PASTEL : Polarity-Aware Sentiment Triplet Extraction with LLM-as-a-Judge

Aaditya Bodke, Avinoor Singh Kohli, Hemant Pardeshi, Prathamesh Bhosale
University of Illinois Urbana-Champaign
{abodke2, askohli2, hemant3, bhosale5}@illinois.edu

Abstract

Aspect Sentiment Triplet Extraction (ASTE) is a subtask of Aspect-Based Sentiment Analysis (ABSA) that aims to extract aspect terms, corresponding opinion terms, and their associated sentiment polarities from text. Current end-to-end approaches, whether employing Large Language Models (LLMs) or complex neural network structures, struggle to effectively model the intricate latent relationships between aspects and opinions. Therefore, in this work, we propose *Polarity-Aware Sentiment Triplet Extraction with LLM-as-a-judge (PASTEL)*, a novel pipeline that decomposes the ASTE task into structured subtasks. We employ finetuned LLMs to separately extract the aspect and opinion terms, incorporating a polarity-aware mechanism to enhance opinion extraction. After generating a candidate set through the Cartesian product of the extracted aspect and opinion-sentiment sets, we leverage an LLM-as-a-Judge to validate and prune these candidates. Experimental evaluations demonstrate that PASTEL outperforms existing baselines. Our findings highlight the necessity of modular decomposition in complex sentiment analysis tasks to fully exploit the capabilities of current LLMs.

1 Introduction and Related Work

Aspect-Based Sentiment Analysis (ABSA) is a fine-grained sentiment analysis task that aims to determine the sentiment polarity associated with specific aspects in a given text (Pontiki et al., 2014). A key subtask of ABSA is **Aspect Sentiment Triplet Extraction (ASTE)** (Peng et al., 2019), which involves extracting triplets of (aspect term, opinion term, sentiment polarity) from text. ASTE provides a more structured understanding of sentiments beyond document or sentence-level analysis. It enables deeper insights and decision-making from user feedback, thereby driving product and service improvements. However, accurately extracting these triplets remains a significant challenge due to the complexity of aspect-opinion interactions and the implicit nature of sentiment dependencies in natural language.

Several methodologies have been proposed for ABSA and its subtasks, such as sequence tagging (Xu et al., 2020; Yan et al., 2021), sequence-to-sequence generation (Naglik and Lango, 2024), table-filling (Zhang et al., 2022), graph-based (Li et al., 2021; Yin and Zhong, 2024; Jian et al., 2025), contrastive learning (Sun et al., 2024), and span classification (Zhao et al., 2020; Xu et al., 2021; Liang et al., 2022). However, these methods have several issues like sensitivity to parsing errors, failure to model long-range dependencies, and struggle with implicit sentiment reasoning

Recently, large language models (LLMs) have demonstrated remarkable performance in ABSA due to their contextual understanding, instruction-following and in-context learning abilities (Scaria et al., 2023; Zhang et al., 2023; Yang et al., 2024; Fan et al., 2025). However, despite excelling at **Aspect Term Extraction (ATE)** and **Opinion Term Extraction (OTE)**, LLMs struggle with structured triplet extraction due to the difficulty in capturing complex latent dependencies between aspect and opinion terms (§A.2.1). Additionally, they often fail to distinguish between aspect and opinion terms, leading to extraction errors (§A.2.2). This highlights the need for a decoupled approach that integrates contextual understanding of LLMs with structured reasoning to enhance triplet extraction accuracy. The argument against such pipeline-based methods is that they ignore the interaction among triplets, which could result in error propagation. To alleviate this problem, we propose the use of an LLM-as-a-Judge to prune the final results, as LLMs have been shown to achieve a high agreement rate with human experts (Zheng et al., 2023; Gu et al., 2024).

To address the challenges of previous methods, we propose **Polarity-Aware Sentiment Triplet Ex-**

Findings of the Association for Computational Linguistics: ACL 2025, pages 25523–25533
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

