Aspect-Based Sentiment Analysis with Syntax-Opinion-Sentiment Reasoning Chain

Rui Fan[1,2,3,†], Shu Li[1,2,4,†], Tingting He[1,2,4,*], Yu Liu[1,2,3]
[1]Hubei Provincial Key Laboratory of Artificial Intelligence and Smart Learning,
[2]National Language Resources Monitor and Research Center for Network Media,
[3]Faculty of Artificial Intelligence in Education,
[4]School of Computer, Central China Normal University, Wuhan, China
{fanrui, lishu-keep, yuliu}@mails.ccnu.edu.cn, tthe@mail.ccnu.edu.cn

Abstract

Despite the impressive capabilities of large language models (LLMs) in aspect-based sentiment analysis (ABSA), the role of syntactic information remains underexplored in LLMs. Syntactic structures are known to be crucial for capturing aspect-opinion relationships. To explore whether LLMs can effectively leverage syntactic information to improve ABSA performance, we propose a novel multi-step reasoning framework, the Syntax-Opinion-Sentiment Reasoning Chain (Syn-Chain). Syn-Chain sequentially analyzes syntactic dependencies, extracts opinions, and classifies sentiment. We introduce Syn-Chain into LLMs via zero-shot prompting, and results show that Syn-Chain significantly enhances ABSA performance, though smaller LLMs[1] exhibit weaker performance. Furthermore, we enhance smaller LLMs via distillation using GPT-3.5-generated Syn-Chain responses, achieving state-of-the-art ABSA performance. Our findings highlight the importance of syntactic information for improving LLMs in ABSA and offer valuable insights for future research[2].

1 Introduction

Aspect-based sentiment analysis (ABSA) is a fine-grained sentiment analysis task which aims to analyze the sentiment polarity associated with specific aspects in a sentence (Pontiki et al., 2014). For example, in the sentence "*Great food but the service was dreadful!*", "*food*" and "*service*" are aspect terms, exhibiting positive and negative sentiments, respectively. The potential to analyze user sentiments at such a granular level has garnered significant attention from both industry and academia.

Identifying an aspect's sentiment depends on capturing the associated opinion, which typically reveals the sentiment orientation (Kandhro et al., 2024). Attention mechanisms have been employed to capture words linked to the aspect term (Yang et al., 2017; Liu and Zhang, 2017; Wang et al., 2016), but these methods often struggle with linguistic complexities (Sun et al., 2019). Consequently, incorporating syntactic information has gained widespread acceptance (Zhang et al., 2019; Nazir et al., 2022). Some studies constructed connections between words based on syntactic dependency trees and utilized graph neural networks to capture latent relationships among sentiment elements (Wang et al., 2020; Wu et al., 2021; Chen et al., 2022; Yang et al., 2023; Zhong et al., 2023). Leveraging additional syntactic information along with word dependencies significantly enhances ABSA performance.

Recently, large language models (LLMs) have demonstrated exceptional generalization and contextual understanding capabilities. They also exhibit impressive sentiment comprehension abilities (Wang et al., 2024b). The incorporation of instructions and demonstrations within prompts, following the in-context learning (ICL) paradigm, has further boosted LLMs performance (Zhang et al., 2023). Yang et al. (2024) improved LLMs performance in cross-domain ABSA tasks by providing diverse examples. Wang et al. (2024a) trained a retriever that selects examples based on semantic relevance, syntactic structure, and aspect-sentiment alignment, but LLMs have not directly leveraged syntactic information.

Given the success of syntactic information in prior ABSA studies, and the findings by Roy et al. (2023), which show that LLMs possess a certain degree of syntactic and semantic parsing abilities, we are motivated to explore whether LLMs can utilize syntactic information to further enhance their performance in ABSA. Drawing inspiration from the multi-hop reasoning Chain-of-Thought (CoT) approach (Fei et al., 2023) for ABSA , we propose a

[*]Corresponding author
[†]Equal contribution

[1]In this study, smaller LLMs refer to language models with fewer than 10 billion(B) parameters.

[2]https://github.com/rf-x/Syn-Chain-ABSA

Proceedings of the 31st International Conference on Computational Linguistics, pages 3123–3137
January 19–24, 2025. ©2025 Association for Computational Linguistics

Syntax-Opinion-Sentiment Reasoning Chain (Syn-Chain): (1) analyzing syntactic dependency information, (2) capturing aspect-related opinions using dependency relations, and (3) performing comprehensive sentiment analysis.

Our experiments on several ABSA benchmark datasets demonstrate that Syn-Chain significantly improves LLM performance in ABSA, although smaller LLMs with limited syntactic understanding benefit less. Additionally, we leverage GPT-3.5 to automatically generate Syn-Chain reasoning information for supervised training. We fine-tune T5 (Raffel et al., 2020) and Flan-T5 (Chung et al., 2024) using full-parameter tuning, and Llama-2-7B (Touvron et al., 2023) and Llama-3-8B (AI@Meta, 2024) using parameter-efficient tuning, both achieving results that surpass baseline models. These findings highlight that even in LLM-based research, syntactic information continues to offer valuable insights for ABSA.

The main contributions of this work include:

- We propose Syn-Chain, a multi-step reasoning framework that integrates syntactic information into LLMs to enhance ABSA performance. To the best of our knowledge, we are the first to explore the direct use of syntactic information in LLMs for ABSA.
- We construct Syn-Chain ABSA datasets for supervised training, enabling smaller LLMs to effectively utilize syntactic information.
- Extensive experiments on ABSA benchmarks validate the effectiveness of Syn-Chain and provide insights into integrating syntactic information within LLMs.

2 Related Work

As a subtask of sentiment analysis, ABSA poses a more intricate challenge, garnering substantial attention in recent years. A key point of ABSA is capturing information related to aspect terms. Attention mechanisms have been widely adopted to capture contextual relationships with target words (Tang et al., 2016; Wang et al., 2016; Cheng et al., 2017; Ma et al., 2017; Li et al., 2018; Gu et al., 2018; Fan et al., 2018). However, these mechanisms often underperform when opinion words are distant from their corresponding aspects. To address this, researchers have explored leveraging syntactic information to identify relevant words for aspects (Huang and Carley, 2019; Zhang et al., 2019). Specifically, syntactic dependency trees and graph neural networks have been constructed to model the intricate connections between words and target aspects (Huang and Carley, 2019; Sun et al., 2019; Wang et al., 2020; Zhang et al., 2022; Liang et al., 2022).

