strong few-shot and zero-shot performance (Scialom et al., 2022), while parameter-efficient adaptation (e.g., LoRA) is practical at scale and retrieval augmentation supplies contextually relevant examples in low-data systems (Guu et al., 2020). Recent work also targets reasoning through structured prompts (syntax→opinion→sentiment) and chain-of-thought to align inference with the compositional requirements of quadruple extraction (Fei et al., 2023; Wei et al., 2022).

GAS (Zhang et al., 2021b) formulates aspect sentiment quad prediction as a generative paraphrase task, enabling flexible modeling of aspect-opinion-sentiment relations. ILO (Hu et al., 2022b) and GenDA (Wang et al., 2023) enhance generative models through template-order and generation-based data augmentation, while MvP (Gou et al.,

2023) adopts multi-view prompting to capture complementary semantic perspectives. IVLS (Nie et al., 2024) introduces implicit variables and latent structures to model hidden dependencies among quadruple elements, improving the handling of implicit aspects and opinions. MUL (Hu et al., 2023) and OTCL (Li et al., 2024) further enhance model robustness through uncertainty-aware learning and opinion-tree-guided contrastive objectives.

3 Method

We propose an ASQP framework that integrates four key components: (1) syntax-aware prompting, where we construct $Prompt(x)$ containing $[TaskDefinition;\ x;\ D(x)]$, where $D(x)$ is the dependency parse of sentence x; (2) instruction tuning with Parameter Efficient Fine-Tuning (PEFT) for efficient task adaptation; (3) retrieval-augmented inference in which semantically similar examples $S(x)$ are incorporated to construct $Prompt_{\text{ICL}}(x)$ containing $[TaskDefinition;\ S(x);\ x;\ D(x)]$, enabling one-pass decoding of aspect–sentiment quadruples; and (4) an LLM-based judge J that resolves prediction–reference mismatches and span disagreements. The overall workflow is shown in Figure 2.

3.1 Problem Formulation

Given an input sentence $x = (w_1, \ldots, w_n)$, the goal of ASQP is to extract a set of aspect–sentiment quadruples:

$$Y(x) = \{\, q_i \,\}_{i=1}^{m} \qquad q_i = (a_i, c_i, s_i, o_i) \quad (1)$$

where a_i denotes the *aspect term*, o_i the *opinion term*, $c_i \in \mathcal{C}$ the *aspect category* selected from a predefined category set $\mathcal{C}$, and $s_i \in \{Positive, Negative, Neutral\}$ the *sentiment polarity*. The variable m represents the number of aspect–opinion quadruples identified in the sentence x.

3.2 Syntax-Aware Prompting

Given a sentence x, we compute its dependency parse $D(x)$ and construct a syntax-aware prompt:

$$Prompt(x) = \big[\, task;\ x;\ D(x) \,\big] \quad (2)$$

By using spaCy[2], we obtain $D(x)$ and linearize the dependency tree into a token-wise sequence that preserves head positions and relation labels. This representation captures long-range aspect–opinion dependencies and remains architecture-agnostic, allowing the same linearization $D(x)$ to be integrated into encoder-only, decoder-only, or encoder–decoder models under fine-tuning or in-context learning.

[2]https://spacy.io/

3.3 Instruction Tuning with PEFT

A causal language model is fine-tuned using QLoRA on instruction-formatted samples. Each training instance is a pair $\big(Prompt(x),\ y\big)$, where y denotes the set of quadruples $\{(a_i, c_i, s_i, o_i)\}_i$. Training employs a masked next-token likelihood objective in which loss is computed only over target tokens, while prompt tokens are excluded. The model optimizes the following objective:

$$\mathcal{L}_{\text{SFT}} = -\sum_{t=1}^{T} \log p_\theta\big(y_t \mid y_{<t}, Prompt(x)\big) \quad (3)$$

Here, t indexes target token positions ($t = 1, \ldots, T$); y_t is the t-th gold token of the target sequence y. The prompt $Prompt(x)$, including the task definition, input sentence x, and its dependency tree $D(x)$, conditions the decoder but is masked during loss computation. The conditional next-token distribution $p_\theta(\cdot \mid y_{<t}, Prompt(x))$ is parameterized by θ, representing the trainable LoRA (Hu et al., 2022a) adapter parameters, while the base model remains frozen.