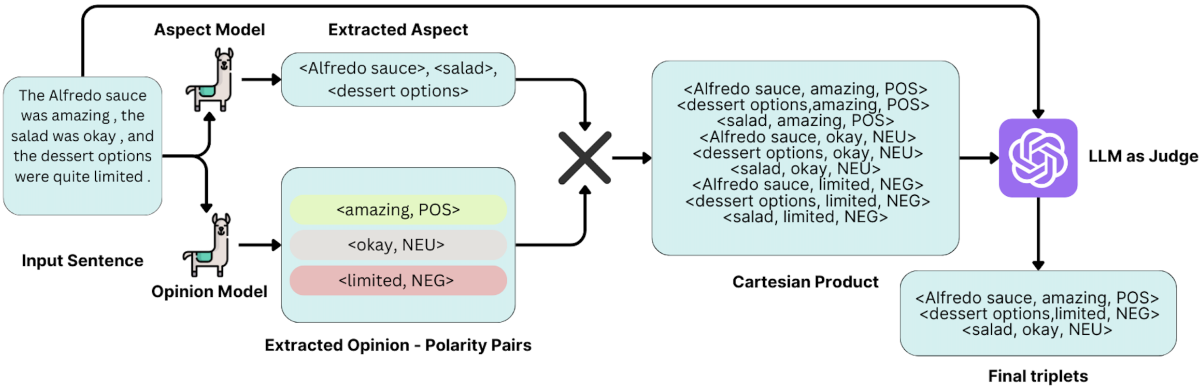
Aspect Model
Extracted Aspect
<Alfredo sauce>, <salad>, <dessert options>
The Alfredo sauce was amazing , the salad was okay , and the dessert options were quite limited .
Input Sentence
Opinion Model
<amazing, POS>
<okay, NEU>
<limited, NEG>
Extracted Opinion - Polarity Pairs
<Alfredo sauce, amazing, POS>
<dessert options,amazing, POS>
<salad, amazing, POS>
<Alfredo sauce, okay, NEU>
<dessert options, okay, NEU>
<salad, okay, NEU>
<Alfredo sauce, limited, NEG>
<dessert options, limited, NEG>
<salad, limited, NEG>
Cartesian Product
LLM as Judge
<Alfredo sauce, amazing, POS>
<dessert options,limited, NEG>
<salad, okay, NEU>
Final triplets

Figure 1: System diagram.

traction with LLM-as-a-Judge (PASTEL)**, an approach that decomposes ASTE into structured subtasks. Our approach consists of:

- **Structured Decomposition of ASTE:** We first independently extract aspect and opinion terms using task-specific instruction-tuned LLMs.
- **Polarity-Aware Opinion Term Extraction (PAOTE):** To mitigate the problem of sentiment misclassification, we have introduced a polarity-aware mechanism that improves and refines OTE.
- **Triplet Validation with LLM-as-a-Judge:** We construct a comprehensive set candidate triplets as the Cartesian product of the extracted aspects and opinion-sentiment pairs, which are validated and pruned by an LLM-as-a-Judge to ensure higher precision and alignment with human annotations.

2 Proposed Approach

2.1 Task Formulation

Let S_i denote the i-th review sentence in the training dataset. Each S_i includes a set of aspect terms, represented by $A_i = \{a^i_1, a^i_2, \ldots, a^i_m\}$ and their corresponding opinion terms $O_i = \{o^i_1, o^i_2, \ldots, o^i_m\}$. Each opinion and aspect term is either a set of tokens extracted from the sentence or *"none"* when no relevant term is present. The sentiment polarities for the aspect terms are denoted by $SP_i = \{sp^i_1, sp^i_2, \ldots, sp^i_m\}$ where sp^i_k belongs to the set $\{\text{pos}, \text{neg}, \text{neu}\}$. The ASTE task is then formulated as follows:

$$\textbf{PASTEL}_{\text{ASTE}}(S_i) = [A_j, O_j, SP_j], \quad j \in [1, m] \tag{1}$$

2.2 System Architecture

We outline the design of **PASTEL** in Figure [1]. Given a sentence, we aim to predict the triplet(s) of aspect, opinion term and polarity. Our approach splits the extraction of aspect and opinion terms into two parallel pipelines. We extract the opinion terms by using an LLM finetuned for the task in a polarity-aware manner (§A.3.1), giving us opinion terms corresponding to positive, negative and neutral polarities. We simultaneously extract the aspect terms via another finetuned LLM. This is followed up by computing the Cartesian product of the results of the two pipelines to get a list of triplet candidates. Finally, we use an LLM-as-a-judge to prune the list and produce the final triplets.

Polarity-Aware Opinion Term Extraction (PAOTE): The goal of this pipeline is to extract the opinion term set O_i from each review sentence S_i and map them with their respective polarity from SP_i giving us:

$$P_i = \left\{ (sp^i_k, o^i_k) \right\} \tag{2}$$

where P_i is the set of all opinions-polarity pairs.

