Make Compound Sentences Simple to Analyze: Learning to Split Sentences for Aspect-based Sentiment Analysis

Yongsik Seo*, **Sungwon Song***, **Ryang Heo***, **Jieyong Kim**, **Dongha Lee**†
Department of Artificial Intelligence, Yonsei University
{ndata,sungwonok,ryang1119,jieyong99,donalee}@yonsei.ac.kr

Abstract

In the domain of Aspect-Based Sentiment Analysis (ABSA), generative methods have shown promising results and achieved substantial advancements. However, despite these advancements, the tasks of extracting sentiment quadruplets, which capture the nuanced sentiment expressions within a sentence, remain significant challenges. In particular, compound sentences can potentially contain multiple quadruplets, making the extraction task increasingly difficult as sentence complexity grows. To address this issue, we are focusing on simplifying sentence structures to facilitate the easier recognition of these elements and crafting a model that integrates seamlessly with various ABSA tasks. In this paper, we propose **A**spect **T**erm **O**riented **S**entence **S**plitter (ATOSS), which simplifies compound sentence into simpler and clearer forms, thereby clarifying their structure and intent. As a *plug-and-play* module, this approach retains the parameters of the ABSA model while making it easier to identify essential intent within input sentences. Extensive experimental results show that utilizing ATOSS outperforms existing methods in both ASQP and ACOS tasks, which are the primary tasks for extracting sentiment quadruplets.[1]

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) (Pontiki et al., 2014) refers to the crucial task of understanding sentiments at the aspect-level. This technique identifies specific aspects of entities and evaluates their associated sentiments, providing richer contextual insights. In recent years, ABSA has progressed beyond simple sentiment classification to tackle more complex structures like sentiment triplets and quadruplets, which include aspect term, aspect category, opinion term and sentiment polarity for quadruplets. Among these, the Aspect Sentiment Quad Prediction (ASQP) and Aspect-Category-Opinion-Sentiment (ACOS) tasks, which involve predicting comprehensive sentiment quadruplets from a given sentence, are currently the most challenging tasks in ABSA and are being actively researched. Recently, generative methods have been proposed as solutions for predicting quadruplets, gaining significant attention in research due to their simplicity in addressing this problem in an end-to-end manner. (Zhang et al., 2021b; Hu et al., 2022; Gou et al., 2023) propose a framework that uses a sequence-to-sequence learning to transform an input sentence into predetermined output formats for predicting quadruplets.

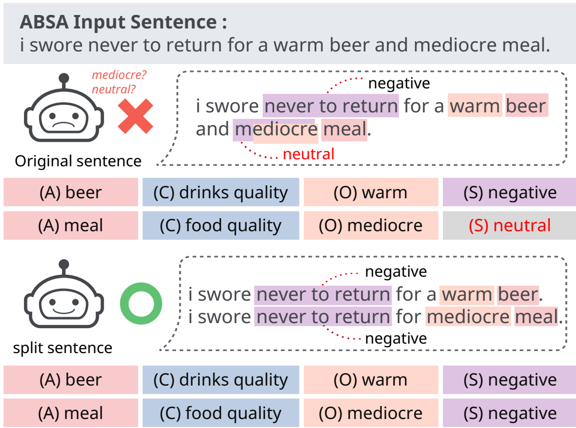

Figure 1: Existing ABSA models struggle to accurately predict quadruplets in sentence with compound syntactic structures but perform well when the sentences are provided in simpler and clearer forms.

Despite the state-of-the-art performances, ABSA models still suffer from the ambiguity of aspect-level sentiments in complex sentence structure, often due to multiple subjects with different states or context-dependent changes in a single subject. For example, in Figure 1 upper, the sentence "*I swore never to return for a warm beer and mediocre meal*" conveys a negative sentiment with "*never to return*".

*Equal contribution
†Corresponding author
[1]Our code is available at https://github.com/ryang1119/ATOSS.git

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 11171–11184
November 12-16, 2024

However, the opinion word "*mediocre*" describing the "*meal*" adds confusion to the overall sentiment of the sentence. As depicted in Figure 1 lower, by splitting this sentence into "*I swore never to return for a warm beer*" and "*I swore never to return for a mediocre meal*", we can clearly identify the intended sentiment quadruplets. Additionally, sentences that involve lists, conjunctions, causal relationships, or storytelling elements are difficult to interpret clearly, as they require complex reasoning and judgment, similar to human thought processes. To address the aforementioned challenges, our research aims to enable ABSA models to more easily identify intent within simple and clear sentence structures, avoiding confusion in complex and intertwined sentence constructions.

Motivated by these observations, we propose a model, named **A**spect **T**erm **O**riented **S**entence **S**plitter (ATOSS), which helps to accurately identify quadruplets by simplifying an original, compound sentence into simpler and clearer forms for ABSA models. Moreover, ATOSS, as a *plug-and-play* module, can be integrated into each ABSA model keeping their parameters. In other words, once ATOSS is pre-trained, it can be immediately applied to any ABSA model without the need for additional training. Specifically, ATOSS is first optimized via LLM distillation, and then aligned with the target ABSA model's sentence preference (Figure 3). We first obtain split sentences by prompting with LLM and train our model to generate the split sentence given an original sentence. Moreover, we address any ambiguities or splitting biases by further tailoring our ATOSS for the target ABSA model that will perform quadruplet prediction. To this end, we adopt preference alignment (Rafailov et al., 2023) with sentence pairs of preferred-dispreferred splitting results. As a result, ATOSS is fine-tuned to be further enhance the target ABSA model's quadruplet prediction accuracy.

Our extensive experiments on main aspect quadruplet prediction tasks, including ASQP and ACOS demonstrate that ATOSS significantly enhances the prediction accuracy of state-of-the-art ABSA models, including fine-tuned models and prompt-based LLMs. Specifically, ATOSS effectively reduces the error rates associated with incorrectly predicting aspect terms by facilitating the identification of aspect terms in input sentences for each sentiment quadruplet. Furthermore, our ATOSS splitter can be seamlessly integrated into the inference stage for other ABSA tasks, highlighting its high level of generalizability.