3.4 Retrieval-Augmented In-Context Learning

During inference, we retrieve k demonstration pairs $S(x) = \{(x_j, y_j)\}_{j=1}^{k}$ based on the semantic similarity between the input x and training instances, computed using a frozen Sentence Transformer encoder. These retrieved examples are used as in-context demonstrations, each paired with its corresponding gold quadruple. The final prompt is constructed as follows:

$$Prompt_{\text{ICL}}(x) = \big[task;\ S(x);\ x;\ D(x)\big] \quad (4)$$

This formulation integrates semantically aligned examples with the syntax-aware representation $D(x)$, enabling the model to perform structured quadruple generation through one-pass decoding.

3.5 LLM-based Adjudication

To address annotation span inconsistencies, we introduce a blind judge J model that compares

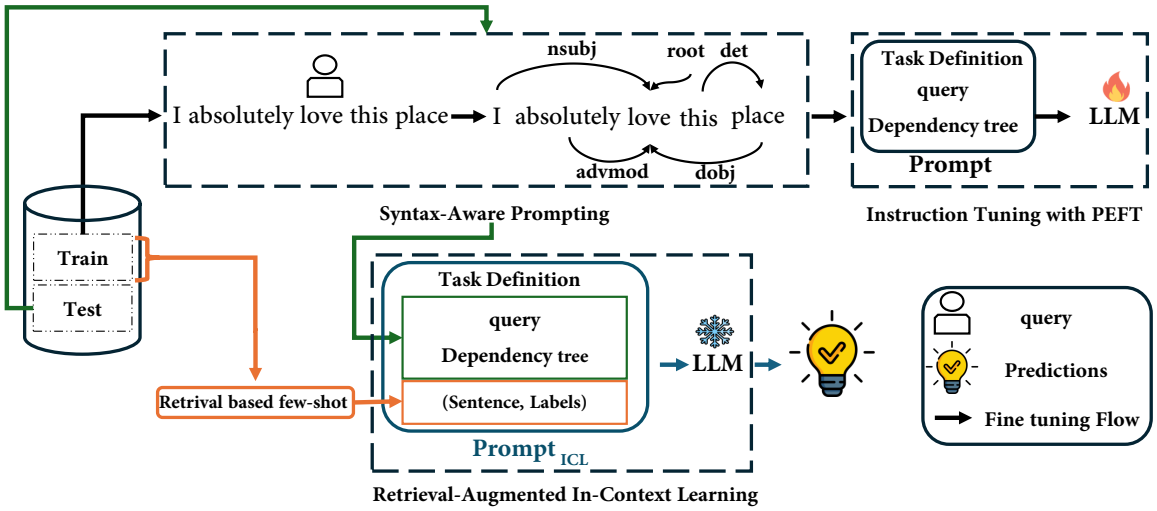


Figure 2: Overview of DRIFT that combining syntax-aware prompting, instruction tuning with PEFT, and retrieval-augmented in-context learning for ASQP.

two candidate quadruple sets U (prediction) and V (ground truth) labels by using the task definition, sentence x and its dependency tree $D(x)$ without identifying which set is the model prediction or ground truth. Formally, J returns a decision–reasoning pair:

$$(d, r) \;=\; J(U, V \mid [task, x, D(x)]) \qquad (5)$$

where $d \in \{0, 1\}$ and r is a textual reasoning. At evaluation time, if $d = 0$ we accept U for scoring, otherwise we accept V. This protocol allows semantically equivalent alternatives (e.g., *"delicious"* vs. *"delicious food"*) to be accepted when supported by the task definition and the dependency structure $D(x)$, thereby reducing unfair penalties caused by span boundary differences. In addition, the approach is model-agnostic and can be applied to predictions from any system, including fine-tuned models or pretrained large language models used in the ICL setting. In our experiments, the adjudication is performed using the pretrained o3 model.