More recently, the advent of pretrained language models such as BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) has revolutionized ABSA, yielding impressive results (Song et al., 2019; Jiang et al., 2019). For instance, Wang et al. (2020) combined BERT with graph neural networks to encode syntactic information (Wang et al., 2020), and (Phan and Ogunbona, 2020) explored grammatical and syntactic features based on BERT to enhance ABSA performance.

LLMs have made significant advancements in sentiment analysis, especially in zero-shot settings (Zhang et al., 2023; Wang et al., 2024b). Their emergence has facilitated novel techniques like CoT reasoning (Wei et al., 2022) and ICL (Dong et al., 2024). For instance, Fei et al. (2023) leveraged multi-step CoT for comprehending implicit sentiments and opinions. Yang et al. (2024) investigated multi-domain ABSA under the ICL paradigm. Additionally, Wang et al. (2024a) trained a retriever to select demonstrations for ICL.

Despite these advancements, current LLM-based ABSA research largely overlooks syntactic information, a crucial component in traditional ABSA methods. Thus, this study aims to explore if incorporating syntactic knowledge can further enhance their ABSA performance.

3 Methodology

3.1 Task Definition

Given a sentence X with an aspect term $A \in X$, the goal of ABSA is to predict the sentiment polarity $y \in \{Positive, Negative, Neutral\}$ towards A. LLMs tackle this task by processing a prompt that includes X and A, subsequently generating a sentiment prediction, that is $P(y|X, A)$.

3.2 Syn-Chain Prompting

As shown in Fig. 1, Syn-Chain consists of three sequential steps: syntactic parsing, opinion extraction, and sentiment classification. We provide the following notation: S represents the syntactic dependency sequence; Q_1, Q_2, and Q_3 denote the analysis targets of the three steps, respectively; and

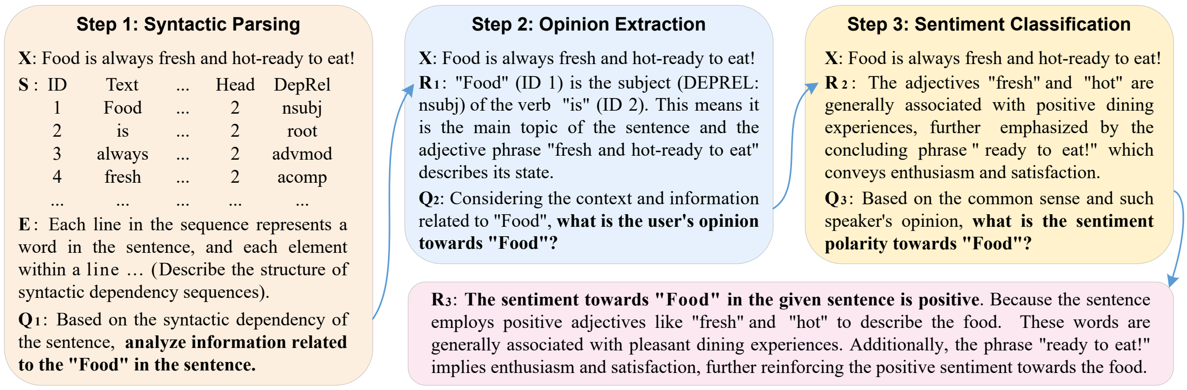

Figure 1: Overview of the Syn-Chain Prompting Process. The response from the previous step is utilized in the subsequent step.

R_1, R_2, and R_3 represent the corresponding outputs of LLMs.

Step 1: Syntactic Parsing. We begin by using spaCy[3] to generate a dependency parse for each sentence, which is converted into the CoNLL-U format, a tabular representation for LLM processing. Each line in the CoNLL-U format corresponds to a word and its associated attributes, such as part-of-speech and dependency relations (see Appendix A for details).

Given the discrepancy between syntactic dependency information and natural language expressions, we add the descriptions of each attribute's meaning, denoted as E. By querying the LLMs about aspect-related information, we obtain natural and readable description of the syntactic structure, denoted as R_1, This process can be formalized as $P(R_1|X, S, E, Q_1)$.

Step 2: Opinion Extraction. Building upon the syntactic analysis provided by R_1, we ask the LLM to analyze user opinions related to specific aspects. The resulting response, $R2$, is generated in the format $P(R_2|X, R_1, Q_2)$.

Step 3: Sentiment Classification. Leveraging the aspect opinions from R_2, the LLM classifies the sentiment polarity of the aspect term, producing a reasoned judgment denoted as R_3. This process is represented as $P(R_3|X, R_2, Q_3)$. By extracting polarity words from R_3, we derive the predicted sentiment label $\hat{y}$.

3.3 Syn-Chain Supervised Learning

The understanding and reasoning capabilities of smaller models are limited, particularly when dealing with syntactic dependency information that is expressed differently in natural language. Leveraging reasoning information from powerful LLMs as supervisory signals and distilling this knowledge into smaller models can effectively enhance their performance on specific tasks (Li et al., 2022; Ho et al., 2023; Fan et al., 2024). To this end, we construct Syn-Chain data for fine-tuning smaller models.

Data Construction. As shown in Fig. 1, the responses R_1, R_2, and R_3 generated by the LLM are directly used as target responses for model training. However, incorrect sentiment predictions by the LLM can result in unreasonable responses that mislead the model during training. To ensure consistency between each step's response and the final sentiment label, we obtain reasoning by inputting prompts conditioned on the true sentiment label y. This process is formalized as $P(\tilde{R}_j|C_j, y)$, where C_j represents the input for each step (where $j \in [1, 2, 3]$), and $\tilde{R}_j$ is the corresponding LLM-generated response.