Our contributions are summarized as follows:

- We propose ATOSS splitter which splits compound sentences into simpler and clearer forms, allowing ABSA models to easily identify intent within the sentence structure.
- ATOSS aligns with the sentence preference of target ABSA model and, as a *plug-and-play* module, can be seamlessly integrated into existing models to enhance performance without the need to update their parameters.
- Experiments show that integrating ATOSS improves the quad prediction accuracy of existing ABSA models while also enabling them to adapt well to other ABSA tasks.

2 Related Work

In this section, we review the existing literature on (1) state-of-the-art approaches to Aspect-Based Sentiment Analysis (ABSA), and (2) recent efforts to distillation of large language models' (LLMs) remarkable reasoning ability on various tasks into small language models (LMs).

2.1 Sentiment Quadruplet Prediction

ABSA has been the focus of extensive research in recent years, aiming to extract sentiment-related elements for more fine-grained sentiment analysis. Earlier work primarily focused on predicting single or dual sentiment elements (Ma et al., 2019; Zhang and Qian, 2020). As the field evolved, more challenging ABSA tasks were proposed, such as Aspect Sentiment Triplet Extraction (ASTE) (Peng et al., 2020) and Target Aspect Sentiment Detection (TASD) (Wan et al., 2020), which focus on predicting sentiment triplet, and Aspect Category Opinion Sentiment (ACOS) (Cai et al., 2021) and Aspect Sentiment Quad Prediction (ASQP) (Zhang et al., 2021b), which target sentiment quadruplet.

To tackle sentiment quad prediction problems, early approaches proposed pipeline methods (Cai et al., 2021), but more recently, generative methods have emerged as the primary research focus because of their simplicity and end-to-end approach. Zhang et al. (2021b) solve the ASQP task by transforming target quads into natural language sentences, using the knowledge from the pre-trained generative model, which leads to better performance. Hu et al. (2022) was the first to investigate element ordering based on the quad structure, and proposed a method for predicting quads by augmenting the targets of the ASQP dataset with various permutations. Building on this, Gou et al. (2023) introduces an element order-based prompt learning method that improves sentiment tuple prediction by aggregating multi-view results.

Despite the promising results, long and complex text poses significant challenges for the model in predicting quadruplet. In this paper, we focus on a strategy for providing simpler sentences that allow ABSA models to handle complexity of sentences more accurately and effectively.

2.2 Distillation of LLM's Reasoning Ability

Knowledge distillation, which trains smaller models based on larger models, aims to reduce their size and latency while maintaining accuracy and generalization capabilities (Hinton et al., 2015; Sanh et al., 2019). Large Language Models (LLMs) have demonstrated an emergent ability in reasoning by generating explanations through Chain-of-Thought (CoT) prompting (Wei et al., 2022; Wang et al., 2023; Kojima et al., 2022). With the remarkable performance of LLMs across a wide range of tasks, recent research has focused on distilling their reasoning capabilities into smaller language models. Fine-tune-CoT (Ho et al., 2022) has demonstrated outstanding performance across various tasks by enabling LLMs to generate diverse reasoning paths and distill them into LMs. Some studies have conducted more detailed, task-specific knowledge distillation using LLMs (Magister et al., 2022; Chae et al., 2023; Hsieh et al., 2023). Recently, there has been an attempt to leverage LLM's CoT reasoning to address imprecise predictions and limited interpretability in the ASQP task (Kim et al., 2024).

3 Preliminaries

In this section, we formally define our target ABSA tasks and analyze ABSA models' behavior based on the structural complexity of input sentences.

3.1 Problem Formulation

In this work, we focus on tasks of predicting aspect-level sentiment from input sentence, in the form of structured quadruplets, (i.e., ASQP and ACOS). Formally, given an input sentence, these task aim to predict all aspect sentiment quadruplets consisting of four components, i.e., {(`at`, `ac`, `ot`, `sp`)}. The aspect term "`at`" and opinion term "`ot`" are detected within the sentence, while the aspect category "`ac`" and sentiment polarity "`sp`" are classified within their respective pre-defined sets.

Task	Datasets	Ratio of S / C	Acc of S	Acc of C
ASQP	Rest15	32.93 / 67.07	53.57	40.08
	Rest16	32.63 / 67.37	57.87	50.21
ACOS	Laptop16	32.72 / 67.28	39.28	30.23
	Rest16	32.16 / 67.84	54.45	48.25

Table 1: Proportion of *simple* (S) / *compound* (C) inputs, and quad prediction accuracy (i.e., Recall) of existing ABSA models for *simple* and *compound* sentences.

If the target aspect term is not explicitly mentioned, it is implicitly expressed and mapped to *"NULL"*. Note that ACOS differs from ASQP in its definition of the opinion term, focusing on more implicit aspects and opinions, which may result in the opinion term being *"NULL"* as well. The aspect category is classified as an element within the category set that is pre-defined for each domain or dataset; for example, in restaurant review datasets, it includes various categories such as "*food prices*" and "*ambience general*". The sentiment polarity is predicted as one of the three sentiment classes: "*positive*", "*neutral*", and "*negative*", each indicating the corresponding aspect-level sentiment.

3.2 Analysis of ABSA Performance based on Sentence Structural Complexity

We first investigate the prediction accuracy of existing ABSA models according to the degree of complexity in an input sentence structure. To this end, we categorize all test inputs into two sets: *simple sentence* and *compound sentence*. In this work, we define *simple sentence* as a sentence containing only a single independent clause and annotated with a single aspect quadruplet. In contrast, *compound sentence* is connected by conjunctions such as "*and*", "*or*", "*but*", and punctuated with commas; it can be annotated with one or more aspect quadruplets. Each sentence in the ASQP and ACOS datasets may contain several quadruplets for a single aspect or multiple aspects.