4 Experiments

4.1 Datasets

We perform experiments on the ASQP benchmarks in the restaurant-domain Rest15 (SemEval-2015 Task 12) (Pontiki et al., 2015) and REST16 (SemEval-2016 Task 5) (Pontiki et al., 2016). Each sentence is labeled with one or more quadruples (a, c, s, o) denoting the *aspect term*, *category*, *sentiment polarity*, and *opinion term*. We use the official train/validation/test splits and follow standard ASQP preprocessing. Dataset statistics are provided in Table 1.

Dataset	Rest15			Rest16		
	Train	Valid	Test	Train	Valid	Test
N	834	209	537	1264	316	544
Positive	1005	252	453	1369	341	583
Neutral	34	14	37	62	23	40
Negative	315	81	305	558	143	176
Total	1354	347	795	1989	507	799

Table 1: Dataset statistics for Rest15 and Rest16. N denotes the number of sentences. Positive, Neutral, and Negative represent the number of quadruples for each sentiment.

4.2 Evaluation Metrics

We adopt F1 as the primary metric and also report precision and recall. A prediction is considered correct only when all four elements of a quadruple *aspect term*, *category*, *polarity*, and *opinion term* exactly match the corresponding gold labels. Evaluation uses normalized spans and exact matching across all four elements.

4.3 Implementation Details

We fine-tune Meta-Llama-3-8B-Instruct[3] using QLoRA with 4-bit NF4 quantization and train with LoRA (Low-Rank Adaptation) (Hu et al., 2022a) techniques with a single A100-40GB GPU. Training employs the paged AdamW optimizer with a learning rate of 2×10^{-4}, cosine learning rate de-

[3]https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

cay, and a warmup ratio of 0.05 over 4 epochs. The effective batch size is 16 achieved by accumulating gradients 4 times with a batch size of 4. For LoRA, we set the rank $r = 128$, resulting in approximately 500 million trainable parameters. This configuration allows us to capture high-dimensional task-specific information while training only a small fraction of the total model weights, significantly reducing the VRAM overhead compared to full fine-tuning. We choose LLaMA-3-8B-Instruct for its strong instruction-following ability and efficient trade-off between performance and compute, ideal for parameter-efficient fine-tuning.

4.4 Main Results

Table 2 presents a comprehensive comparison between DRIFT and state-of-the-art generative ASQP models on Rest15 and Rest16 benchmarks. DRIFT achieves 54.47% on Rest15 and 64.10% on Rest16, establishing new state-of-the-art performance with an average of 59.28%. DRIFT outperforms the strongest baseline, SimRP (Jian et al., 2025), by +1.17 points on Rest15 and +0.68 points on Rest16, achieving +0.92 average improvement. Compared to earlier methods, the gains are more substantial: +8.49 over GAS (Zhang et al., 2021b) on Rest15, +5.42 over ILO (Hu et al., 2022b), and +3.19 over IVLS (Nie et al., 2024). These consistent improvements across diverse baseline architectures demonstrate the effectiveness of dependency-guided reasoning using the dependency tree. Recent advanced methods incorporating various optimization strategies still fall short of DRIFT. MvP (Gou et al., 2023) with multi-view prompting achieves only 51.04% (Rest15) and 60.39% (Rest16), while GenDA (Wang et al., 2023) with generation-based augmentation reaches 50.01% and 60.88%. Even sophisticated learning mechanisms like MUL (Hu et al., 2023) and OTCL (Li et al., 2024) underperform DRIFT by considerable margins. These comparisons indicate that syntactic structure integration provides more fundamental improvements than optimization-focused approaches alone.