Given that sp^i_j can take on three possible values, separate entries are generated in dataset for each value. If there is more than one opinion present in the sentence corresponding to a sentiment, all of them are concatenated to a list. For instance, consider (s_i, {([o_1], $sp_1(positive)$), ([o_2], $sp_2(negative)$}). This entry is expanded to (s_i, [o_1], sp_1), (s_i, [o_2], sp_2) and (s_i, [$none$], $sp_3(neutral)$), Here *none* depicts the absence of opinion term corresponding to particular sentiment polarity (For example sentence refer §A.2.3).

With this expanded dataset, we finetune the model using task-specific prompts (§A.3.1). This

Methods	14Res	14Lap	15Res	16Res
Baseline				
HAST+TOWE	75.10	67.50	68.45	75.71
SpanMlt	83.98	80.61	78.91	85.33
PAOTE				
LLaMA 3.2 1B	**86.47**	**81.03**	**79.34**	**85.67**

Table 1: Opinion Term Extraction F1 Scores on the 14Res, 14Lap, 15Res, and 16Res datasets. For full results refer to Table [5].

Methods	14Res	14Lap	15Res	16Res
HAST+TOWE	82.56	79.14	79.84	81.44
SpanMlt	87.42	84.51	81.76	85.62
Full Finetuning				
Instruct-ABSA	92.30	92.76	76.64	81.48
Our Approach				
LLaMA 3.2 1B	**94.18**	**94.26**	**86.07**	**86.63**

Table 2: Aspect Term Extraction F1 Scores on the 14Res, 14Lap, 15Res, and 16Res datasets. For full results refer to Table [6].

prompt includes auxiliary input in the form of sentiment polarity sp_j^i, and the model's output will be the corresponding opinion term's $o_1, o_2, ..$ associated with that polarity. This process is iterated over all possible values of sp_j^i during inference for a given sentence S_i to construct final prediction set P_i.

Aspect Term Extraction (ATE): The model is fine-tuned on a dataset consisting of sentences annotated with corresponding aspect terms. During finetuning, model was provided a task-specific prompt along with some contrastive examples following (Scaria et al., 2023) (§A.3.2). If no relevant aspects present for a given sentence, models outputs *none*. Formally, the task is defined as:

$$A_i = \text{LM}_{\text{ATE}}(S_i)$$

where LM_{ATE} refers to the fine-tuned model that performs ATE, S_i is the input sentence and A_i is the set of aspect terms.

Cartesian Product for Candidate Set: We compute cartesian product between the set of extracted opinion-polarity pairs P_i and aspect terms A_i to generate a candidate triplet set T_i. The triplets can be shown as:

$$T_i = \{(a_i^k, o_i^j, sp_i^j) \mid a_i^k \in A_i, (o_i^j, sp_i^j) \in P_i\}$$

LLM as a Judge: All candidate triplets from T_i are passed to GPT-4o, which evaluates the given sentence S_i to determine if each triplet has enough supporting evidence. We employ a chain-of-thought prompt template (See example §A.3.3), which frames the task as an entailment problem. LLMs possess implicit and rich commonsense knowledge about the world. Chain-of-Thought prompt improves LLM's ability to access this implicit knowledge (Fei et al., 2023), generating reasoning trace that improves both accuracy and transparency of the output (Saha et al., 2025).

Each triplet is transformed into a natural language query that prompts the model to verify the relationship between the triplets terms. The model is required to assess whether the evidence in the input sentence supports the query. The prompt includes some few shot examples demonstrating the reasoning behind each decision. The LLM's output is further processed to extract triplet terms from the queries it determined to be supported by the sentence.

3 Experiments

We conducted our experiments using the Llama 3.2 (1B) models and evaluated the proposed system component-wise for task-specific extraction i.e., the ATE and OTE pipelines, along with the overarching task of ASTE. For ATE and OTE tasks we have benchmarked the approach against (Zhao et al., 2020).

For ASTE, we compared our pipeline against the performance of GPT-4o in zero-shot and few-shot settings as highlighted in (Sun et al., 2024). Additionally we also evaluated our pipeline against existing methods, such as Span-BART (Yan et al., 2021), ASTE Transformer (Naglik and Lango, 2024), MiniConGTS (Sun et al., 2024), and InstructABSA (Scaria et al., 2023).

