2025 International Conference on Intelligent Computing
July 26-29, Ningbo, China
https://www.ic-icc.cn/2025/index.php

Representative Chain-of-Reasoning Framework for Aspect Sentiment Quad Prediction

Jiajian Li[1], Zhongquan Jian[1,2,] Yancheng Wang[4], Qingqiang Wu[2,1,3], Meihong Wang[2]

[3] School of Film, Xiamen University, Xiamen 361000, China
[1] Institute of Artificial Intelligence, Xiamen University, Xiamen 361000, China
[2] School of Informatics, Xiamen University, Xiamen 361000, China
[4] School of Computing and Data Science, Xiamen University Malaysia, Kuala Lumpur 43900, Malaysia
lijiajian@stu.xmu.edu.cn

Abstract. Aspect Sentiment Quad Prediction (ASQP) is a crucial sentiment analysis task that has attracted increasing attention. The most recent studies focus on generating complete sentiment quadruples through end-to-end generative models. However, these methods heavily depend on labeled data quality and quantity, performing poorly in low-resource scenarios and less suitable for real-world applications. To overcome these challenges, we introduce a novel Representative Chain-of-Reasoning framework (RCR), with the aim of providing representative knowledge for large language models (LLMs) and fully activating their reasoning capabilities for ASQP. Specifically, we develop a Chain Prompting (ChaPT) module to decompose the ASQP task into three subtasks using the step-by-step reasoning mechanism. Then, a Representative Demonstration Retriever (RepDR) is introduced to provide ChaPT with representative demonstrations, balancing diversity and similarity, and progressively enhancing the reasoning capabilities of LLMs at each step. Experimental results demonstrate the superiority of RCR in low-resource scenarios, with its optimal performance even surpassing that of the fully supervised BERT baseline.

Keywords: Aspect Sentiment Quad Prediction, In-Context Learning, Demonstration Retrieval.

1 Introduction

Given a review text, Aspect Sentiment Quad Prediction (ASQP) aims to predict a comprehensive sentiment view in the form of quadruples[1], each consisting of aspect category, aspect term, opinion term, and sentiment polarity, denoted as (c, a, o, s). For example, given the review sentence, “The food is great and the environment is even better”, the ASQP task requires predicting two sentiment quadruples: (food quality, food, great, positive) and (ambiance general, environment, better, positive). ASQP is a challenging task due to the complexity of sentence structure and the diversity of sentiment expressions, making it difficult to recognize all sentiment quadruples.

Recently, the end-to-end generative models have been extensively applied to solve the ASQP task by generating sentiment quadruples directly from the review text and achieved promising results[2]. A successful application is to construct sequences in natural language format as generation targets, including annotated sentences[3] , paraphrased sentences[1], and sentiment element sequences[4]. Furthermore, sentiment clues within sentences have been utilized to promote the quadruple generation[5]. For example, Gou et al.[6] enhanced the model’s expressive capability by increasing the output views by adjusting the generation order of quadruples. Despite their potential, a notable issue is that these models are less suitable in low-resource scenarios[7]. That is because generative models heavily rely on the scale and quality of the labeled dataset while annotating datasets is costly and time-consuming in practical applications.

With the rise of In-Context Learning (ICL), addressing the ASQP task by generative models in zero-shot and few-shot scenarios becomes feasible[8]. Sun et al.[9] propose a multi-LLM negotiation strategy, demonstrating LLMs’ ability to solve sentiment analysis problems involving complex contexts (e.g., clauses and irony) under zero-shot conditions. Moreover, the Three-Hop Reasoning (THOR) framework[10]achieves state-of-the-art results in implicit sentiment analysis. Despite this, existing research lacks a discussion on applying LLMs to ASQP tasks, and the reasoning capabilities of LLMs are underutilized.