Model Training. Following the construction of the Syn-Chain dataset, we employ supervised learning to train the model. Technically, Syn-Chain supervised learning is a multi-task learning framework that addresses three primary tasks: syntactic parsing, opinion extraction, and sentiment classification. The model's inputs and outputs align with the steps outlined in Section 3.2, where the goal of outputs is to generate $\tilde{R}_j$. The unified training objective for all three tasks can be formulated as a generative loss:

$$\mathcal{L}_G = -\frac{1}{N}\sum_{i}^{N}\sum_{t=1}^{T}\log P(g_{i,t}|\hat{g}_{i,<t}, C) \qquad (1)$$

where N represents the total number of samples, and T represents the length of the sequence for

[3] https://spacy.io/

Dataset	Split	Pos.	Neu.	Neg.	Total	Len of S	Len of R_1	Len of R_2	Len of R_3
Rest14	Train	2164	633	805	3602	17.54	108.05	59.55	38.28
	Test	728	196	196	1120	16.31	108.72	58.67	38.15
Lap14	Train	987	460	866	2313	19.28	108.84	62.03	39.42
	Test	341	169	128	638	15.99	111.98	61.89	39.34
Rest15	Train	912	36	256	1204	16.98	110.00	58.83	37.89
	Test	326	34	182	542	18.17	109.63	59.69	37.38
Rest16	Train	1240	69	439	1748	17.33	110.20	59.38	37.83
	Test	469	30	117	616	19.71	110.86	59.20	37.68

Table 1: Statistics of the ABSA datasets. "Len" represents the average length of the corresponding sequence.

each sample, $g_{i,t}$ denotes the true token at position t, $\hat{g}_{i,<t}$ denotes the generated sequence up to position t, and $P(g_{i,t}|\hat{g}_{i,<t}, C)$ is the probability of generating token $g_{i,t}$ given $\hat{g}_{i,<t}$ and input C.

Error Propagation. In executing *Step 2* and *Step 3* of Syn-Chain, the model depends on the output from the prior step. During training, ground-truth responses are used, but during inference, the model relies on its generated outputs, leading to potential error propagation. Errors in earlier steps can accumulate, affecting sentiment predictions. To mitigate this issue, we propose a training strategy to "break the chain", which prevents error propagation by ensuring that the response from the previous step is not used as input for the subsequent step. For example, the notation $_1 \oplus {_2} \oslash {_3}$ indicates that *Step 1* and *Step 2* are linked, and the connection between *Step 2* and *Step 3* is deliberately severed. In other words, R_2 from *Step 2* is not used as input to *Step 3*, which is formally represented as $P(R_3|X, Q_3)$. This ensures that any errors from the first two steps do not affect predictions in the third step.

The training of *Step 1* and *Step 2* is designed to equip the model with the ability to comprehend syntactic information and capture aspect-related opinions. Even when the chain is broken, the model retains these capabilities within the multi-task learning framework, while minimizing the risk of error accumulation. Notably, our breaking strategy is based on independent *Step 3*, and the model is trained to focus exclusively on sentiment prediction from the first sentence of $\tilde{R}_3$, excluding subsequent reasoning content.

4 Experimental Setup

4.1 Datasets

We utilize four ABSA datasets in our experiments: Rest14 and Lap14, sourced from (Pontiki et al., 2014), Rest15 from (Pontiki et al., 2015), and Rest16 from (Pontiki et al., 2016). To balance experimental rigor with resource constraints, we focus on Rest14 and Lap14 for zero-shot experiments due to the high cost of LLM API usage. This allows us to conduct in-depth ablation studies. Fine-tuning experiments are conducted on local GPUs. This setup enabled us to compare a broader range of methods using four common datasets.

Detailed statistics of these datasets are shown in Table 1. Instances with multiple aspects within a single sentence are split into multiple single-aspect samples. The supplementary Syn-Chain data substantially enrich the informational content of the original sentences.

4.2 Implement Details and Metrics

We evaluate Syn-Chain prompting on several LLMs including GPT-3.5, GPT-4o[4], Gemini-1.5[5], Llama-2-7B (Touvron et al., 2023), and Llama-3-8B (AI@Meta, 2024). Details of the prompts can be found in Appendix A. For supervised fine-tuning, we ensure a fair comparison with previous methods by performing full-parameter fine-tuning on T5 and Flan-T5. Additionally, we employ LoRA fine-tuning (Hu et al., 2022) for Llama-2-7B and Llama-3-8B. All model parameters are obtained from the Transformers library[6]. For full fine-tuning, we set the learning rate to 5e-5, batch size to 4, and train for 10 epochs. For LoRA fine-tuning, we use a learning rate of 1e-4, batch size of 2, and train for 10 epochs with $\alpha = 16$ and $r = 8$. Both setups use the AdamW optimizer.

The experiments are implemented in PyTorch and execute on two A5000 GPUs. We employ Accuracy (Acc) and Macro-F1 score (F1) for evaluation. For fine-tuned models, results are reported as the average of three independent runs.

[4]OpenAI API: https://openai.com/api/
[5]Google Gemini API: https://ai.google.dev/
[6]https://huggingface.co/

Method	Rest14		Lap14	
	Acc	F1	Acc	F1
GPT-3.5	82.50	71.27	79.62	74.63
+ Syn-Chain	**85.08**	**78.61**	**80.09**	**76.45**
GPT-4o	86.96	76.36	82.13	76.40
+ Syn-Chain	**90.26**	**83.85**	**82.60**	**78.01**
Gemini-1.5	87.41	80.22	78.52	71.66
+ Syn-Chain	**87.58**	**81.06**	**79.31**	**74.54**
Llama-2-7B	74.73	**65.02**	**73.04**	**69.10**
+ Syn-Chain	**77.58**	64.10	71.00	64.56
Llama-3-8B	**82.23**	67.83	**78.36**	**71.73**
+ Syn-Chain	81.42	**71.73**	75.23	70.97
T5	87.50	81.44	80.87	76.69

Table 2: Experimental results of Syn-Chain Zero-shot Prompting on ABSA. T5 is fine-tuned on ABSA and provided for comparison.

Format	Syn-Chain$^{1\oplus3}$		Syn-Chain	
	Acc	F1	Acc	F1
CoNLL-U	**77.56**	**71.68**	80.09	76.45
XML	76.33	68.29	**81.50**	**77.21**
JSON	76.80	68.87	**81.50**	77.15
HTML	76.48	68.76	80.87	77.11

Table 3: Performance comparison of GPT-3.5 on Lap14 using different syntactic representations. $1 \oplus 3$ indicates that only Steps 1 and 3 of the Syn-Chain are used.

5 Experimental Results and Analysis

5.1 Syn-Chain Zero-shot Prompting

Main Results. As shown in Table 2, we compare the performance of various LLMs before and after applying the Syn-Chain strategy. For more powerful models, such as GPT-3.5 and GPT-4o, the performance improves significantly after using Syn-Chain. Specifically, on the Res14 dataset, the F1 scores of GPT-3.5 and GPT-4o increase by 7.34% and 7.49%, respectively. GPT-4o achieves the best performance on both datasets, even surpassing fine-tuned models. However, smaller LLMs, Llama-2-7B and Llama-3-8B, generally experience a decline in performance. Their limited capacity makes it difficult to benefit from the complex reasoning process of Syn-Chain.