Beyond overall performance improvements, DRIFT demonstrates an impressive balance between precision and recall. It achieves 54.55% precision and 54.40% recall on Rest15, while performing even better on Rest16 with 62.76% precision and 65.51% recall. This balance effectively reduces both over-extraction, which lowers precision, and under-extraction, which lowers recall. In contrast, IVLS exhibits a clear imbalance (54.46% precision versus 48.53% recall on Rest15), reflecting a more conservative prediction behavior. DRIFT's balanced results are consistent with trends observed in syntax-aware models, where structural information helps stabilize generative outputs. Finally, the LLM-based adjudication results report judge-adjusted F1 scores of 72.41% on Rest15 and 77.20% on Rest16, accounting for semantically equivalent predictions. The gap between exact-match and semantic-match scores indicates that many apparent errors are actually semantically correct predictions that differ only in surface form. This evaluation method better reflects true semantic correctness, and DRIFT's strong results show that using syntax helps the model correctly capture sentiment relations even when word boundaries change. These results confirm that explicitly integrating dependency structure through reasoning mechanisms substantially improves generative ASQP performance and establishes DRIFT as a new state-of-the-art approach.

4.5 Syntactic Information Format Analysis

As shown in Table 3, we compare the effects of different dependency tree formats on ASQP performance using GPT-4o mini as our base language model. We evaluate five representations: SpaCy Default (compact token-head notation), JSON (structured key-value format), CoNLL-U (standard tabular format), Plain Text (explicit linguistic descriptions), and Hierarchical (tree structure). Each format encodes identical syntactic information but presents it differently to align with the language model's sequential processing nature. The results reveal format-dependent performance variations across datasets. On Rest15, Plain Text achieves the best F1-score 35.92%, while CoNLL-U performs best on Rest16 44.16%. More lengthy formats with explicit dependency descriptions (Plain Text, Hierarchical) generally outperform compact symbolic representations (SpaCy Default), suggesting that large language models benefit from human-readable syntactic information rather than compressed notations.

The variation in optimal formats across datasets suggests that the effectiveness of syntactic representation depends on linguistic complexity and dataset characteristics. Although more lengthy formats yield superior performance, we choose SpaCy Default for our PEFT and ICL experiments due to its computational efficiency. The compact notation significantly reduces token overhead during

Methods	Venue	Rest15			Rest16			Avg (F1)
		Pre	Rec	F1	Pre	Rec	F1	
GAS†	ACL21	45.31	46.70	45.98	54.54	57.62	56.04	51.01
Paraphrase	EMNLP21	46.16	47.72	46.93	56.63	59.30	57.93	52.43
DLO	EMNLP22	47.08	49.33	48.18	57.92	61.80	59.79	53.99
ILO	EMNLP22	47.08	50.38	49.05	57.58	61.17	59.32	54.19
MvP	ACL23	-	-	51.04	-	-	60.39	55.72
GenDA	*SEM23	49.74	50.29	50.01	60.08	61.70	60.88	55.45
MUL	ACL23	49.12	50.39	49.75	59.24	61.75	60.47	55.11
OTCL	CSCWD24	47.86	50.77	49.27	58.31	62.02	60.11	54.69
IVLS	Neurocomputing24	54.46	48.53	51.28	62.69	59.75	61.04	56.16
SimRP	AAAI25	53.12	53.50	53.30	62.74	64.12	63.42	58.36
DRIFT (Ours)		**54.55**	**54.40**	**54.47**	**62.76**	**65.51**	**64.10**	**59.28**
LLM-based Adjudication		*73.12*	*71.72*	*72.41*	*75.64*	*78.96*	*77.2*	*74.80*

Table 2: Comparison results in terms of precision (Pre, %), recall (Rec, %) and F1 score (F1, %).

fine-tuning and in-context learning, while still delivering competitive performance.

Format	Rest15	Rest16
SpaCy Default	34.27	43.08
JSON	35.11	42.94
CoNLL-U	34.55	**44.16**
Plain Text	**35.92**	43.54
Hierarchical	35.85	43.94

Table 3: F1-Scores (%) for Different Dependency Tree Formats

4.6 Zero and Few-Shot Performance in ICL

Table 4 demonstrates DRIFT's consistent superiority across zero-shot and few-shot settings. Smaller models show dramatic improvements Llama-3.1-8B advances from 15.41% to 30.01% (10-shot) on Rest15, nearly doubling performance. Larger models also benefit substantially: Llama-3.1-70B reaches 42.76% on Rest15 and 49.36% on Rest16, while GPT-4o achieves 54.77% on Rest16.