To this end, we propose a Representative Chain-of-Reasoning framework (RCR) that aims to provide representative knowledge for LLMs and fully activate their reasoning capabilities for the ASQP task. Inspired by the Chain-of-Thought (CoT) prompting, we first introduce a Chain Prompt (ChaPT) module to decompose the one-step ASQP task into three sub-steps, where each step progressively infers aspect-opinion pairs, category-aspect-opinion triplets, and complete quadruples. Hence, a complete sentiment view is obtained through step-by-step reasoning, effectively reducing the ASQP task's complexity. Additionally, considering LLM's reasoning capability is greatly influenced by the quality of demonstrations[11], we develop a Representative Demonstration Retriever (RepDR) module to provide ChaPT with representative demonstrations, balancing diversity and similarity, and thus enhancing their reasoning capabilities at each step. Specifically, we first paraphrase the sentiment quadruples into natural sentences and calculate their semantic similarities using SBERT[12]. Based on semantic similarities, the triplet that contains an anchor sentence, a positive sentence, and a negative sentence is picked for further fine-tuning this SBERT model. Hence, this fine-tuned SBERT model is good at retrieving representative demonstrations that possess semantic information of different attributes [13]. To summarize, the primary contributions of this study are as follows:

-This work introduce ChaPT, a prompting framework based on the chain-of-reasoning concept, which mitigates task complexity through task decomposition and step-by-step reasoning, fully leveraging the reasoning capabilities of LLMs.

-This work use the RepDR module to retrieve demonstrations, providing more representative prior information for model reasoning. To the best of our knowledge, this study is the first to propose retrieving both diversity and similarity samples as demonstrations.

2 Method

2.1 Problem Definition

The ASQP task is defined as follows: given a sentence X, the model predicts all aspect-based sentiment quadruples, each formulated as $(\mathrm{c}, \mathrm{a}, \mathrm{o}, \mathrm{s})$ which corresponds to aspect category, aspect term, opinion term, and sentiment polarity, respectively. The aspect category c is part of the predefined category set U_c. The aspect term a is the target entity of opinion. The opinion term o is the subjective statement. Moreover, the sentiment polarity s belongs to the sentiment set $U_s = \{\mathrm{positive}, \mathrm{neutral}, \mathrm{negative}\}$.

Notably, aspect a is generally within the text scope of sentence X, and if aspect a is not explicitly mentioned, it is represented by the specific tag NULL.

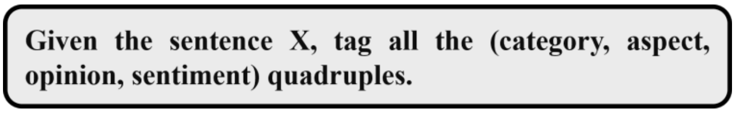
Xie et al.[14] believe that ICL infers conditional probabilities of the predictive target from the prompt, formulated as

$$p(y \mid prompt) = \int_{prompt} p(y \mid prompt)\, d(prompt) \tag{1}$$

where y represents the prediction target and $d(prompt)$ represents the prompt set. ICL infers the maximum probability of generating y by integrating y over the prompt. Therefore, we can use ICL to model the ASQP task as

$$\hat{y} = arg\ max\ P(y \mid X, Prompt) \tag{2}$$

where $\hat{y}$ represents all quadruples, and Zhang et al.[8] attempt to construct the following standard prompt paradigm as input to the LLMs:



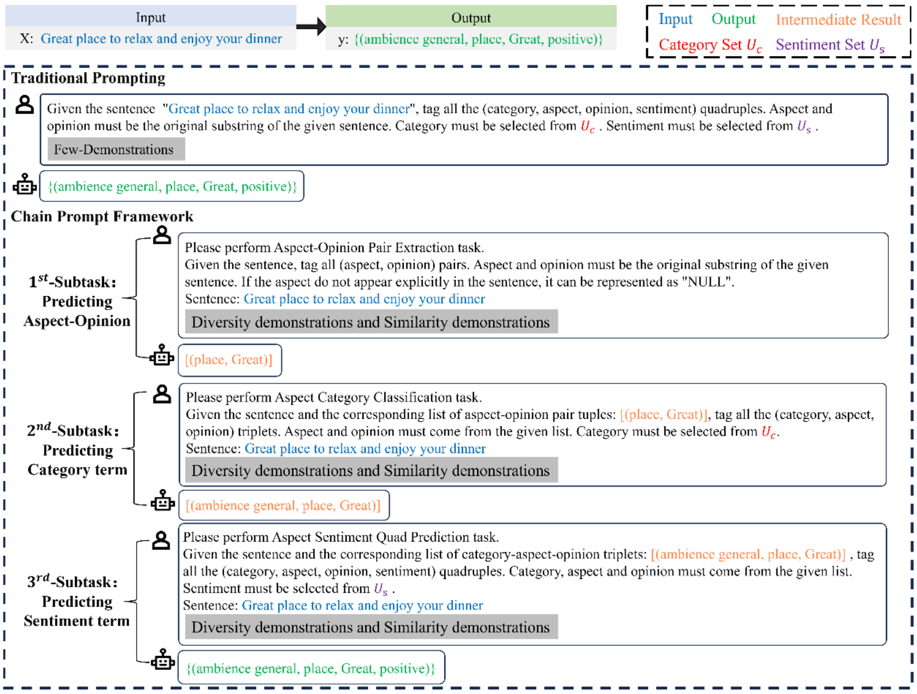
Given the sentence X, tag all the (category, aspect, opinion, sentiment) quadruples.

2.2 Representative Chain-of-Reasoning

To fully activate the reasoning capabilities of LLMs for the ASQP task, we propose a Representative Chain-of-Reasoning framework (RCR) consisting of two sub-modules: Chain Prompt Framework (ChaPT) and Representative Demonstration Retriever (RepDR). The former is designed to decompose the ASQP task into three subtasks, while the latter is responsible for providing representative demonstrations to enhance LLMs’ reasoning capabilities.

2.2.1 Chain Prompt Framework

Inspired by the impressive reasoning capabilities demonstrated by Chain of Thought (CoT) in handling complex tasks, we propose the Chain Prompt framework(ChaPT), shown in Fig. 1, to address the ASQP task by decomposing a one-step ASQP solution into three subtasks. The details are as follows.


Input
X: Great place to relax and enjoy your dinner
Output
y: {(ambience general, place, Great, positive)}
Input
Output
Intermediate Result
Category Set U_c
Sentiment Set U_s
Traditional Prompting
Few-Demonstrations
{(ambience general, place, Great, positive)}
Chain Prompt Framework
1^{st}-Subtask: Predicting Aspect-Opinion
Diversity demonstrations and Similarity demonstrations
[(place, Great)]
2^{nd}-Subtask: Predicting Category term
Diversity demonstrations and Similarity demonstrations
[(ambience general, place, Great)]
3^{rd}-Subtask: Predicting Sentiment term
Diversity demonstrations and Similarity demonstrations
{(ambience general, place, Great, positive)}

Given the sentence "Great place to relax and enjoy your dinner", tag all the (category, aspect, opinion, sentiment) quadruples. Aspect and opinion must be the original substring of the given sentence. Category must be selected from U_c . Sentiment must be selected from U_s .

Please perform Aspect-Opinion Pair Extraction task.
Given the sentence, tag all (aspect, opinion) pairs. Aspect and opinion must be the original substring of the given sentence. If the aspect do not appear explicitly in the sentence, it can be represented as "NULL".
Sentence: Great place to relax and enjoy your dinner

Please perform Aspect Category Classification task.
Given the sentence and the corresponding list of aspect-opinion pair tuples: [(place, Great)], tag all the (category, aspect, opinion) triplets. Aspect and opinion must come from the given list. Category must be selected from U_c.
Sentence: Great place to relax and enjoy your dinner

Please perform Aspect Sentiment Quad Prediction task.
Given the sentence and the corresponding list of category-aspect-opinion triplets: [(ambience general, place, Great)] , tag all the (category, aspect, opinion, sentiment) quadruples. Category, aspect and opinion must come from the given list. Sentiment must be selected from U_s .
Sentence: Great place to relax and enjoy your dinner

Fig. 1. An illustration of our ChaPT framework for Aspect Sentiment Quad Prediction task.