Dataset and Metrics: For the ASTE task, (Peng et al., 2019) introduced a dataset that builds upon the SemEval 2014 Task dataset (Pontiki et al., 2014). Later (Xu et al., 2020) released an improved version [1], which includes explicitly defined triplets for cases where a single opinion span is associated

[1] https://github.com/xuuuluuu/SemEval-Triplet-data?tab=readme-ov-file

Methods	**14Res**			**14Lap**			**15Res**			**16Res**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Sequence-tagging												
Span-BART (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
Seq2seq												
ASTE Transformer (Naglik and Lango, 2024)	76.43	75.71	76.06	67.58	62.48	64.90	**72.91**	**71.34**	**72.10**	76.27	76.12	76.19
Constrastive Learning												
MiniConGTS (Sun et al., 2024)	76.1	75.08	75.59	66.82	60.68	63.61	66.50	63.86	65.15	75.52	74.14	74.83
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
Instruct-ABSA (Scaria et al., 2023)	-	-	71.17	-	-	61.86	-	-	60.63	-	-	70.72
Ours												
PASTEL	**81.95**	**79.81**	**80.87**	**67.83**	**62.61**	**65.22**	72.54	71.18	71.86	**80.89**	**77.43**	**79.12**

Table 3: PASTEL vs Baseline: Results on the 14Res, 14Lap, 15Res, and 16Res datasets

with multiple aspect terms. We evaluate all models based on their Precision, Recall, and F1 score.

Triplet Pruning using LLM-as-a-Judge: We experimented with various prompt templates for pruning candidate triplets (Gu et al., 2024). The first prompt type scored each triplet based on the supporting evidence. However, this method was unreliable as the ratings were inconsistent, making it difficult to set a meaningful threshold (§A.2.4). We then tried a yes/no type of prompt, which was more consistent but lacked reasoning for complex sentences (§A.2.5). To address this, we incorporated chain of thought along with few shot examples. This approach not only gave stabilized results but also enhanced the model's reasoning, resulting in transparent and accurate pruning.

4 Results

We evaluated PAOTE (LLaMA 3.2-1B) against two strong baselines—HAST + TOWE (Li et al., 2018; Fan et al., 2019) and SpanMLT (Zhao et al., 2020). As shown in Table [1], PAOTE attains the highest F1 scores, underscoring the effectiveness of our approach.

Table [2] demonstrates that the full finetuning significantly improves accuracy, particularly recall in LLaMA 3.2 1B for ATE task. It achieves the highest F1 across all datasets, outperforming SpanMLT (Zhao et al., 2020) and InstructABSA (Scaria et al., 2023).

The results in Table [3] indicate PASTEL achieves state-of-the-art performance across most datasets in ASTE, outperforming existing methodologies. PASTEL also performs comparably to the ASTE transformer (Naglik and Lango, 2024) on 15Res dataset, further validating its effectiveness. The observed performance loss in 15Res arises primarily from dataset limitations. As illustrated below in Table [4], there is a strong positive correlation between the number of training sentences, average sentence length, and the change (Δ) in F1 score over previous state-of-the-art method (Naglik and Lango, 2024). 15Res has fewer training sentences and shorter average sentence lengths compared to 14Lap, 14Res and 16Res, limiting contextual information essential for effective learning. PASTEL excels when trained on abundant and contextually rich data, as reflected by its superior F1 scores on larger datasets like 14Res and 16Res. Conversely, the limited data and shorter contexts in 15Res restrict such performance gains.

5 Conclusion

We present PASTEL, a novel approach for ASTE using LLMs, which leverages ATE, PAOTE and LLM-as-a-Judge for structured triplet validation. According to Tables [1] and [2], our approach consistently performs better than the benchmarks on both ATE and OTE, showing the effectiveness of full fine-tuning to achieve optimal model performance. By splitting the pipeline to extract the aspects and opinion terms separately, we are able to mitigate the limitation of LLMs in understanding the syntactic and semantic dependencies between the aspect and opinion terms, ensuring more precise extraction of triplets. PASTEL outperforms on majority of the datasets and remains on par with ASTE Transformer on 15Res. Additionally, LLM-as-a-Judge effectively prunes noisy candidates generated with Cartesian product of the outputs of the

Dataset	ΔF1 Score	Avg. Sentence Length	Training Sentences
15Res	−0.24	72.67	605
14Lap	+0.32	93.62	906
16Res	+2.93	74.55	857
14Res	+4.81	85.44	1266

Table 4: Dataset characteristics showing Δ F1 score from previous SOTA (Naglik and Lango, 2024), average sentence length, and number of training samples.

two pipelines. We leverage LLMs' reasoning abilities (via chain-of-thought prompts) to ensure that extracted triplets are coherent within each sentence.