DRIFT outperforms the strongest baseline, SimRP, across all configurations, with gains of +2.44 F1 (GPT-4o, 10-shot) on Rest15 and +5.03 F1 (GPT-4o, 10-shot) on Rest16[4]. Figure 3 shows smooth, monotonic scaling as demonstrations increase, confirming that syntax-aware retrieval stabilizes reasoning and helps models effectively leverage additional examples.

4.7 Ablation study

Tables 5 and 6 evaluate the contributions of retrieval and dependency trees in PEFT and ICL paradigms. In fine-tuning settings (Table 5), full DRIFT achieves 51.01% and 63.17% F1-scores. Removing both components drops performance to 49.69% and 61.87% (-1.32, -1.30), while removing only dependency trees yields 50.13% and 61.34%. Notably, removing retrieval while retaining syntax achieves 53.78% and 64.10%, suggesting that syntactic guidance alone is highly effective when the model has been fine-tuned on task data.

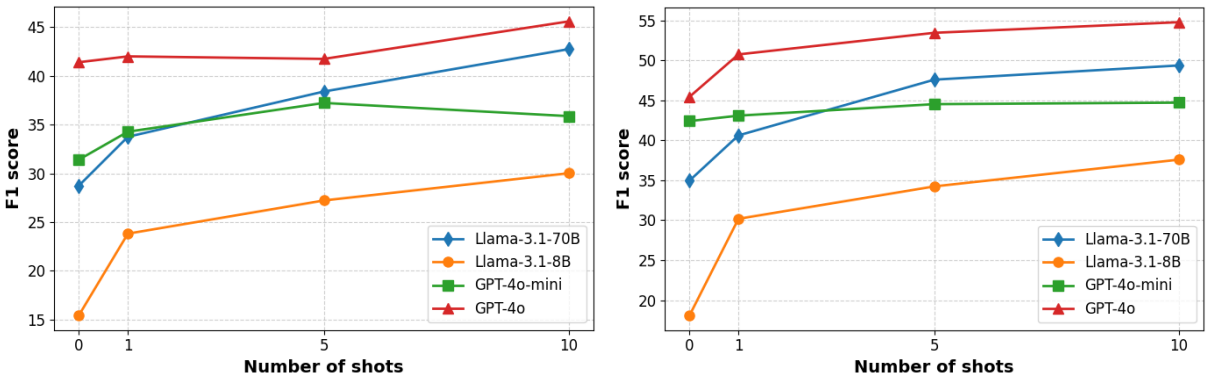
In ICL settings (Table 6), the importance of both components amplifies. Full DRIFT achieves 34.27% and 43.08%. Removing both components causes dramatic drops to 24.89% and 29.70% (-9.38, -13.38), substantially larger than in PEFT, highlighting their critical role in few-shot learning. Removing dependency trees yields 34.79% and 41.52%, while removing retrieval results in 31.40% and 42.40%, demonstrating that example-based guidance is essential for ICL effectiveness.

Overall, the ablation results highlight the complementary roles of syntactic structure and retrieval across different learning paradigms. In the PEFT, syntactic structure provides stronger independent benefits as the model has learned task patterns and in ICL, retrieval becomes more critical for providing task context through demonstrations. On Rest15, dependency trees contribute more in PEFT (+3.65 when comparing w/o Retrieval vs. w/o Dependency Tree), while retrieval contributes more in ICL (+3.39). This paradigm-dependent behavior demonstrates that DRIFT's effectiveness arises from the complementary interaction of its components, with each contributing in distinct ways across different learning settings. Notably, the PEFT configuration employs three retrieved shots, while the ICL configuration relies on a single retrieved shot.