Effectiveness of Syn-Chain. As shown in Table 3, a comparison between removing *Step 2* ($1 \oplus 3$) and the full Syn-Chain strategy highlights the importance of opinion extraction (*Step 2*). The performance of $1 \oplus 3$ is lower than that of direct sentiment prediction on aspect terms using LLMs. This result indicates that Syn-Chain effectively leverages syntactic information to enhance LLM performance.

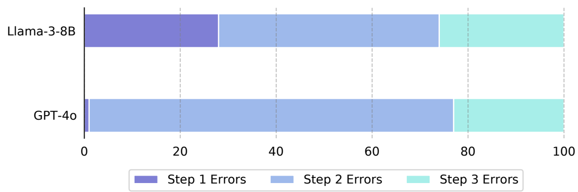

Figure 2: Proportions of different error types in Syn-Chain sentiment prediction errors.

In Syn-Chain, syntactic information supports opinion extraction, which in turn strengthens sentiment analysis. Directly incorporating syntactic information for sentiment analysis introduces weaker task interdependencies, making the model's reasoning more complex and limiting performance improvements. By decomposing the task into smaller, more manageable steps, Syn-Chain more effectively harnesses the model's reasoning and comprehension capabilities.

Syntactic Information Format Analysis. As shown in Table 3, we compare the effects of different data formats. We devise a tabular representation using newlines and tabs to incorporate syntactic dependency information effectively into LLMs. This approach replaces traditional tree structures to better align with the LLMs' sequential processing nature. Additionally, we try other structured formats such as XML, JSON, and HTML. When employing *Step 1* and *Step 3* ($1 \oplus 3$), the CoNLL-U format achieves the best performance. However, with the full Syn-Chain strategy, XML format outperforms the others. This suggests that LLMs can understand and reason with various syntactic formats. In subsequent fine-tuning experiments, we use the CoNLL-U format due to its simplicity and minimal introduction of special symbols, reducing the model's learning complexity.

Error Analysis. To gain deeper insights into the models' understanding of syntactic information, we conduct a manual analysis of Syn-Chain results from GPT-4o and Llama-3-8B the best-performing models among the large and small models, respectively. Specifically, we randomly select 100 samples from the sentiment prediction errors: 50 from Rest14 and 50 from Lap14. Through manual evaluation, we identify at which step in Syn-Chain errors occur. We categorize the errors into three types: 1) *Step 1* errors, where the syntactic description does not align with the provided syntactic information, 2) *Step 2* errors, where the opinion description is inconsistent with the original sentence, and 3) *Step 3* errors, where the final sentiment judgment is incorrect. Since the process follows a chain structure, if an earlier step is incorrect, subsequent steps are not evaluated (e.g., if *Step 1* is incorrect, *Step 2* and *Step 3* are not considered). The evaluation is conducted independently by three master's students with expertise in English and natural language processing (NLP). To mitigate the risk of overlooking errors in the lengthy model responses, a consensus-based approach is adopted to resolve any discrepancies among the evaluators.

As shown in Fig. 2, GPT-4o rarely makes mistakes in *Step 1*, demonstrating a solid understanding of the provided syntactic information. In contrast, Llama-3-8B exhibits more errors in syntactic comprehension. Additionally, we observe that GPT-4o's outputs are more structured and easier to read (see Fig. 5 in the Appendix). Both models show higher error rates in *Step 2* than in *Step 3*, as the sentiment inclination is often conveyed in *Step 2*, making it account for a higher proportion of errors in these sentiment misclassification cases. This also highlights the importance of generating reasoning information conditioned on sentiment labels when constructing Syn-Chain fine-tuning data, which avoid inconsistencies between reasoning and sentiment labels.

5.2 Syn-Chain Fine-tuning

Baseline. We select BERT-SPC (Song et al., 2019) and BERT-PT (Xu et al., 2019), both of which are BERT-based models. Additionally, we consider several works that combine BERT with graph neural networks to leverage syntactic dependency trees. BERT-RGAT (Wang et al., 2020) employs relational graph attention to capture syntactic dependencies. BERT-DualGCN (Li et al., 2021) integrates both syntactic and semantic knowledge. BERT-TGCN (Tian et al., 2021) incorporates dependency types into the graph convolutional network. BERT-SenticGCN (Liang et al., 2022) further enriches the dependency graph with affective knowledge. Moreover, we fine-tune T5, Flan-T5, Llama-2-7B, and Llama-3-8B on ABSA to provide additional baselines.

Main Results. As shown in Table 4, generative models such as T5 and Flan-T5 outperform BERT-based methods. Specifically, T5-base and Flan-T5-base excess BERT-SPC by 4.44% and 5.24% in average F1 scores. While BERT combined with graph neural networks to incorporate syntactic information yields significant performance improvements, our Syn-Chain$^{1\oplus 2\oslash 3}$ training strategy achieves the best results with T5-base and Flan-T5-base, surpassing the previously top-performing BERT-SenticGCN across all four datasets.

As model size increases, Flan-T5-large demonstrates noticeably better performance than Flan-T5-base. Furthermore, LoRA fine-tuning on larger models such as Llama-2-7B and Llama-3-8B leads to further improvements in ABSA performance.

Effectiveness of Breaking Chain. We evaluate the full Syn-Chain and three variations of the broken Syn-Chain: $_1 \oslash _3$, $_1 \oslash _2 \oslash _3$, and $_1 \oplus _2 \oslash _3$. The numbers 1, 2, and 3 correspond to the three sequential steps in the Syn-Chain process, while $\oplus$ and $\oslash$ represent linked and broken steps, respectively. The broken versions are designed with an independent *Step 3* to prevent the final sentiment prediction from being influenced by erroneous information.

As shown in Table 4, the best performance across different models is consistently achieved by Syn-Chain$^{1\oplus 2\oslash 3}$, which breaks the connection between *Step 2* and *Step 3*. This ensures sufficient learning of syntactic information while avoiding the propagation of errors to the final sentiment classification. In contrast, the full Syn-Chain$^{1\oplus 2\oplus 3}$ generally performs worse than the baseline. Smaller models have limited reasoning capabilities, and even with supervised learning, they may struggle to maintain accuracy in multi-step reasoning.