Table 1 reports the error rates of state-of-the-art ABSA models for both *simple* and *compound sentence* inputs. From the results, we observe a high error rate in *compound sentence* across all tasks. As illustrated in Figure 1, when multiple quadruplets exist for multiple aspect terms in a sentence, it

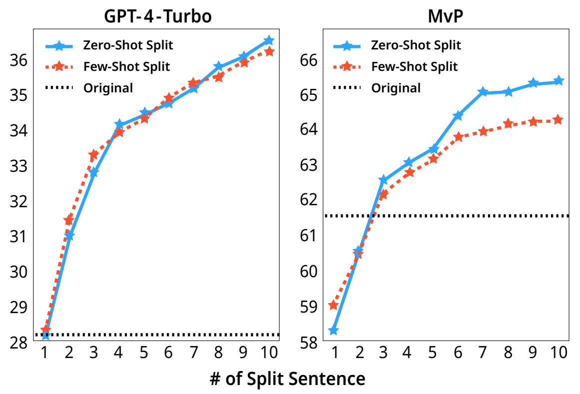

Figure 2: Performance changes of ABSA models (Left: GPT-4-turbo, Right: MvP) w.r.t. the number of candidate split sentences. (Task: ACOS, Dataset: Rest16)

becomes more complex and intricately intertwined. In such cases, the model struggles to detect each term accurately, especially when the distance between the aspect term and sentiment component in text is relatively large, making accurate quadruplet prediction difficult. This issue arises often when multiple aspects each have multiple associated opinions or sentiment quadruplets within the same sentence. Based on these observations, we hypothesize that splitting a *compound sentence* into simple ones based on aspect terms could reduce the error rate and lead to higher overall accuracy. We provide more detailed analysis in Section 5.2.

In addition, we conduct a preliminary study to validate our hypothesis that simplifying sentences via splitting can enhance their clarity and thereby improve the quadruplet prediction accuracy of ABSA models. To this end, we examine how much F1 score in quadruplet prediction improves when using split sentences instead of the original ones. Specifically, for each test input, we generate 10 candidate sentence splits by prompting LLM to perform sentence splitting in both zero-shot and few-shot manners. We then select the best split sentence that yields the highest F1 score (i.e., *oracle* voting). Figure 2 shows the changes in prediction F1 score as the number of candidate sentence splits increases. Both the fine-tuned model (i.e., MvP) and the prompting-based LLM (i.e., GPT-4-turbo) demonstrate significant performance improvements with split sentences, especially when more candidates of split sentences are available. Notably, split sentences obtained through zero-shot prompting, which result in more diverse split forms compared to few-shot prompts, show great potential for enhancing the performance of existing ABSA models. Consequently, we confirm that the strategy of splitting sentences into appropriate simple forms can indeed aid in enhancing ABSA performance.

4 Methodology

In this section, we present a *plug-and-play* module that splits an input sentence into multiple simple sentences that facilitates ABSA tasks. Our proposed model, named **A**spect **T**erm **O**riented **S**entence **S**plitter (ATOSS), is a small LM trained via knowledge distillation from a teacher LLM and further refined to align its output with helpfulness for enhancing the target ABSA model's accuracy. The overview of ATOSS is illustrated in Figure 3.

4.1 Aspect-Oriented Splitting Strategy

Our research aims to enable ABSA models to easily find intent within simple and clear sentence structures. To achieve this goal, we employ *Aspect-Oriented Splitting Strategy*, which ensures that the split sentences contain aspect terms. As mentioned in Section 3.2, existing datasets consist of both *simple* and *compound* sentences for ABSA models to process. For the *simple sentence*, which has a straightforward structure and allows the model to accurately understand its intent without additional steps, we retain it without modification. However, the *compound sentence* often have a complex structure, such as multiple quadruplets oriented from a single aspect term, making it challenging for existing models to discern the intent within sentence. To address this issue, we only split *compound sentence* employing the *Aspect-Oriented Splitting* strategy.

4.2 Distillation of LLM's Splitting Ability

We first optimize our sentence splitter to generate diverse split sentences given an original input sentence. To train the sentence splitter, we augment the ABSA training dataset of sentence-quadruplet pairs $(s, \mathcal{Q})$ into $\mathcal{X} = \{(s, s', \mathcal{Q})\}$, where s' is the split sentence from the original sentence s.

Split sentence generation To distill a teacher LLM's sentence splitting ability into ATOSS, we generate training data for distillation by prompting the LLM. In this step, we simply adopt zero-shot prompting for split sentence generation, allowing the LLM to diversify split s into s' using its pre-trained knowledge. To explore effective and diverse s', we instruct the LLM to generate 10 diverse s' for each s with given matching spellings.