Both modular fine-tuning strategy and LLM-as-a-Judge mechanism contribute significantly to better accuracy, with a better alignment of human annotations. Overall, all of these results verify the impact of our structured pipeline approach, mirroring the need for modular decomposition and LLM-based verification within ASTE applications.

6 Limitations

While PASTEL performs well across datasets, some limitations persist. First, as shown in Table [4], ΔF1 scores correlate with average sentence length and training size. Richer datasets like 14Res and 16Res yield higher gains, while 15Res shows limited improvement due to shorter and fewer training examples. This aligns with findings from (Naglik and Lango, 2024), where pretraining on larger sentiment datasets improved performance on 15Res. Thus, the model remains susceptible to the size and richness of the training data, as reflected by its lower performance on 15Res.

Second, PASTEL sometimes misses aspect terms when they are proper nouns or named entities (see §A.2.6), suggesting that the aspect extraction component requires better handling of such cases. Third, the modular pipeline may introduce error propagation, as aspect and opinion terms are extracted independently and paired via Cartesian product. Errors in early stages can result in invalid triplets, and while the LLM-as-a-Judge with chain-of-thought reasoning filters these, its effectiveness depends on input quality from earlier components.

Finally, our use of LLMs introduces additional computational cost. To address this, we selected a compact model, LLaMA 3.2 with approximately one billion parameters, which provides a favorable balance between efficiency and performance. Future work may investigate further optimizations in model size and inference latency.

7 Ethical Considerations

In our experiments, the datasets have primarily focused upon the reviews of the products and services on e-commerce platform and restaurants, which inherently have lower risk of having any offensive content. We have considered datasets that are widely accepted and extensively referenced within the academic community. We have thoroughly reviewed data to scrutinize data for any potential bias against gender, race, and marginalized groups. Despite our precautions, there might be a possible case that our model generates sentiment assessments that could be perceived as offensive. In such cases, we reserve the right to limit or modify the use of our technology to prevent misuse.

References

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, Abu Dhabi, UAE. Association for Computational Linguistics.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2509–2518, Minneapolis, Minnesota. 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. *arXiv preprint arXiv:2305.11255*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. 2024. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

Zhongquan Jian, Daihang Wu, Shaopan Wang, Yancheng Wang, Junfeng Yao, Meihong Wang, and

Qingqiang Wu. 2025. AGCL: Aspect graph construction and learning for aspect-level sentiment classification. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 841–854, Abu Dhabi, UAE. Association for Computational Linguistics.

Ruifan Li, Hao Chen, Fangxiang Feng, Zhanyu Ma, Xiaojie Wang, and Eduard Hovy. 2021. Dual graph convolutional networks 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 6319–6329, Online. Association for Computational Linguistics.

Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. 2018. Aspect term extraction with history attention and selective transformation. In *International Joint Conference on Artificial Intelligence*.

Shuo Liang, Wei Wei, Xian ling Mao, Yuanyuan Fu, Rui Fang, and Dangyang Chen. 2022. Stage: Span tagging and greedy inference scheme for aspect sentiment triplet extraction. *ArXiv*, abs/2211.15003.

Iwo Naglik and Mateusz Lango. 2024. Aste transformer modelling dependencies in aspect-sentiment triplet extraction. *Preprint*, arXiv:2409.15202.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2019. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *AAAI Conference on Artificial Intelligence*.

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.

Swarnadeep Saha, Xian Li, Marjan Ghazvininejad, Jason Weston, and Tianlu Wang. 2025. Learning to plan reason for evaluation with thinking-llm-as-a-judge. *Preprint*, arXiv:2501.18099.

Kevin Scaria, Himanshu Gupta, Siddharth Goyal, Saurabh Arjun Sawant, Swaroop Mishra, and Chitta Baral. 2023. Instructabsa: Instruction learning for aspect based sentiment analysis. *Preprint*, arXiv:2302.08624.

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.

Lu Xu, Yew Ken Chia, and Lidong Bing. 2021. Learning span-level interactions for aspect sentiment triplet extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4755–4766, Online. Association for Computational Linguistics.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2339–2349. Association for Computational Linguistics.

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.

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, Torino, Italia. ELRA and ICCL.

Shuo Yin and Guoqiang Zhong. 2024. Textgt: A double-view graph transformer on text for aspect-based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19404–19412.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023. Sentiment analysis in the era of large language models: A reality check. *Preprint*, arXiv:2305.15005.