When comparing Syn-Chain$^{1\oslash 3}$ and Syn-Chain$^{1\oslash 2\oslash 3}$, the latter demonstrates superior performance, with higher average F1 scores. This suggests that learning from more related tasks effectively enhances the model's performance.

Impact of Error Propagation. To investigate the impact of error propagation, we conduct two comparative experiments on Llama-3-8B trained with Syn-Chain$^{1\oplus 2\oplus 3}$. In one experiment, the ground-truth response from *Step 1* is used in *Step 2* during inference, ensuring that no errors are introduced in the first step. In the other experiment, ground-truth responses of previous steps are used as inputs for both *Step 2* and *Step 3*, guaranteeing that the first two steps are error-free. These experiments allow us to evaluate the impact of errors introduced at different stages of Syn-Chain.

As shown in Table 5, when only $\tilde{R}_1$ is used, there is a slight performance improvement. How-

Method	Rest14		Lap14		Rest15		Rest16		AVG
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	F1
Full Fine-Tuning									
BERT-SPC	84.11	76.68	77.59	73.28	83.48	66.18	90.10	74.16	72.58
BERT-PT	84.95	76.96	78.07	75.08	-	-	-	-	-
BERT-RGAT	85.18	78.38	78.21	73.27	82.84	69.33	90.91	75.76	74.19
BERT-DualGCN	**87.13**	**81.16**	81.80	78.10	-	-	-	-	-
BERT-TGCN	86.16	79.95	80.88	77.03	85.26	**71.69**	**92.32**	77.29	76.49
BERT-SenticGCN	86.92	81.03	**82.12**	**79.05**	**85.32**	71.28	91.07	**79.56**	**77.73**
T5-base	87.50	81.44	80.87	76.69	86.34	72.03	93.83	77.92	77.02
+ Syn-Chain$^{1\oplus2\oplus3}$	81.07	72.43	78.19	73.68	78.59	65.52	90.58	69.28	70.02
+ Syn-Chain$^{1\oslash3}$	87.76	81.20	81.50	78.03	86.34	73.13	92.85	77.54	77.48
+ Syn-Chain$^{1\oslash2\oslash3}$	88.12	82.05	81.22	78.71	86.34	73.76	93.50	79.98	78.63
+ Syn-Chain$^{1\oplus2\oslash3}$	**88.30**	**82.47**	**81.81**	**79.12**	**86.71**	**74.17**	**94.37**	**80.15**	**78.98**
Flan-T5-base	87.58	81.36	81.66	78.42	86.90	74.20	93.34	77.30	77.82
+ Syn-Chain$^{1\oplus2\oplus3}$	81.78	72.23	80.56	76.09	85.97	67.47	93.66	74.26	72.51
+ Syn-Chain$^{1\oslash3}$	86.69	79.91	82.28	79.00	85.05	72.84	92.50	78.86	77.65
+ Syn-Chain$^{1\oslash2\oslash3}$	87.75	82.47	81.97	78.72	87.26	76.04	93.18	76.77	78.50
+ Syn-Chain$^{1\oplus2\oslash3}$	**88.39**	**82.79**	**83.22**	**80.04**	**87.82**	**76.86**	**93.50**	**79.25**	**79.73**
Flan-T5-large	89.62	84.19	84.16	81.04	89.22	78.08	94.61	84.62	81.98
+ Syn-Chain$^{1\oplus2\oplus3}$	85.98	76.19	80.09	74.84	88.00	73.70	94.80	80.25	76.25
+ Syn-Chain$^{1\oslash3}$	89.08	84.15	84.01	81.44	87.82	78.54	92.12	83.31	81.86
+ Syn-Chain$^{1\oslash2\oslash3}$	89.71	84.81	84.63	82.61	89.81	78.91	95.12	84.61	82.74
+ Syn-Chain$^{1\oplus2\oslash3}$	**90.35**	**85.30**	**85.10**	**82.72**	**90.22**	**79.04**	**95.29**	**85.03**	**83.62**
LoRA Fine-Tuning									
Llama-2-7B	89.73	84.86	84.79	81.85	90.22	78.52	94.48	82.26	81.87
+ Syn-Chain$^{1\oplus2\oplus3}$	86.87	79.61	82.75	79.43	90.03	69.43	93.18	72.86	75.33
+ Syn-Chain$^{1\oslash3}$	89.91	84.67	84.63	82.41	89.66	81.17	94.80	81.94	82.55
+ Syn-Chain$^{1\oslash2\oslash3}$	90.44	85.14	85.57	83.07	**91.88**	82.31	**94.96**	84.43	83.74
+ Syn-Chain$^{1\oplus2\oslash3}$	**90.71**	**86.10**	**85.89**	**83.28**	**91.88**	**82.71**	**94.96**	**84.78**	**84.22**
Llama-3-8B	90.62	85.10	84.48	81.33	91.69	81.39	94.96	83.46	82.82
+ Syn-Chain$^{1\oplus2\oplus3}$	87.23	81.67	82.60	79.27	89.37	72.30	93.99	77.70	77.74
+ Syn-Chain$^{1\oslash3}$	91.25	86.94	86.05	83.45	**92.61**	82.80	95.29	83.41	84.15
+ Syn-Chain$^{1\oslash2\oslash3}$	90.71	86.25	**86.52**	84.05	91.51	81.36	95.45	84.89	84.14
+ Syn-Chain$^{1\oplus2\oslash3}$	**91.87**	**87.61**	86.36	**84.21**	91.88	**83.65**	**95.94**	**85.50**	**85.24**

Table 4: Experimental results of fine-tuned models on ABSA. AVG is the average performance of the model across four data sets. The superscript numbers 1, 2, and 3 correspond to the three sequential steps in the Syn-Chain process. $\oplus$ and $\oslash$ denote link and break, respectively. Different combinations represent various training strategies.

Input	Rest14		Lap14		Rest15		Rest16	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Syn-Chain$^{1\oplus2\oplus3}$	87.23	81.67	82.60	79.27	89.37	72.30	93.99	77.70
w/ $\tilde{R}_1$	88.14	82.57	83.54	80.35	89.72	73.70	94.50	78.95
w/ $\tilde{R}_1$, $\tilde{R}_2$	**97.94**	**97.00**	**97.02**	**96.49**	**94.49**	**94.22**	**95.19**	**92.53**

Table 5: Performance of Syn-Chain fine-tuned Llama-3-8B using ground-truth $\tilde{R}_1$ or $\tilde{R}_2$ during inference, illustrating the impact of error propagation.

ever, when both $\tilde{R}_1$ and $\tilde{R}_2$ are utilized, the model achieves a significant boost in performance. The majority of errors tend to originate from *Step 2*, as the response in this step typically conveys sentiment polarity, which strongly influences the final outcome. Therefore, the use of the ground-truth $\tilde{R}_2$ results in a substantial performance improvement by minimizing errors propagated from *Step 2*.