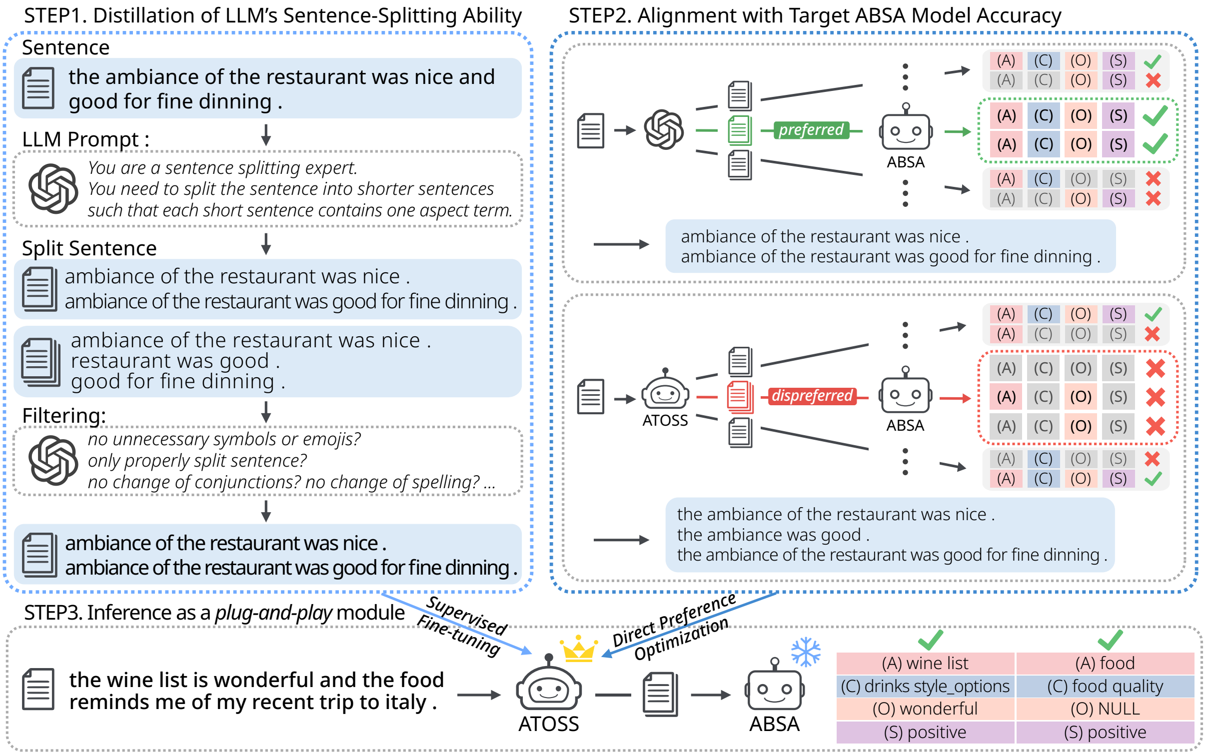

Figure 3: Overall framework for training and utilizing ATOSS for ABSA tasks. The training process involves (1) distillation of LLM's capability for sentence splitting, and (2) alignment with a target model's sentence preference. The inference process (3) predicts the quadruplets by taking sentences split by ATOSS as the input. ATOSS, as a *plug-and-play* module, can enhance prediction accuracy without requiring updates to the target model's parameters.

Split sentence selection However, as s' may still contain noise, we introduce an additional filtering process to select K split sentences that best meet the specific criteria, within the set of generated s'. This filtering process is also performed via LLM prompting with manually written splitting criteria.

Supervised fine-tuning (SFT) We train the model to predict the target s' given an input s. With the input-target pair (s, s'), we fine-tune the sequence-to-sequence LM by minimizing the following negative log-likelihood loss:

$$\mathcal{L}_{NLL} = -\log p(s'|s) = -\sum_{t=1}^{T} \log p(s'_t|s, s'_{<t}),$$

where T is the length of the target sequence s and $s'_{<t}$ denotes previously generated tokens. Note that the purpose of this step is to obtain a general sentence splitter by distilling an LLM's sentence splitting ability; therefore, we use all (s, s') samples collected from multiple available datasets.

4.3 Alignment with Sentence Preference

Even though the general splitter is able to split the sentence to some extent, but the general split sentence may not always be an optimally processed input for a specific model. To further refine the splitter for a target dataset, task, and ABSA model, we additionally tune ATOSS based on preference alignment strategy. While shorter sentences are generally easier for tasks than longer ones, the optimal format for split sentences may vary depending on the models and datasets. To address this issue, we apply Direct preference optimization (DPO) (Rafailov et al., 2023) to the ATOSS for each specific model and dataset to perform the tasks.

Preferred sentence selection We obtain s' that is optimal based on our manual splitting strategy (Section 3.2), but this may exclude other optimal s' formats that we do not consider. To mitigate this issue, we utilize few-shot prompting for split sentence generation through LLM, allowing it to effectively split s into s' via in-context learning of given examples. By providing LLM with $(s, \mathcal{Q})$ pair, we guide it to split s based on its ground-truth aspect terms of $\mathcal{Q}$. To explore effective and appropriate s', we prompt LLM to generate 10 diverse s' for each s, as input to measure the sentence-level F1 score from the ABSA model's inference stage.

We use this comparison to select the preferred sentences: if s is *simple sentence*, we select s' where the number of split sentences matches the number of quadruplets; for *compound sentence*, if s has the higher F1 score, no preferred sentence is chosen; if the scores are equal, s is retained; if s has the lower score, we select all distinct s' with higher score.

Dispreferred sentence selection Even if we construct the appropriate dataset, ATOSS may fail to generate optimal s' for ABSA model. To alleviate this issue, we utilize beam search feature of ATOSS to generate 10 different s'. We use this comparison to select the dispreferred sentences: if s is *simple sentence*, we select s' with the hightest similarity; for *compound sentence*, if s has the lower F1 score, no dispreferred sentence is chosen; if the scores are equal, we select s' with the lowest similarity to s; if s has the higher score, we select s' with the highest similarity to s among those with lower score.

Direct preference optimization Through the aforementioned process, we can construct preference pairs $P = \{(s, p^+, p^-)\}$ using the preferred sentence p^+ and the dispreferred sentence p^-. We apply DPO on our sentence splitter θ to train a preference-tuned sentence splitter θ^* that minimizes the following objective:

$$L_{\mathrm{DPO}}(\theta^*; \theta) = \\ -\mathbb{E}_{(s,p^+,p^-)\sim P} \log \sigma \left[r(s, p^+) - r(s, p^-) \right],$$

where $r(s, p) = \frac{p_{\theta^*}(p|s)}{p_\theta(p|s)}$. By optimizing the model using preferred-dispreferred sentence pairs, our obtained model θ^* is trained to prefer sentences that are clearly split based on the aspect term while avoiding those that are ambiguously or unclearly split. Note that θ has been specifically trained for target ABSA model, since the preferred split sentences vary across different models.

4.4 Applying ATOSS as a *plug-and-play*

During inference, our final ATOSS model, obtained via a two-step optimization process, transforms an input sentence into a split one, which is then provided to the ABSA model. This stage adopts ATOSS tailored for a specific ABSA model and dataset. By utilizing ATOSS as a *plug-and-play* module, existing ABSA models can process input sentences optimized for each task without updating their parameters, thereby improving performance. In other words, as ABSA models themselves are not required any tuning, ATOSS can be universally applied to any off-the-shelf ABSA models, or to any ABSA tasks focusing on aspect-level sentiments. Note that it can also be adopted to closed-source LLMs, such as GPT-3.5-turbo, GPT-4-turbo and GPT-4o which cannot be tuned, demonstrating outstanding applicability and flexibility.