Yice Zhang, Yifan Yang, Yihui Li, Bin Liang, Shiwei Chen, Yixue Dang, Min Yang, and Ruifeng Xu. 2022. Boundary-driven table-filling for aspect sentiment triplet extraction. In *Conference on Empirical Methods in Natural Language Processing*.

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.

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. *Preprint*, arXiv:2306.05685.

A Appendix

A.1 Experimental Setup

We perform all our finetuning experiments on an A100 GPU using SFT trainer [2] for approximately 300 hours. We have used standard instruction-tuning with task-specific prompts as mentioned ahead (§A.3.1, §A.3.2). We have experimented across learning rates from [1e-6 to 1e-4], warmup ratio [0.1 to 0.5]. To ensure that the generation is reliable and consistent we kept the temperature of the generation at 0.1 and top k sampling of 1.

A.1.1 Aspect Extraction

We finetune the Llama 3.2 1B model for 5 epochs using AdamW optimizer with a learning rate of $2e^{-5}$.

A.1.2 Polarity-Aware Opinion Extraction

We finetune the Llama 3.2 1B model for 5 epochs using AdamW optimizer with a learning rate of $2e^{-5}$.

A.2 Examples

A.2.1 Motivation - Correlation failure

With the rise of powerful LLMs, ABSA is a natural problem that these LLMs can be applied to. To verify this, we tested GPT-4o on some examples from the dataset.

A shortcoming of GPT-4o was that it failed to understand the correlation between aspect and opinion terms. An example is shown below to illustrate this issue.

GPT-4o performance example 1

Input sentence: I also enjoy the fact that my MacBook Pro laptop allows me to run Windows 7 on it by using the VMWare program.

Actual triplets: (Windows 7, enjoy, positive), (VMWare program, enjoy, neutral)

Triplets extracted by GPT-4o: (fact, enjoy, positive), (MacBook Pro, -, neutral),(laptop, -, neutral), (Windows 7, run, neutral), (VMWare program, using, neutral)

A.2.2 Motivation - Lack of Distinction

As shown in the below example, GPT-4o could not distinguish between aspect and opinion terms, extracting the same term for both.

GPT-4o performance example 2

Input sentence: it is of high quality , has a killer GUI , is extremely stable , is highly expandable , is bundled with lots of very good applications , is easy to use , and is absolutely gorgeous.

Actual triplets: (quality, high, positive), (GUI, killer, positive), (applications, good, positive), (use, easy, positive)

Triplets extracted by GPT-4o: (quality, high, positive), (GUI, killer, positive), (stable, stable, positive), (expandable, expandable, positive), (applications, good, positive), (use, easy, positive), (gorgeous, gorgeous, positive)

A.2.3 Dataset Expansion

As mentioned in the OTE section, the original dataset was expanded to three entries for each entry. For PAOTE, each sentence is expanded into three entries (positive, negative, neutral), increasing the training set from 3,634 to 10,902 entries, improving sentiment-specific opinion extraction. Here's an example to better illustrate the modifications.

> "The food was good, but the service was bad."

In this case the aspect *food* is linked to the *positive* sentiment, while the aspect *service* is linked to *negative* sentiment. If a particular polarity does not match any opinion term **none** entry is created. Thus, corresponding to each sentence there are three entries in training set. The expanded data set looks like:

Entry 1: S_1, O_1 = good, SP_1 = positive

Entry 2: S_1, O_2 = bad, SP_2 = negative

Entry 3: S_1, O_2 = none, SP_2 = neutral

A.2.4 Rating Prompt Output example

The rating prompt output was often inconsistent in its rating system, assigning a lower score for valid triplets. Here's an example illustrating this issue.

[2] https://huggingface.co/docs/trl/en/sft_trainer

Methods	14Res			14Lap			15Res			16Res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Baseline												
HAST+TOWE	-	-	75.10	-	-	67.50	-	-	68.45	-	-	75.71
SpanMlt	-	-	83.98	-	-	80.61	-	-	78.91	-	-	85.33
PAOTE												
Llama 3.2 1B	**86.12**	**86.82**	**86.47**	**79.92**	**82.14**	**81.03**	**78.70**	**80.0**	**79.34**	**84.19**	**87.15**	**85.67**

Table 5: Opinion Term Extraction Results on the 14Res, 14Lap, 15Res, and 16Res datasets.