Input	Rest14		Lap14	
	Acc	F1	Acc	F1
X, A	87.50	81.44	80.87	76.69
X, A, $\tilde{R}_1$	87.32	81.17	81.81	77.98
X, A, $\tilde{R}_2$	**98.12**	**97.29**	**97.49**	**97.07**

Table 6: Automatic evaluation results of $\tilde{R}_1$ and $\tilde{R}_2$. Perform sentiment classification on different inputs using the fine-tuned T5.

Method	Rest14		Lap14	
	Acc	F1	Acc	F1
GPT-3.5	82.50	71.27	79.62	74.63
+ Syn-Chain$^{1\oplus3}$	82.05	69.01	77.56	71.68
+ Syn-Chain$^{2\oplus3}$	83.83	69.38	77.27	67.15
+ Syn-Chain$^{1\oplus2\oplus3}$	**85.08**	**78.61**	**80.09**	**76.45**
Llama-3-8B	90.62	85.10	84.48	81.33
+ Syn-Chain$^{1\oslash3}$	91.25	86.94	86.05	83.45
+ Syn-Chain$^{2\oslash3}$	90.35	85.28	86.05	83.86
+ Syn-Chain$^{1\oplus2\oslash3}$	**91.87**	**87.61**	**86.36**	**84.21**

Table 7: Performance comparison of models under zero-shot and fine-tuned settings using syntactic information, opinion extraction, and both.

5.3 Syn-Chain Data Quality Analysis

We use GPT-3.5 to automatically generate Syn-Chain reasoning information for supervised learning. To better understand the quality of the generated data, we conduct data quality analysis through both automatic and manual evaluation. In the third step of training, we exclude the reasoning information. Therefore the training data of *Step 3* can be guaranteed to be consistent with sentiment labels. We do not assess its quality.

Automatic Evaluation. We train two specialized aspect-level sentiment classifiers based on T5 to assess the influence of $\tilde{R}_1$ and $\tilde{R}_2$. These classifiers take X and A as inputs, with $\tilde{R}_1$ or $\tilde{R}_2$ added respectively, formulated as $P(y|X, A, \tilde{R}_1)$ and $P(y|X, A, \tilde{R}_2)$. We train and test the models on the Rest14 and Lap14 datasets. The results, shown in Tabel 6, indicate that $\tilde{R}_1$ has no significant impact on sentiment classification, while $\tilde{R}_2$ achieves near-perfect performance. This is likely because $\tilde{R}_2$ typically contains words that explicitly express sentiment polarity, indicating a high consistency between $\tilde{R}_2$ and sentiment labels. $\tilde{R}_1$ is less directly useful for sentiment classification, making its quality difficult to assess automatically.

This raises the question of whether the improvement in Syn-Chain is driven by syntactic parsing or opinion extraction. In other words, does solely relying on opinion extraction lead to enhanced sentiment classification performance? We explore in Section 5.4.

Manual Evaluation. We randomly selected 100 samples, 50 from the Lap14 dataset and 50 from the Rest14 dataset, to assess the correctness of the generated $\tilde{R}_1$ and $\tilde{R}_2$. Three master's students proficient in English and NLP independently evaluated the correctness of these outputs. The evaluation of $\tilde{R}_1$ focus on its alignment with the provided syntactic information, while the evaluation of $\tilde{R}_2$ center on its consistency with the original sentence and sentiment label. In cases of disagreement, a consensus is reached through discussion. GPT-3.5 exhibits a low error rate of only 5% for both $\tilde{R}_1$ and $\tilde{R}_2$, demonstrating strong comprehension capabilities. The few errors observed in $\tilde{R}_1$ are primarily due to incorrect dependency relation descriptions, whereas those in $\tilde{R}_2$ are mainly attributed to inconsistencies between the generated content and the sentiment polarity.

5.4 Role of Syntax and Opinion

As shown in Table 7, we conduct experiments under both zero-shot GPT-3.5 and fine-tuned Llama-3-8B, comparing models that rely solely on syntactic information, those that rely solely on opinion extraction, and those that utilize both. Experimental results demonstrate that models relying on a single information source experience a performance decline. Specifically, for GPT-3.5, only by combining the syntactic analysis step and the opinion extraction step can we significantly surpass the baseline. Llama-3-8B under fine-tuned settings also validates this synergistic effect. These findings indicate that syntactic information and opinion extraction play complementary roles in sentiment classification, jointly promoting a deeper understanding of text by the model. The Syn-Chain design effectively enhances model performance by organically combining these two information sources, validating its effectiveness in ABSA.

6 Conclusion

In this paper, we propose Syn-Chain, a novel framework that integrates syntactic information to improve the performance of LLMs in ABSA. By decomposing ABSA into three sequential steps: syntactic parsing, opinion extraction, and sentiment classification, Syn-Chain enables more effective utilization of syntactic dependencies and enhances the model's reasoning capabilities. Furthermore, we fine-tune smaller LLMs and introduce a breaking-chain strategy that mitigates error propagation, ensuring accurate sentiment classification even when earlier steps contain errors. Our experimental results demonstrate that Syn-Chain significantly improves LLMs' zero-shot ability in ABSA and out fine-tuned performance significantly outperforms previous models. This work highlights the potential of syntactic information to enhance LLM performance in ABSA tasks.

Limitations

While our proposed Syn-Chain framework demonstrates notable improvements in ABSA, there are several limitations to our approach. First, ABSA consists of various sub-tasks beyond sentiment classification, such as aspect term and opinion extraction. LLMs generally struggle with extraction tasks, particularly in identifying the correct boundaries of aspect terms and opinions, leading to inconsistencies with the golden labels. This limitation arises from the inherent difficulty LLMs face in aligning extracted spans with precise human annotations. Although our method incorporates syntactic information to improve sentiment classification, its effectiveness in significantly enhancing extraction tasks remains uncertain.