5 Experiments

We conducted our experiments while addressing the following research questions:

- **RQ1:** Can ATOSS enhance the quad prediction accuracy of existing ABSA models?
- **RQ2:** Can ATOSS improve aspect-level F1 in quad prediction for existing ABSA models?
- **RQ3:** Can ATOSS be effective for other ABSA tasks beyond quad prediction?

5.1 Experimental Settings

Tasks and datasets We validate the effectiveness of ATOSS on 4 datasets across 2 tasks, ASQP and ACOS. For ASQP, we utilize two restaurant domain datasets, i.e., Rest15 and Rest16 (Pontiki et al., 2015, 2016). In the case of ACOS, we adopt restaurant-ACOS and laptop-ACOS datasets, i.e., Rest16 and Laptop16 (Cai et al., 2021). Refer to Appendix C for more details.

Implementation details We employ the T5-base model (Raffel et al., 2020) from Huggingface Transformers[2] (Wolf et al., 2020) as the backbone model of our splitter. We adopt the *plug-and-play* module in our experiments, maintaining the existing parameters of target ABSA model while only tuning the parameters of the ATOSS model. To filter out noisy sentences and select those that best meet the criteria, we set *K*=2. We use two variants of our model: the splitter trained by using only LLM Distillation, named *General* (Section 4.2), and the one aligned for preference, named *Specific* (Section 4.3). More details about implementation for our model are given in Appendix A.2

Evaluation metrics For all tasks, a sentiment quadruplet is considered correct if and only if every element matches gold quadruplet exactly. We utilize F1 score as primary evaluation metric (Zhang et al., 2021a; Mao et al., 2022), with all reported F1 score for fine-tuned models (i.e. MvP) using randomly selected seeds. We use same F1 score

[2]https://github.com/huggingface/transformers

Methods	ASQP						ACOS					
	Rest15			Rest16			Laptop16			Rest16		
	Pre	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**
Fine-tuned models												
Paraphrase (Zhang et al., 2021b)	43.70	47.55	45.54	56.28	59.45	57.82	43.23	42.89	43.06	58.74	60.55	59.63
+ ATOSS (*General*)	45.44	47.04	46.23	57.28	59.57	58.40	43.49	43.15	43.32	57.84	59.12	58.47
+ ATOSS (*Specific*)	46.06	47.80	**46.91**	58.05	60.45	**59.23**	43.79	43.76	**43.77**	59.01	62.21	**60.57**
ILO (Hu et al., 2022)	48.51	49.06	48.78	55.98	60.95	58.36	45.10	45.56	45.33	56.32	57.51	56.91
+ ATOSS (*General*)	50.45	49.31	**49.87**	57.62	61.08	59.30	44.51	44.70	45.55	57.05	57.33	57.33
+ ATOSS (*Specific*)	49.42	48.43	48.92	57.96	61.08	**59.48**	46.19	46.43	**46.31**	58.20	58.40	**58.30**
DLO (Hu et al., 2022)	46.88	49.18	48.00	57.28	61.08	59.12	43.65	43.84	43.75	59.30	59.96	59.62
+ ATOSS (*General*)	47.76	49.56	**48.64**	58.60	60.33	59.40	43.92	43.58	43.99	59.87	60.07	59.97
+ ATOSS (*Specific*)	48.01	48.68	48.34	61.50	58.57	**60.15**	45.13	43.93	**44.52**	61.18	59.96	**60.56**
MvP (Gou et al., 2023)	49.81	48.68	49.24	61.33	62.33	61.82	43.76	43.69	43.72	61.27	57.86	59.52
+ ATOSS (*General*)	51.99	49.18	50.55	61.61	61.45	61.53	45.21	42.91	44.03	61.05	56.99	58.95
+ ATOSS (*Specific*)	51.99	49.18	**50.55**	62.80	61.70	**62.25**	45.32	43.17	**44.22**	63.12	58.30	**60.61**
Prompting-based LLMs												
GPT-3.5-turbo	15.38	16.73	16.02	21.23	23.28	22.21	7.27	7.92	7.58	21.46	23.80	22.57
+ ATOSS (*General*)	20.15	23.14	**21.55**	26.13	29.66	27.78	8.53	9.39	**8.94**	23.13	26.31	24.62
+ ATOSS (*Specific*)	20.07	22.89	21.39	27.42	30.79	**29.01**	8.33	9.22	8.75	23.40	26.42	**24.82**
GPT-4-turbo	20.11	25.96	22.66	23.76	28.16	25.77	9.24	10.51	9.83	28.44	30.68	28.99
+ ATOSS (*General*)	21.11	25.91	23.26	23.52	28.79	25.89	9.31	10.77	**9.98**	28.18	31.66	29.82
+ ATOSS (*Specific*)	21.59	26.67	**23.86**	25.73	30.91	**28.08**	9.16	10.85	9.94	28.68	31.88	**30.20**
GPT-4o	18.30	20.88	19.51	24.05	27.03	25.46	10.48	11.28	10.87	21.63	21.72	21.68
+ ATOSS (*General*)	25.88	29.43	27.55	32.05	35.54	33.71	11.44	12.83	12.10	27.63	28.71	28.16
+ ATOSS (*Specific*)	26.25	30.31	**28.14**	33.22	36.92	**34.97**	11.83	13.35	**12.55**	28.14	29.37	**28.74**

Table 2: Performance (%) of various ABSA models and the ones equipped with ATOSS.

metric to evaluate in a single run for prompting-based models (i.e. GPT-4-turbo).We also report precision (Pre) and recall (Rec) scores.

ABSA models For ABSA models, we use fine-tuned models that have recently shown outstanding performance, i.e., **Paraphrase** (Zhang et al., 2021b), **ILO** & **DLO** (Hu et al., 2022) and **MvP** (Gou et al., 2023). We also use the prompting-based LLMs, i.e., **GPT-3.5-turbo** (`gpt-3.5-turbo-0125`), **GPT-4-turbo** (`gpt-4-1106-preview`) and **GPT-4o** (`gpt-4o`) [3], employing zero-shot prompting.