Methods	14Res			14Lap			15Res			16Res		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
SpanMlt	-	-	87.42	-	-	84.51	-	-	81.76	-	-	85.62
Full Finetuning												
Instruct-ABSA	-	-	92.30	-	-	92.76	-	-	76.64	-	-	81.48
Our Approach												
LLaMA 3.2 1B	**93.58**	**94.78**	**94.18**	**93.67**	**94.85**	**94.26**	**85.50**	**86.64**	**86.07**	**86.14**	**87.12**	**86.63**

Table 6: Aspect Term Extraction Results on the 14Res, 14Lap, 15Res, and 16Res datasets.

Rating Prompt issue example

Input sentence: The phone's battery is amazing, but the camera is horrible. The price is a bit too high, but the brand's reliability is top-notch.

Actual triplets: (battery, amazing, positive), (camera, horrible, negative), (brand's reliability, top-notch, positive)

Triplets & Ratings extracted by GPT-4o:
(battery, amazing, positive) - 4,
(camera, horrible, negative) - 9,
(price, top-notch, negative) - 6,
(brand's reliability, top-notch, positive) - 4
(camera, high, positive) - 7

A.2.5 Question Answering prompt Output example

The prompt randomly gave incorrect answers for some of the triplets, the prompt lacked transparency and reasoning behind its decision. This is an example showing this issue:

Question Answering Prompt issue example

Input sentence: The phone's battery is amazing, but the camera is horrible. The price is a bit too high, but the brand's reliability is top-notch.

Actual triplets: (battery, amazing, positive), (camera, horrible, negative), (brand's reliability, top-notch, positive)

Triplets & Ratings extracted by GPT-4o:
(battery, amazing, positive) - Yes,
(camera, horrible, negative) - Yes,
(price, top-notch, negative) - Yes,
(brand's reliability, top-notch, positive) - No

A.2.6 Error Example PASTEL

The PASTEL approach sometimes struggles with identifying proper nouns as aspects, Here's an example highlighting this error:

PASTEL Error example

Input sentence: Servers are all different, Greg is my favorite.

Actual triplets: (Greg, favorite, positive)

Output:
No Valid triplets

A.3 Prompts

A.3.1 Opinion Extraction

The prompt for extracting opinion terms consists of instructions that the model must adhere to. This prompt is prefixed to the input sentence. Finally, the model is given an output format to follow, which makes processing of results easier for the subsequent steps in the pipeline. The main point in the instructions is that, the model is instructed to identify terms that correspond to a certain polarity. A prompt is created for each polarity value (positive, negative and neural) for each sentence. This polarity aware method of prompting helps the model to identify associated opinion terms by narrowing down the possible opinion terms.

Opinion Extraction Prompt

Prompt: *Task: Extract all opinion phrases contributing to a <**POLARITY**> sentiment from the given sentence.*
Instructions:

1. Identify all the phrases responsible for the <***POLARITY***> sentiment in the sentence.
2. Do not explain the output.
3. Provide only the extracted phrases in the output.

Input: <*SENTENCE*>

Output: <$OPINION_1$, POLARITY>, ...

A.3.2 Aspect Extraction

Similar to the previous prompt, the prompt used for aspect extraction task also includes some instructions, the input sentence and the output format. In addition to that, some contrastive examples are also provided to guide the model. The examples are contrastive because one example is associated with positive sentiment, one with negative and one with neutral. This helps the model in better disambiguation and also avoids bias towards frequent aspect terms that appear with a certain sentiment.

Aspect Extraction Prompt

Prompt:
Definition: Given a sentence, you must extract the explicit aspects which have an associated opinion. In cases where there are no aspects, the output should be **none**.
Positive Examples:
1. **Input:** I charge it at night and skip taking the cord with me because of the good battery life.
Output: battery life
2. **Input:** I even got my teenage son one, because of the features that it offers, like, iChat, Photobooth, garage band and more!.
Output: features, iChat, Photobooth, garage band

Negative Examples:
1. **Input:** Speaking of the browser, it too has problems.
Output: browser
2. **Input:** The keyboard is too slick.
Output: keyboard

Neutral Examples:
1. **Input:** I took it back for an Asus and same thing- blue screen which required me to remove the battery to reset.
Output: battery
2. **Input:** Nightly my computer defrags itself and runs a virus scan.
Output: virus scan
Now complete the following example:
Input: *"""<SENTENCE>"""*

Output: <ASPECT_1>, ...

A.3.3 LLM as Judge

The prompt for this section consists of the input sentence and triplets in the form of the sentence. The triplets are provided in the form of sentences to utilize LLM's context understanding abilities. The LLM is instructed to attach yes to each triplet sentence that is valid and no otherwise. Some few shot examples are also provided to help the LLM understand the input and output.