Subtask 1. Aspect-Opinion Pair Extraction. Empirical studies find that extracting a single aspect or opinion alone would ignore their pairwise relationships, leading to pairing errors. Therefore, instead of fine-grained aspect term or opinion term extraction subtask, we first consider predicting all aspect-opinion pairs appearing in the sentence. Mathematically, this subtask is formulated as:

$$\hat{z}_1 = arg\ max\ P(y \mid X, Prompt_1) \tag{3}$$

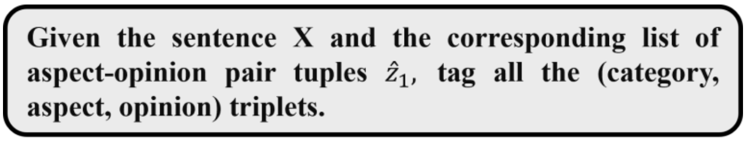
where $\hat{z}_1$ denotes predicted aspect-opinion pairs, the template of $Prompt_1$ is defined:

Given the sentence X, tag all the (aspect, opinion) pairs.

Subtask 2. Aspect Category Classification. Based on X and the intermediate results $\hat{z}_1$, we classify category c from the predefined set U_c and obtain the category aspect-opinion triplets $\hat{z}_2$. This process is represented as:

$$\hat{z}_2 = arg\ max\ P(y \mid X, \hat{z}_1, Prompt_2) \tag{4}$$

the template of $Prompt_2$ is as follows.

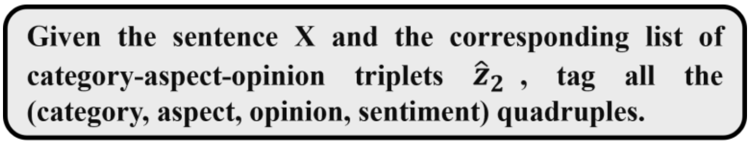


Given the sentence X and the corresponding list of aspect-opinion pair tuples $\hat{z}_1$, tag all the (category, aspect, opinion) triplets.

Subtask 3. Aspect Sentiment Quad Prediction. Based on the intermediate results $\hat{z}_2$, we finally predict the complete quadruples $\hat{y}$. The final step is denoted as:

$$\hat{y} = arg\ max\ P(y \mid X, \hat{z}_2, Prompt_3) \tag{5}$$

and the template of $Prompt_3$ is as follows.



Given the sentence X and the corresponding list of category-aspect-opinion triplets $\hat{z}_2$, tag all the (category, aspect, opinion, sentiment) quadruples.

2.2.2 Representative Demonstration Retriever

In few-shot scenarios, we propose the Representative Demonstration Retriever (RepDR), a demonstration retriever that balances diversity and similarity. First, we explain the generation of labeled sample, then describe training the retriever with the labeled sample, and finally show using the trained retriever to retrieve representative demonstrations. As shown in Fig. 2.

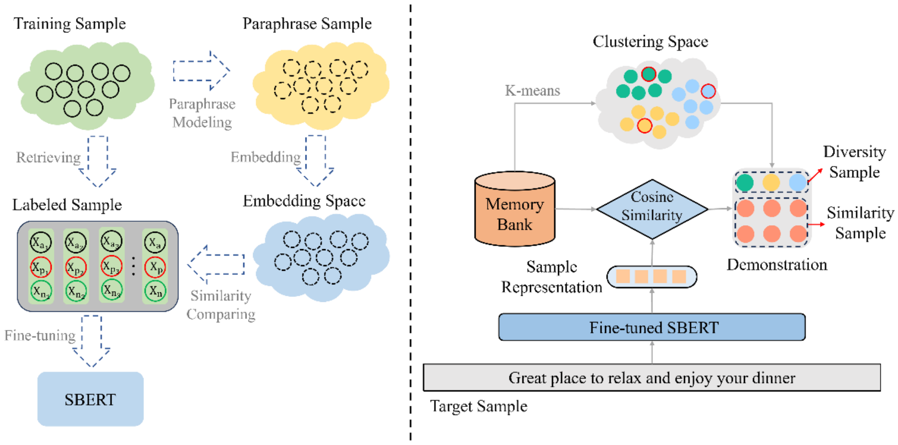
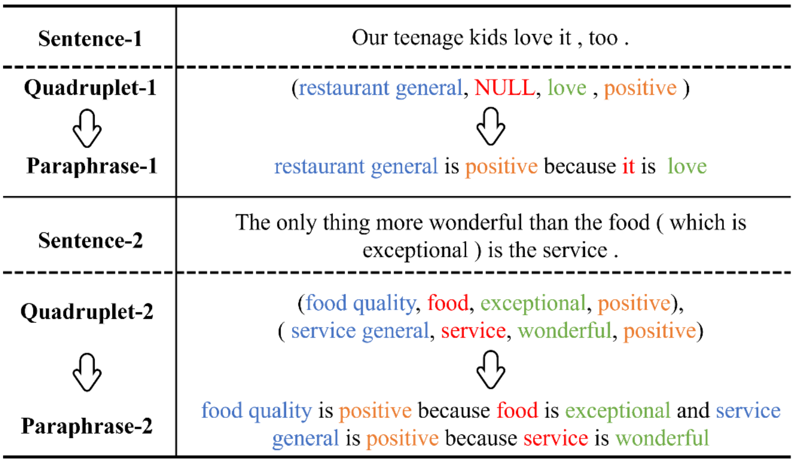