Furthermore, although Syn-Chain demonstrates that LLMs can effectively understand and utilize syntactic information, the datasets used in our experiments are primarily derived from user reviews, which generally feature relatively simple syntactic structures. This simplicity may not fully capture the complexity of more diverse sentence structures found in other domains, such as news articles or academic texts. As a result, the applicability and robustness of Syn-Chain in handling more complex syntactic dependencies remain to be fully evaluated. Future work should explore its effectiveness across a wider range of text types with more intricate syntactic characteristics to determine whether the framework can generalize to more complex linguistic environments.

Ethics Statement

In this research, we utilize publicly available datasets, which have been widely used in the academic community. As such, there are no concerns regarding copyright infringement or proprietary data. The use of these datasets adheres to the terms of use specified by the dataset providers.

For the manual evaluation, we employed three master's students, each tasked with analyzing 300 samples. Each evaluator were compensated $100 for their time and effort. This payment rate reflects a fair and competitive compensation for the amount of work involved. Additionally, all evaluators participated voluntarily and were informed of the tasks and compensation prior to the study.

Acknowledgements

This work was supported by the National Language and Character Research Base.

References

AI@Meta. 2024. Llama 3 model card.

Hao Chen, Zepeng Zhai, Fangxiang Feng, Ruifan Li, and Xiaojie Wang. 2022. Enhanced multi-channel graph convolutional network for aspect sentiment triplet extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2974–2985, Dublin, Ireland. Association for Computational Linguistics.

Jiajun Cheng, Shenglin Zhao, Jiani Zhang, Irwin King, Xin Zhang, and Hui Wang. 2017. Aspect-level sentiment classification with heat (hierarchical attention) network. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, page 97–106, New York, USA. Association for Computing Machinery.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, Xu Sun, Lei Li, and Zhifang Sui. 2024. A survey on in-context learning. *Preprint*, arXiv:2301.00234.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442, Brussels, Belgium. Association for Computational Linguistics.

Rui Fan, Tingting He, Menghan Chen, Mengyuan Zhang, Xinhui Tu, and Ming Dong. 2024. Dual causes generation assisted model for multimodal aspect-based sentiment classification. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–15.

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.

Shuqin Gu, Lipeng Zhang, Yuexian Hou, and Yin Song. 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 774–784, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. Large language models are reasoning teachers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14852–14882, Toronto, Canada. Association for Computational Linguistics.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Binxuan Huang and Kathleen 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 and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5469–5477, Hong Kong, China. Association for Computational Linguistics.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285, Hong Kong, China. Association for Computational Linguistics.

Irfan Ali Kandhro, Fayyaz Ali, Mueen Uddin, Asadullah Kehar, and Selvakumar Manickam. 2024. Exploring aspect-based sentiment analysis: an in-depth review of current methods and prospects for advancement. *Knowledge and Information Systems*, pages 1–31.

Lishuang Li, Yang Liu, and AnQiao Zhou. 2018. Hierarchical attention based position-aware network for aspect-level sentiment analysis. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 181–189, Brussels, Belgium. 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.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, Wenhu Chen, and Xifeng Yan. 2022. Explanations from large language models make small reasoners better. *Preprint*, arXiv:2210.06726.

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.

Jiangming Liu and Yue Zhang. 2017. Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 572–577, Valencia, Spain. Association for Computational Linguistics.

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

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4068–4074.

Ambreen Nazir, Yuan Rao, Lianwei Wu, and Ling Sun. 2022. Issues and challenges of aspect-based sentiment analysis: A comprehensive survey. *IEEE Transactions on Affective Computing*, 13(2):845–863.

Minh Hieu Phan and Philip O. Ogunbona. 2020. Modelling context and syntactical features for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3211–3220, Online. Association for Computational Linguistics.

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

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

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

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. *Journal of Machine Learning Research*, 21(140):1–67.

Subhro Roy, Samuel Thomson, Tongfei Chen, Richard Shin, Adam Pauls, Jason Eisner, and Benjamin Van Durme. 2023. Benchclamp: A benchmark for evaluating language models on syntactic and semantic parsing. In *Advances in Neural Information Processing Systems*, volume 36, pages 49814–49829. Curran Associates, Inc.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Targeted sentiment classification with attentional encoder network. *Preprint*, arXiv:1902.09314.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5679–5688, Hong Kong, China. Association for Computational Linguistics.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016. Effective LSTMs for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307, Osaka, Japan. The COLING 2016 Organizing Committee.

Yuanhe Tian, Guimin Chen, and Yan Song. 2021. Aspect-based sentiment analysis with type-aware graph convolutional networks and layer ensemble. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2910–2922, Online. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3229–3238, Online. Association for Computational Linguistics.

Qianlong Wang, Hongling Xu, Keyang Ding, Bin Liang, and Ruifeng Xu. 2024a. In-context example retrieval from multi-perspectives for few-shot aspect-based sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8975–8985, Torino, Italia. ELRA and ICCL.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas. Association for Computational Linguistics.

Zengzhi Wang, Qiming Xie, Yi Feng, Zixiang Ding, Zinong Yang, and Rui Xia. 2024b. Is chatgpt a good sentiment analyzer? a preliminary study. *Preprint*, arXiv:2304.04339.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*. Survey Certification.

Shengqiong Wu, Hao Fei, Yafeng Ren, Donghong Ji, and Jingye Li. 2021. Learn from syntax: Improving pair-wise aspect and opinion terms extractionwith rich syntactic knowledge. In *International Joint Conference on Artificial Intelligence*.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. 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 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen. 2017. Attention based lstm for target dependent sentiment classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

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.

Songhua Yang, Tengxun Zhang, Hongfei Xu, and Yuxiang Jia. 2023. Improving aspect sentiment triplet extraction with perturbed masking and edge-enhanced sentiment graph attention network. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Syntax-aware aspect-level sentiment classification with proximity-weighted convolution network. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*.

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.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022. SSEGCN: Syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4916–4925, Seattle, United States. Association for Computational Linguistics.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, Hua Jin, and Dacheng Tao. 2023. Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 35(10):10098–10111.

Appendix

A Prompt Details

The prompt consists of two parts: the system prompt and the user prompt. The system prompt informs the LLMs of their role, task, and basic requirements, ensuring consistent and coherent responses. For the three steps in Sys-Chain, we design specific system prompts as illustrated in Fig. 3. The system prompts for *Step 1* and *Step 2* primarily focus on limiting the response length, while the system prompt for *Step 3* guides the format of the generated content to facilitate accurate extraction and classification of sentiment polarity. The user prompt, shown in Fig. 4, provides an example of Syn-Chain completing ABSA. In *Step 1*, we provide the syntactic information of the sentence and the corresponding meaning. Fig. 5 presents the results of GPT-4o's syntactic parsing, demonstrating a more structured and readable output.