Prompt Example

prompt structure = """ Task: Given an input sentence and a list of queries, your task is to determine whether a given query can be entailed from the input sentence, ensuring that there is enough evidence to support the entailment, Here's an example:

<Few shot example>

Input Sentence: <Input Sentence>

Dataset		**# of Target with One Opinion Span**	**# of Target with Multiple Opinion Spans**	**# of Opinion with One Target Span**	**# of Opinion with Multiple Target Spans**
14Res	Train	1809	242	1893	193
	Dev	433	67	444	59
	Test	720	128	767	87
14Lap	Train	1121	160	1114	154
	Dev	252	44	270	34
	Test	396	67	420	54
15Res	Train	734	128	893	48
	Dev	180	33	224	12
	Test	385	47	438	23
16Res	Train	1029	169	1240	67
	Dev	258	38	304	15
	Test	396	56	452	23

Table 7: Statistics of 4 datasets.

Dataset	14Res				14Lap				15Res				16Res			
	#S	# +	# 0	# -	#S	# +	# 0	# -	#S	# +	# 0	# -	#S	# +	# 0	# -
Train	1266	1692	166	480	906	817	126	517	605	783	25	205	857	1015	50	329
Dev	310	404	54	119	219	169	36	141	148	185	11	53	210	252	11	76
Test	492	773	66	155	328	364	63	116	322	317	25	143	326	407	29	78

Table 8: Dataset statistics for 4 datasets (#S - number of sentences, #+, #0 and #- number of positive, neutral and negative triplets respectively.)

Query: Verify if the sentence expresses a “<Sentiment1>” sentiment toward “<Aspect1>” solely based on the use of the opinion term “<Opinion1>“. Reasoning:

Query: Verify if the sentence expresses a “<Sentiment2>” sentiment toward “<Aspect2>” solely based on the use of the opinion term “<Opinion2>“. Reasoning:

. . . .

Query: Verify if the sentence expresses a “<SentimentN>” sentiment toward “<AspectN>” solely based on the use of the opinion term “<OpinionN>“. Reasoning:

Output must be in this format for each query: Query: Reasoning: """

Few shot examples :
Input Sentence: “The Alfredo sauce was decent , the salad was okay , and the dessert options were quite limited .”

Query: Verify if the sentence expresses a “positive” sentiment toward “Alfredo sauce” solely based on the use of the opinion “decent“. Reasoning: The aspect Alfredo sauce is directly described by decent, which conveys a positive sentiment. Since the opinion correctly modifies the aspect with the right polarity, this triplet is valid. Thus, the output is ‘yes’.

Query: Verify if the sentence expresses a “positive” sentiment toward “salad” solely based on the use of the opinion term “decent“. Reasoning: The salad is described as okay, not decent, which exaggerates the sentiment. The mismatch in opinion and aspect makes this query invalid. Thus, the output is ‘no’.

Query: Verify if the sentence expresses a “negative” sentiment toward “dessert options” solely based on the use of the opinion term “limited“. Reasoning: The aspect dessert options is described by limited, which suggests a negative sentiment due to lack of variety. The opinion correctly aligns with the aspect, so this query is valid. Thus, the output is ‘yes’.

Query: Verify if the sentence expresses a “negative” sentiment toward “Alfredo sauce” solely based on the use of the opinion term “limited“. Reasoning: The opinion limited is incorrectly assigned to Alfredo sauce, which was actually described as decent. Since the opinion-aspect match is wrong, this query is invalid. Thus, the output is ‘no’.

Query: Verify if the sentence expresses a “neutral” sentiment toward “dessert options” solely based on the use of the opinion term “okay“. Reasoning: The opinion okay does not describe dessert options; instead, limited does, which carries a negative sentiment. Since the polarity is misclassified, this query is invalid. Thus, the output is ‘no’.

Query: Verify if the sentence expresses a “neutral” sentiment toward “salad” solely based on the use of the opinion term “okay“. Reasoning: The aspect salad is described as okay, which implies a neutral sentiment. The polarity is correctly assigned as neutral, making this query valid. Thus, the output is ‘yes’.

A.4 Dataset Statistics

The Table [7] presents the dataset statistics as provided by (Xu et al., 2020), it includes the statistics of the number of targets with one opinion span and the number of targets with multiple opinion spans, and also shows the number of opinions corresponding with single or multiple target spans respectively.

The statistics related to the number of opinion terms for each polarity for *14Rest*, *14Lap*, *15Rest* and *16Rest* an be found in Table [8], also provided by (Xu et al., 2020).