5.2 Effectiveness of ATOSS (RQ1 & RQ2)

Performance comparison Table 2 presents ASQP and ACOS performance of various models. Overall, ABSA models integrating our ATOSS, which takes split sentences as inputs, lead to better performance compared to those that use original sentences as inputs. This improvement is consistently observed across fine-tuned models and prompting-based LLM; this highlights efficacy of our sentence splitting approach in enhancing the accuracy of existing ABSA models without parameter tuning and extensive modifications. Both the *General* and *Specific* versions of ATOSS yield improved results, demonstrating its consistent ability to generate clear split sentences for ABSA models.

Methods	ASQP		ACOS	
	Rest15	**Rest16**	**Laptop16**	**Rest16**
GPT-3.5-turbo	16.02	22.21	7.58	22.57
+ ATOSS (*General*)	**21.55**	27.78	**8.94**	24.62
+ ATOSS (*Specific*)	21.39	**29.01**	8.75	**24.82**
w / CoT Splitting	17.84	23.70	8.89	22.48
GPT-4o	19.51	25.46	10.87	21.68
+ ATOSS (*General*)	27.55	33.71	12.10	28.16
+ ATOSS (*Specific*)	**28.14**	**34.97**	**12.55**	**28.74**
w / CoT Splitting	21.96	27.08	12.07	24.87

Table 3: Comparative performance (%) of LLMs equipped with ATOSS and those utilizing split-then-quadruplet prediction via zero-shot CoT prompting.

LLM's capability in quad prediction: a comparative analysis with ATOSS We conduct experiments that leverage zero-shot CoT prompting (Kojima et al., 2022) to make the LLM analyze the structure of sentences, split them, and then predict quads from the sentences, without using our

[3] `https://chat.openai.com/`

Methods	ASQP						ACOS					
	Rest15			Rest16			Laptop16			Rest16		
	S	C	T	S	C	T	S	C	T	S	C	T
Paraphrase	86.21	68.30	**71.98**	90.91	74.89	78.27	89.39	73.10	76.80	91.92	76.01	79.00
+ ATOSS (*Specific*)	86.44	67.61	71.55	91.01	75.25	**78.61**	89.38	74.40	**77.90**	91.60	76.88	**79.79**
MvP	80.33	63.24	67.17	91.88	73.36	77.44	92.45	80.52	**83.30**	85.86	75.14	77.43
+ ATOSS (*Specific*)	84.92	70.31	**73.64**	90.20	78.15	**80.86**	89.86	75.29	78.85	91.10	78.99	**81.61**
GPT-4o	42.37	51.48	49.59	45.51	57.42	54.92	69.20	51.67	55.50	47.83	60.49	57.95
+ ATOSS (*Specific*)	79.66	61.61	**65.34**	86.52	68.37	**72.20**	80.30	63.92	**67.42**	84.47	67.38	**70.73**

Table 4: Performance (%) of ABSA models and one equipped with ATOSS in terms of aspect-level F1. categorized by sentence type: *simple* (S), *compound* (C), and *total* (T).

Methods	TASD		ASTE	
	Rest15	Rest16	Rest15	Rest16
MvP	63.46	71.23	63.29	73.09
+ ATOSS (*General*)	63.32	70.26	64.66	73.34
+ ATOSS (*Specific*)	**64.02**	**71.33**	**65.60**	**74.06**
GPT-3.5-turbo	34.66	40.83	42.41	51.43
+ ATOSS (*General*)	**37.28**	**41.87**	42.14	51.60
+ ATOSS (*Specific*)	36.77	41.67	**42.90**	**53.52**
GPT-4o	50.43	53.80	49.04	54.82
+ ATOSS (*General*)	**51.46**	**55.80**	49.59	55.58
+ ATOSS (*Specific*)	50.93	55.79	**50.78**	**56.99**

Table 5: Performance (%) of various ABSA models equipped with ATOSS in the *cross-task* setting. ATOSS is trained for ASQP then tested for TASD and ASTE.

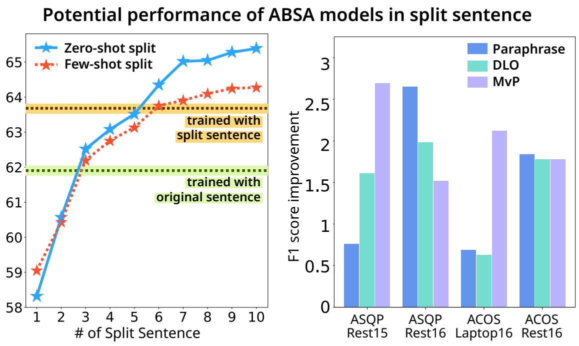

Figure 4: **Left**: Model: MvP, Task: ACOS, Dataset: Rest16. **Right**: F1 improvement when each model is trained on split sentences instead of original sentences.

ATOSS splitter. As shown in Table 3, applying ATOSS significantly improves quad prediction performance, whereas relying on the LLMs' reasoning ability without ATOSS results in a smaller improvement. Therefore, the effectiveness of ATOSS in predicting quads has been demonstrated.

Aspect-level performance analysis We observed in Figure 2 that splitting sentences into simpler forms enhances the detection of aspect terms. Based on this observation, we further investigate whether using ATOSS improves the performance of aspect term extraction in ABSA models. As indicated in Table 4, we can observe an overall improvement in aspect-level F1 scores not only for the *compound sentence* targeted by our ATOSS but also for the *total sentence*. In other words, these findings indicate that splitting sentences helps ABSA models more effectively recognize primary intents within the input sentence structure.

5.3 Generalizability of ATOSS in ABSA (RQ3)

We evaluate our approach in *cross-task* setting by utilizing ATOSS, trained for quadruplet prediction task, to preprocess the inputs of target ABSA model performing the triplet prediction task, TASD and ASTE. When evaluating performance, we integrate a ATOSS (*General*) trained on ASQP with a ATOSS (*Specific*) tailored to the restaurant dataset from each task. As shown in Table 5, the results consistently demonstrate performance improvements across all datasets, highlighting the effectiveness of sentence simplification through splitting in *cross-task* setting. These findings suggest that even when the elements to be predicted from the sentence differ across tasks, ATOSS remains effective, showcasing its versatility and utility in various scenarios.