Step 1: Syntactic Parsing

You are an AI assistant that helps people find information. Please refine your reply and ensure its accuracy. The reply length is limited to 200 words.

Step 2: Opinion Extraction

You are an AI assistant that helps people find information. Please refine your reply and ensure its accuracy. The reply length is limited to 120 words.

Step 3: Sentiment Classification

You are an sentiment analysis expert. I will provide you with a sentence and a certain aspect mentioned in the sentence. Please analyze the sentiment polarity of that aspect in a given sentence.
Output: The sentiment towards {Aspect} in the given sentence is {positive, negative or neutral}. Because...

Figure 3: System content prompts in Syn-Chain

Step 1: Syntactic Parsing

Sentence: " I think the meatball parm is good."

The following is the CoNLL-U format for the syntactic dependency relationship of this sentence:

ID	TEXT	LEMMA	POS	TAG	FEATS	HEAD	DEPREL	DEPS	MISC
1	I	I	PRON	PRP	-	2	nsubj	-	-
2	think	think	VERB	VBP	-	2	root	-	-
3	the	the	DET	DT	-	5	det	-	-
4	meatball	meatball	NOUN	NN	-	5	compound	-	-
5	parm	parm	NOUN	NN	-	6	nsubj	-	-
6	is	be	AUX	VBZ	-	2	ccomp	-	-
7	good	good	ADJ	JJ	-	6	acomp	-	-
8	.	.	PUNCT	.	-	2	punct	-	-

Each row in the table represents a word in the sentence, and each column represents some specific properties of the word, including ID (Word position in the sentence, starting from 1), TEXT (word itself), Lemma (the base form of the word), POS (the simple universal part-of-speech tag), TAG (the detailed part-of-speech tag), FEATS (other grammatical features, empty here), HEAD (the header word of the dependency relationship of the current word), DEPREL (dependency label, describing the relationship between the current word and the header word), DEPS (word dependency, empty here), MISC (other additional information, empty here).

Based on the syntactic dependency information of the sentence, analyze information related to "Meatball parm" in the sentence.

"Meatball parm" is a compound noun phrase in this sentence, acting as the subject of the verb phrase "is good". The dependency parse shows both words are nouns ("NN") and "meatball" modifies " parm" (compound relationship). This tells us the sentence expresses a positive opinion about a dish called "meatball parm".

Step 2: Opinion Extraction

Sentence: " I think the meatball parm is good."

"Meatball parm" is a compound noun phrase in this sentence, acting as the subject of the verb phrase "is good". The dependency parse shows both words are nouns ("NN") and "meatball" modifies " parm" (compound relationship). This tells us the sentence expresses a positive opinion about a dish called "meatball parm".

Considering the context and information related to "meatball parm", what is the speaker's opinion towards "meatball parm"?

The speaker believes that the "meatball parm" is good. This is clear from the sentence's structure where "meatball parm," a compound noun referring to a food dish, is the subject, and "is good" is the predicate attributing a positive quality to the subject.

Step 3: Sentiment Classfication

Sentence: " I think the meatball parm is good."

The speaker believes that the "meatball parm" is good. This is clear from the sentence's structure where "meatball parm," a compound noun referring to a food dish, is the subject, and "is good" is the predicate attributing a positive quality to the subject.

Based on the common sense and such speaker's opinion, what is the sentiment polarity towards "meatball parm"?

The sentiment towards meatball parm in the given sentence is positive. Because the sentence directly states "the meatball parm is good," using the positive adjective "good" to describe it.

Figure 4: A zero-shot Syn-Chain prompt for GPT-3.5.

Step 1: Syntactic Parsing

Sentence: " I think the meatball parm is good."

The following is the CoNLL-U format for the syntactic dependency relationship of this sentence:

ID	TEXT	LEMMA	POS	TAG	FEATS	HEAD	DEPREL	DEPS	MISC
1	I	I	PRON	PRP	-	2	nsubj	-	-
2	think	think	VERB	VBP	-	2	root	-	-
3	the	the	DET	DT	-	5	det	-	-
4	meatball	meatball	NOUN	NN	-	5	compound	-	-
5	parm	parm	NOUN	NN	-	6	nsubj	-	-
6	is	be	AUX	VBZ	-	2	ccomp	-	-
7	good	good	ADJ	JJ	-	6	acomp	-	-
8	.	.	PUNCT	.	-	2	punct	-	-

Each row in the table represents a word in the sentence, and each column represents some specific properties of the word, including ID (Word position in the sentence, starting from 1), TEXT (word itself), Lemma (the base form of the word), POS (the simple universal part-of-speech tag), TAG (the detailed part-of-speech tag), FEATS (other grammatical features, empty here), HEAD (the header word of the dependency relationship of the current word), DEPREL (dependency label, describing the relationship between the current word and the header word), DEPS (word dependency, empty here), MISC (other additional information, empty here).

Based on the syntactic dependency information of the sentence, analyze information related to "Meatball parm" in the sentence.

In the sentence "I think the meatball parm is good," the phrase "meatball parm" consists of two words: "meatball" and "parm." Here is the analysis of these words based on the CoNLL-U format provided:

1. **meatball (ID 4)**
   - **Text:** meatball
   - **Lemma:** meatball
   - **POS:** NOUN
   - **Tag:** NN
   - **FEATS:** _
   - **HEAD:** 5 (parm)
   - **DEPREL:** compound
   - **DEPS:** _
   - **MISC:** _

2. **parm (ID 5)**
   - **Text:** parm
   - **Lemma:** parm
   - **POS:** NOUN
   - **Tag:** NN
   - **FEATS:** _
   - **HEAD:** 6 (is)
   - **DEPREL:** nsubj
   - **DEPS**: _
   - **MISC:** _

Analysis:
"meatball" is a noun that functions as a compound modifier of "parm." This means "meatball" is providing additional descriptive information about "parm."
- **"parm"** is a noun that serves as the nominal subject (nsubj) of the verb "is." This indicates that "parm" is the main subject of the clause "parm is good."

In summary, "meatball parm" is a compound noun phrase where "meatball" modifies "parm," and "parm" is the subject of the clause describing it as "good."

Figure 5: The result of GPT-4o output in syntactic parsing step.