[4]OpenAI API: https://openai.com/api/

Methods	Venue	Backbone	Rest15				Rest16			
			0-shot	1-shot	5-shot	10-shot	0-shot	1-shot	5-shot	10-shot
THOR	ACL23	Llama-3.1-8B	7.88	8.02	8.65	10.01	9.44	10.78	11.37	11.95
		Llama-3.1-70B	17.62	22.49	26.78	31.47	27.61	30.04	34.34	35.91
		GPT-4o-mini	18.18	21.52	24.29	29.36	29.83	31.50	32.84	34.05
		GPT-4o	30.79	33.37	35.12	36.81	37.23	39.22	42.05	43.57
LLMs for SA	NAACL24	Llama-3.1-8B	8.61	8.82	8.66	10.69	11.43	9.99	11.67	12.59
		Llama-3.1-70B	18.20	20.79	27.81	32.84	29.01	30.99	35.04	37.10
		GPT-4o-mini	19.80	22.16	24.93	30.11	31.43	29.42	35.84	35.45
		GPT-4o	34.56	35.62	36.29	37.08	39.15	39.61	43.36	45.00
SimRP	AAAI25	Llama-3.1-8B	8.88	9.26	14.13	15.26	12.55	15.46	16.71	21.60
		Llama-3.1-70B	18.12	26.83	35.56	38.19	28.71	33.73	40.91	43.81
		GPT-4o-mini	19.47	26.46	33.33	35.55	31.25	39.36	40.44	43.13
		GPT-4o	34.44	37.88	41.08	43.17	39.33	45.50	48.62	49.74
DRIFT(Ours)		Llama-3.1-8B	15.41	23.80	27.21	30.01	18.09	30.18	34.23	37.58
		Llama-3.1-70B	28.73	33.75	38.40	42.76	34.96	40.61	47.58	49.36
		GPT-4o-mini	31.40	34.27	37.23	35.87	42.40	43.08	44.52	44.71
		GPT-4o	41.41	42.01	41.75	**45.61**	45.44	50.74	53.44	**54.77**

Table 4: Experimental results of LLMs under zero-shot and few-shot settings on Rest15 and Rest16.



(a) (b)

Figure 3: Shot scaling on (a) REST15 and (b) REST16: F1 scores across different numbers of in-context examples ($k \in \{0, 1, 5, 10\}$) for Llama-3.1-8B, Llama-3.1-70B, GPT-4o-mini, and GPT-4o.

Methods	Rest15	Rest16
w/o Dependency Tree and Retrival	49.69	61.87
w/o Dependency Tree	50.13	61.34
w/o Retrieval	**53.78**	**64.10**
DRIFT	51.01	63.17

Table 5: Ablation studies in the PEFT paradigm

Methods	Rest15	Rest16
w/o Dependency Tree and Retrival	24.89	29.70
w/o Dependency Tree	**34.79**	41.52
w/o Retrieval	31.40	42.40
DRIFT	34.27	**43.08**

Table 6: Ablation studies in the ICL paradigm

4.8 Span Boundary Variation Analysis

Table 7 highlights how span boundary variation and aspect grounding differences influence ASQP evaluation. In many cases, model predictions express the same underlying semantics as the reference but differ in the exact boundaries chosen for aspects or opinion terms. For example, the model often extracts more complete opinion spans (e.g., capturing both Great and enjoy) or selects more concise spans that still preserve meaning. Such differences reflect natural linguistic variability. Annotators may include full descriptive phrases, whereas models tend to choose shorter spans that remain semantically equivalent. In several instances, the reference annotations themselves are incorrectly grounded, and the model's prediction more accurately reflects the sentence. For instance, attributing delivers to spot rather than modern Japanese, or assigning fair to