5.4 Potential performance of ABSA models in split sentences

To assess the potential performance of the ABSA models in split sentences, we train the models using split sentences and then evaluate the inference results on split sentences. To this end, we collect a set of split sentences via GPT-3.5-turbo few-shot prompting so that it aligns well with our criteria for splitting sentence, and measure performance with this dataset. As illustrated in Figure 4 (**Left**), ABSA models on split sentences adapts better during the inference stage to these split sentences, approaching the *oracle voting* performance mentioned earlier in Figure 2. Additionally, in Figure 4 (**Right**), more significant performance improvements can be achieved by using split sentences as inputs for both training and inference in models, compared to using them only for inference.

6 Conclusion

In this paper, we aim to simplify sentences through strategic splitting, allowing ABSA models to better understand the inherent structure within input sentences. This splitting strategy, termed *Aspect-Oriented Splitting*, divides sentences more concisely and clearly while retaining the essential element of the aspect term. Based on these points, we demonstrate that ATOSS, as a *plug-and-play* module, can be seamlessly integrated with various fine-tuned models as well as prompt-based LLMs, improving performance without necessitating any updates to the models themselves. Our research highlights the significant impact of sentence structure on sentiment analysis, presenting substantial implications for the broader field of ABSA.

7 Limitations

Despite achieving state-of-the-art performance, our study faces several limitations. First, the available dataset is not large enough, limiting our training effectiveness. The scarcity of real data for tasks such as predicting sentiment quadruplet (i.e., ASQP and ACOS) required cross-training with unseen data, which might have yielded better results with a more extensive dataset. Secondly, the high cost associated with building our ATOSS presents a significant challenge. We use LLMs not only to generate split sentences but also for evaluation purposes. Especially, we use repeated prompting for each input sentence during the data regeneration process for sentence splitting. Lastly, ATOSS as a *plug-and-play* module and is therefore dependent on existing ABSA models. Due to this dependency, our model cannot be utilized independently and its performance may vary based on the quality and characteristics of the ABSA models it uses.

8 Ethical Statement

We employ datasets that are well-recognized and previously utilized within the scientific community, ensuring both transparency and integrity in our experiments. In other words, our methodologies and findings do not cause harm to any individuals or groups, we have publicly released our code as open-source. We are aware of the potential biases in sentiment polarity predictions that may arise from using large pre-trained language models, as these models can reflect societal biases present in their training data (Tan and Celis, 2019). We recognize the importance of ongoing efforts to address these biases. Moreover, we emphasize the necessity of continuous monitoring and rigorous evaluation to prevent our smaller downstream models from replicating or amplifying the biases of their larger language model counterparts.

Acknowledgements

This work was supported by the IITP grants funded by the Korea government (MSIT) (No. RS-2020-II201361; RS-2024-00457882, AI Research Hub Project), and the NRF grant funded by the Korea government (MSIT) (No. RS-2023-00244689).

References

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 and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 340–350.

Hyungjoo Chae, Yongho Song, Kai Tzu iunn Ong, Taeyoon Kwon, Minjin Kim, Youngjae Yu, Dongha Lee, Dongyeop Kang, and Jinyoung Yeo. 2023. Dialogue chain-of-thought distillation for commonsense-aware conversational agents. *ArXiv*, abs/2310.09343.

Zhibin Gou, Qingyan Guo, and Yujiu Yang. 2023. Mvp: Multi-view prompting improves aspect sentiment tuple prediction. *arXiv preprint arXiv:2305.12627*.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2022. Large language models are reasoning teachers. In *Annual Meeting of the Association for Computational Linguistics*.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander J. Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *ArXiv*, abs/2305.02301.

Mengting Hu, Yike Wu, Hang Gao, Yinhao Bai, and Shiwan Zhao. 2022. Improving aspect sentiment quad prediction via template-order data augmentation. *arXiv preprint arXiv:2210.10291*.

Jieyong Kim, Ryang Heo, Yongsik Seo, SeongKu Kang, Jinyoung Yeo, and Dongha Lee. 2024. Self-consistent reasoning-based aspect-sentiment quad prediction with extract-then-assign strategy. *arXiv preprint arXiv:2403.00354*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Dehong Ma, Sujian Li, Fangzhao Wu, Xing Xie, and Houfeng Wang. 2019. Exploring sequence-to-sequence learning in aspect term extraction. In *Annual Meeting of the Association for Computational Linguistics*.

Lucie Charlotte Magister, Jonathan Mallinson, Jakub Adamek, Eric Malmi, and Aliaksei Severyn. 2022. Teaching small language models to reason. *ArXiv*, abs/2212.08410.

Yue Mao, Yi Shen, Jingchao Yang, Xiaoying Zhu, and Longjun Cai. 2022. Seq2Path: Generating sentiment tuples as paths of a tree. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2215–2225, Dublin, Ireland. Association for Computational Linguistics.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8600–8607.

Maria Pontiki, Dimitrios Galanis, Harris 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.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *ProWorkshop on Semantic Evaluation (SemEval-2016)*, pages 19–30. Association for Computational Linguistics.

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

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Yi Chern Tan and Elisa Celis. 2019. Assessing social and intersectional biases in contextualized word representations. *ArXiv*, abs/1911.01485.

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 *AAAI Conference on Artificial Intelligence*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Mi Zhang and Tieyun Qian. 2020. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *Conference on Empirical Methods in Natural Language Processing*.

Wenxuan Zhang, Yang Deng, Xin Li, Lidong Bing, and Wai Lam. 2021a. Aspect-based sentiment analysis in question answering forums. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4582–4591, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. 2021b. Aspect sentiment quad prediction as paraphrase generation. *arXiv preprint arXiv:2110.00796*.

A Experiment Details

A.1 Software and Hardware

We use Pytorch to implement all the models (Python 3.8). Our all experiments are conducted with a single NVIDIA A6000 with 48GB of RAM.