Fig. 2. The proposed RepDR module consists of two stages.

Generating labeled Sample. Since we utilize clustering and similarity comparison for demonstration retrieval, it is vital to train a model that precisely captures the similarity among sentence pairs. We choose SBERT, a BERT-based text embedding model, as our target model. By fine-tuning SBERT with generated triplet data, we enhance its ability to capture semantic similarity between sentences. Triplet data consists of an anchor sentence, a positive sentence, and a negative sentence without additional labels. Inspired by paraphrase generation[1], we propose modeling paraphrases for the training set by linearizing sentiment quadruples (c, a, o, s) into natural sentences I, as shown in Fig. 3.

Paraphrase modeling allows us to focus on the quadruples and ignore unnecessary details in sentences. Subsequently, semantic embeddings of the paraphrase set U_I are computed using the pre-trained SBERT model. Triplet labeled samples (X_a, X_p, X_n) are constructed based on cosine similarity, where X_a denotes the original sentence corresponding to the target paraphrase; X_p is the original sentence corresponding to the most semantically similar paraphrase in U_I; and X_n corresponds to the original sentence of the least similar paraphrase. This design employs a semantic alignment mechanism to encourage the model to focus on the essential features of sentiment quadruples.



Sentence-1	Our teenage kids love it , too .
Quadruplet-1 ⇩ **Paraphrase-1**	(restaurant general, NULL, love , positive) ⇩ restaurant general is positive because it is love
Sentence-2	The only thing more wonderful than the food (which is exceptional) is the service .
Quadruplet-2 ⇩ **Paraphrase-2**	(food quality, food, exceptional, positive), (service general, service, wonderful, positive) ⇩ food quality is positive because food is exceptional and service general is positive because service is wonderful

Fig. 3. Two examples of paraphrase modeling. Notably, if the aspect is not explicitly mentioned, it is represented by the implicit pronoun "it".

Training Retriever. We fine-tune the SBERT model using the typical triplet network. Specifically, given a triplet (X_a, X_p, X_n) with corresponding embedding vectors (O_a, O_p, O_n), the fine-tuning objective is to minimize the distance between O_a and O_p, while maximizing the distance between O_a and O_n. We use triplet loss as the loss function, as shown below in Equation 6:

$$\mathcal{L}(O_a, O_p, O_n) = max\left(d\left(O_a, O_p\right) - d(O_a, O_n) + \alpha, 0\right) \tag{6}$$

where $d\left(O_a, O_p\right) = \left\|O_a - O_p\right\|_2$ represents the Euclidean Distance between embeddings. The hyperparameter α specifies the expected difference between $d\left(O_a, O_p\right)$ and $d(O_a, O_n)$.