Sentence (Query)	Reference	Prediction	Reasoning
Great place to relax and enjoy your dinner.	[place, ambience general, positive, Great]	[place, ambience general, positive, Great] [place, ambience general, positive, enjoy]	Prediction tuple is better because it includes both valid opinion words Great and enjoy so its quadruples are more complete than the reference tuple, which omits one of them.
The hot dogs are good, yes, but the reason to get over here is the fantastic pork croquette sandwich, perfect on its supermarket squishy bun	[hot dogs, food quality, positive, good] [pork croquette sandwich, food quality, positive, fantastic] [bun, food quality, positive, perfect]	[hot dogs, food quality, positive, good] [pork croquette sandwich, food quality, positive, fantastic]	Prediction tuples are correct; the reference tuple is wrong because it misassigns perfect to bun instead of the sandwich.
In the end you end up with a fair tab and NOTHING BUT A GREAT TIME !!!	[NULL, restaurant prices, positive, fair], [NULL, restaurant general, positive, GREAT]	[tab, restaurant prices, positive, fair] [NULL, restaurant general, positive, GREAT TIME]	Prediction tuples are correct because they include the aspect tab, which the reference tuples miss.
Which lets face it ... at times it's a good thing.	[service, service general, positive, good]	[NULL, restaurant miscellaneous, positive, good]	Prediction tuple is better, assigning restaurant miscellaneous, while the reference tuple wrongly adds an ungrounded service aspect.
I've had my fair share of modern Japanese, and this spot delivers.	[modern Japanese, food quality, positive, delivers]	[spot, restaurant general, positive, delivers]	The prediction tuple is correct, linking delivers to the restaurant (spot, category restaurant general); the reference tuple wrongly assigns it to modern Japanese.

Table 7: Examples Where the Model Correctly Identifies Quadruples (O3 Evaluation). Each entry includes the input sentence, reference, predictions, and reasoning.

the explicit aspect tab instead of using a NULL-aspect placeholder. These examples show that span boundary mismatches are not always model errors; they often arise from annotation inconsistencies or subjective span choices. Although these variations rarely affect sentiment polarity or the core aspect–opinion relationship, exact-match scoring penalizes them equally. This indicates that semantic-level matching or adjudication-based evaluation would yield a fairer assessment, especially for implicit aspects, multi-word expressions, and flexible opinion phrasing.

5 Conclusion

We proposed DRIFT, a novel unified framework for Aspect Sentiment Quadruple Prediction (ASQP) that combines dependency-based syntactic reasoning with retrieval-augmented instruction tuning. DRIFT establishes new state-of-the-art results on Rest15 and Rest16, and our analyses reveal complementary effects across learning regimes: explicit structural (syntactic) guidance is the main contributor under parameter-efficient fine-tuning, while retrieved few-shot demonstrations account for most of the gains in in-context learning. We also introduced an LLM-based adjudicator that mitigates the brittleness of exact-match evaluation by recognizing semantically equivalent predictions despite minor span-boundary differences. Overall, DRIFT provides a practical baseline for structure-aware ASQP and supports more reliable evaluation for this task.

Limitations

Despite DRIFT's effectiveness, several limitations remain. Its reliance on external dependency parsers can cause error propagation, particularly for informal, ungrammatical, or domain-specific text. The benefits of retrieval-augmented demonstrations are sensitive to database quality and coverage, making DRIFT less reliable under low-resource conditions or significant domain shift. Moreover, the model continues to struggle with neutral sentiment and rare aspect categories, where polarity cues are subtle and supervision is limited. Although the LLM-based adjudicator improves evaluation robustness, it introduces additional cost and potential bias, highlighting the need for complementary human verification. Future work includes developing more robust alternatives to external parsing, more robust retrieval strategies, and broader evaluation across multilingual and cross-domain settings.

References

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 33 (NeurIPS 2020)*. Curran Associates, Inc.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL 2021)*, pages 340–350, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186. Association for Computational Linguistics.

Hao Fei, Bobo Li, Qian Liu, Lidong Bing, Fei Li, and Tat-Seng Chua. 2023. Reasoning implicit sentiment with chain-of-thought prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1171–1182, Toronto, Canada. Association for Computational Linguistics.

Zhibin Gou, Qingyan Guo, and Yujiu Yang. 2023. MvP: Multi-view prompting improves aspect sentiment tuple prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4380–4397, Toronto, Canada. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022a. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Mengting Hu, Yinhao Bai, Yike Wu, Zhen Zhang, Liqi Zhang, Hang Gao, Shiwan Zhao, and Minlie Huang. 2023. Uncertainty-aware unlikelihood learning improves generative aspect sentiment quad prediction. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13481–13494, Toronto, Canada. Association for Computational Linguistics.