A.2 Implementation Details

ATOSS *(General)* is trained by performing Supervised fine-tuning (SFT) on split sentences by LLM zero-shot prompting. The training batch size is set to 64, and the validation batch size is set to 8. Training is conducted for 50 epochs with the learning rate of 6e-5. Early stopping is implemented with a patience of 20 epochs. ATOSS *(Specific)* is trained on ATOSS *(General)* applying Direct preference optimization (DPO) (Rafailov et al., 2023) to reflect each ABSA model's preferred split sentences. Both the training and validation batch sizes are set to 8. Training is conducted for 1 epoch with the learning rate is set to 1e-4. The beta parameter is set at 0.1 and the loss function used is sigmoid.

B Prompts for Sentence Splitting

We use GPT-4-turbo (`gpt-4-1106-preview`) to generate split sentences for ATOSS. Table 6 shows the zero-shot prompt for ATOSS *(General)*, and Table 7 shows the few-shot prompt for ATOSS *(Specific)*.

C Dataset Statistics

Tables 8 presents the dataset statistics for all ASQP, ACOS, ASTE, and TASD task covered in our work.

Prompt: ATOSS (*General*)
[Task Description] You are a sentence splitting expert. You will be provided with a review sentence and a few [aspect, category, sentiment, opinion] quadruplets from that review sentence. Here is the definition of each element in the quadruplet: - The ‘aspect’ refers to a specific feature, attribute, or aspect of a product or service that a user may express an opinion about. The aspect term might be ‘null’ for an implicit aspect. - The ‘opinion’ refers to the sentiment or attitude expressed by a user towards a particular aspect or feature of a product or service. The opinion term might be ‘null’ for an implicit opinion. - The ‘category’ refers to the category that the aspect belongs to (e.g. food quality, restaurant general, etc.). - The ‘sentiment’ refers to the sentiment class of the aspect (e.g. positive, negative, neutral). You need to split the sentence into shorter sentences such that each short sentence contains one aspect term. When splitting, sentences connected by conjunctions must be divided into individual sentences along with their conjunctions. This process must specify the subject in every sentence. This process must retain the existing spellings exactly as in the original sentence. This process must also retain the existing spacings exactly as in the original sentence. If the sentence is too short to split or does not need to be split, use the original sentence as is. No numbering, line breaks, or explanations are needed.

Table 6: The zero-shot prompt for the distillation of LLM’s splitting ability on ABSA.

Prompt: ATOSS (*Specific*)

[Task Description]
You are a sentence splitting expert. You will be provided with a review sentence and a few [aspect, category, sentiment, opinion] quadruplets from that review sentence. Here is the definition of each element in the quadruplet:
- The 'aspect' refers to a specific feature, attribute, or aspect of a product or service that a user may express an opinion about. The aspect term might be 'null' for an implicit aspect.
- The 'opinion' refers to the sentiment or attitude expressed by a user towards a particular aspect or feature of a product or service. The opinion term might be 'null' for an implicit opinion.
- The 'category' refers to the category that the aspect belongs to (e.g. food quality, restaurant general, etc.).
- The 'sentiment' refers to the sentiment class of the aspect (e.g. positive, negative, neutral).

You need to split the sentence into shorter sentences such that each short sentence contains one aspect term. When splitting, sentences connected by conjunctions must be divided into individual sentences along with their conjunctions. This process must specify the subject in every sentence. This process must retain the existing spellings exactly as in the original sentence. This process must also retain the existing spacings exactly as in the original sentence. If the sentence is too short to split or does not need to be split, use the original sentence as is. No numbering, line breaks, or explanations are needed.

[Example 1]
Original sentence: i will be going back and heartily recommend it !

Quadruplets: [['null', 'restaurant general', 'positive', 'recommend']]

Split sentence: i will be going back and heartily recommend it !

[Example 2]
Original sentence: i ' ve never had bad service and the fish is fresh and delicious .

Quadruplets: [['service', 'service general', 'positive', 'never had bad'], ['fish', 'food quality', 'positive', 'fresh'], ['fish', 'food quality', 'positive', 'delicious']]

Split sentence: i ' ve never had bad service . and the fish is fresh and delicious .

[Example 3]
Original sentence: very immature bartender , didnt know how to make specific drinks , service was so slowwwww , the food was not fresh or warm , waitresses were busy flirting with men at the bar and werent very attentive to all the customers .

Quadruplets: [['bartender', 'service general', 'negative', 'immature'], ['service', 'service general', 'negative', 'slowwwww'], ['food', 'food quality', 'negative', 'not fresh or warm'], ['waitresses', 'service general', 'negative', 'werent very attentive']]

Split sentence: very immature bartender, didnt know how to make specific drinks. service was so slowwwww. the food was not fresh or warm. waitresses were busy flirting with men at the bar and werent very attentive to all the customers .

[Example 4] ...

Table 7: The few-shot prompt for aligning with sentence preference of each ABSA model (Examples of 4 to 10 are omitted in this table).

Task	Dataset (#C)	Train (POS/NEU/NEG)	Dev (POS/NEU/NEG)	Test (POS/NEU/NEG)
ASQP	Rest15 (#13)	834 1,005 / 34 / 315	209 252 / 14 / 81	537 453 / 37 / 305
	Rest16 (#13)	1,264 1,369 / 62 / 558	316 341 / 23 / 143	544 584 / 40 / 177
ACOS	Rest16 (#13)	1,530 1,656 / 95 / 733	171 180 / 12 / 69	583 668 / 44 / 205
	Laptop16 (#121)	2,934 2,583 / 227 / 1,364	326 279 / 24 / 137	816 716 / 65 / 380
ASTE	Rest14 (-)	1,266 1,692 / 166 / 480	310 404 / 54 / 119	492 773 / 66 / 155
	Laptop14 (-)	906 817 / 126 / 517	219 169 / 36 / 141	328 364 / 63 / 116
TASD	Rest15 (#13)	1,120 1,198 / 53 / 403	10 6 / 0/ 7	582 454 / 45 / 346
	Rest16 (#13)	1,708 1,657 / 101 / 749	29 23 / 1 / 297	587 611 / 44 / 204

Table 8: Dataset statistics for various tasks. #C denote the the number of aspect categories in the pre-defined set. POS, NEU, and NEG refer to the number of positive, neutral, and negative quadruplets or triplets respectively.