Mengting Hu, Yike Wu, Hang Gao, Yinhao Bai, and Shiwan Zhao. 2022b. Improving aspect sentiment quad prediction via template-order data augmentation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7889–7900, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yan Cathy Hua, Paul Denny, Jörg Wicker, and Katerina Taškova. 2025. Data-efficient adaptation and a novel evaluation method for aspect-based sentiment analysis. *arXiv preprint arXiv:2511.03034*.

Binxuan Huang and Kathleen M. Carley. 2019. Syntax-aware aspect level sentiment classification with graph attention networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5469–5477. Association for Computational Linguistics.

Zhongquan Jian, Yanhao Chen, Jiajian Li, Shaopan Wang, Xiangjian Zeng, Junfeng Yao, Xinying An, and Qingqiang Wu. 2025. Simrp: Syntactic and semantic similarity retrieval prompting enhances aspect sentiment quad prediction. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI 2025)*, volume 39, pages 24248–24256.

Zhijun Li, Zhenyu Yang, Yiwen Li, and Xiaoyang Li. 2024. Opinion-tree-guided contrastive learning for aspect sentiment quadruple prediction. *2024 27th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pages 1944–1951.

Bin Liang, Hang Su, Lin Gui, Erik Cambria, and Ruifeng Xu. 2022. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 235:107643.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2511–2522, Singapore. Association for Computational Linguistics.

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 35th AAAI Conference on Artificial Intelligence (AAAI-21)*, pages 13543–13550. AAAI Press.

Yu Nie, Jianming Fu, Yu Zhang, and Chao Li. 2024. Modeling implicit variable and latent structure for aspect-based sentiment quadruple extraction. *Neurocomputing*, 586:127642.

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 Eryğit. 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.

Kevin Scaria, Himanshu Gupta, Siddharth Goyal, Saurabh Sawant, Swaroop Mishra, and Chitta Baral. 2024. InstructABSA: Instruction learning for aspect based sentiment analysis. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 720–736, Mexico City, Mexico. Association for Computational Linguistics.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6107–6122, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 380–385. Association for Computational Linguistics.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z. Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI-20)*, pages 9122–9129. Association for the Advancement of Artificial Intelligence.

An Wang, Junfeng Jiang, Youmi Ma, Ao Liu, and Naoaki Okazaki. 2023. Generative data augmentation for aspect sentiment quad prediction. In *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 128–140, Toronto, Canada. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35 (NeurIPS 2022)*, pages 24824–24837. Curran Associates, Inc.

Soyoung Yang, Hojun Cho, Jiyoung Lee, Sohee Yoon, Edward Choi, Jaegul Choo, and Won Ik Cho. 2025. Single ground truth is not enough: Adding flexibility to aspect-based sentiment analysis evaluation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 12071–12096, Albuquerque, New Mexico. Association for Computational Linguistics.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4568–4578. Association for Computational Linguistics.

Fan Zhang, Wenbin Zheng, and Yujie Yang. 2024. Graph convolutional network with syntactic dependency for aspect-based sentiment analysis. *International Journal of Computational Intelligence Systems*, 17(1):37.

Mao Zhang, Yongxin Zhu, Zhen Liu, Zhimin Bao, Yunfei Wu, Xing Sun, and Linli Xu. 2023. Span-level aspect-based sentiment analysis via table filling. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9273–9284, Toronto, Canada. Association for Computational Linguistics.

Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. 2021a. Aspect sentiment quad prediction as paraphrase generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9209–9219. Association for Computational Linguistics.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021b. Towards generative 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 2: Short Papers)*, pages 504–510, Online. Association for Computational Linguistics.

Guangmin Zheng, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2024. Instruction tuning with retrieval-based examples ranking for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4777–4788, Bangkok, Thailand. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Neural Information Processing Systems (NeurIPS) 2023 – Datasets and Benchmarks Track*.