Retrieving Demonstration. Firstly, we use the fine-tuned SBERT to encode all training set sentences into text embeddings and store them in a Memory bank[15] to avoid redundant computations. Secondly, we measure the similarity of demonstrations utilizing Cosine Similarity, comparing target samples with the training set to extract the top-k most similar samples. Finally, we propose a diversified demonstration retrieval scheme based on K-means clustering. We evaluate the optimal number of clusters by calculating the Silhouette Score, and find that the optimal number for both datasets is

3. Based on this result, we perform K-means clustering and select the samples closest to the cluster centers as diversity samples that highlight key characteristics.

3 Experiments

3.1 Datasets

Experiments were conducted on two publicly available restaurant datasets, Rest15 and Rest16, from the SemEval task[16]. These datasets, with multiple annotations[17], were aligned by Zhang et al.[1] and ultimately served as the standard datasets for the ASQP task. Each sample contains one or more sentiment quadruples. The dataset statistics are shown in Table 1.

Table 1. Dataset statistics for Rest15 and Rest16.

	Rest15				**Rest16**			
	#+	**#0**	**#-**	**#S**	**#+**	**#0**	**#-**	**#S**
Train	1005	34	315	834	1369	62	558	1264
Dev	252	17	81	209	341	23	143	316
Test	453	37	305	537	583	40	176	544

3.2 Implementation Details

We utilized Llama-3.1[1](Llama-3.1-70B), GPT-3.5 (gpt-3.5-turbo3[2]), and the newly released GTP-4o [3] as the backbone for the ChaPT framework to evaluate its effectiveness under zero-shot conditions. The temperature for all models was set to 0 to ensure stable predictions.

[1] https://llama.meta.com/

[2] https://github.com/openai/openai-python

[3] https://chatgpt.com/

Moreover, for few-shot scenarios, we employed all-mpnetbase-v1[4] as pre-trained SBERT, using a typical triplet network for fine-tuning. During fine-tuning, we used a batch size of 64, a learning rate of 2e-5, and 5 training epochs. The hyperparameter α of the model was set to 5. The training of generative models and SBERT in low-resource scenarios follows the settings proposed by Zhang et al. [8], where k-shot represents sampling k examples for each aspect category. We set the batch size of all models to 8, the learning rate to 1e-4, and the training epochs to 100. All experiments were conducted using an NVIDIA RTX 3090 GPU.

During demonstration retrieval, we used the model at the best checkpoint to re-encode sentences for text similarity comparison and clustering, obtaining demonstrations with similarity and diversity. We only considered three k-shot settings: 1-shot, 5-shot, and 10-shot. For each setting, we maintained a constant number of diversity demonstrations and adjusted the number of similarity demonstrations to achieve k-shot. For example, in the 1-shot scenario, we retrieved 3 diversity samples and the 1 most similar sample.

3.3 Baselines

We employ supervised learning models and ICL-based large language models as our comparative baselines. For the supervised learning models, based on architectural differences, they can be broadly classified into BERT-based discriminative methods and T5-based generative methods.

The BERT-based discriminative methods include:

HGCN. A joint extraction framework proposed by Cai et al. [18], which jointly models the extraction of aspect categories and the identification of sentiment polarities. Then, Li et al. [19] employ a BERT encoder to extract aspect and opinion terms.

TASO[20]. An enhanced framework that extends the original triplet extraction (TAS) paradigm to support quadruple extraction.

[4] https://huggingface.co/sentence-transformers/all-mpnet-base-v1

Extract-Classify. A two-stage method proposed by Cai et al.[21], where aspect and opinion terms are first extracted from the original sentence, followed by classification to determine the aspect category and sentiment polarity.

The T5-based generative methods include:

GAS[3]. The first attempt to use generative methods to handle aspect-based sentiment analysis, we modify it to use sentiment quadruple sequences as target sequences.

Paraphrase[1]. A paraphrasing modeling framework, using paraphrased sentences as training targets to generate sentiment quadruples end-to-end.

DLO/ILO[7]. Selecting the appropriate quadruple generation order as a data augmentation method for paraphrase generation.

MvP[6]. Enhances the model predictive capability by increasing output views of different quadruple generation orders.

For ICL methods, we selected the following research approaches:

LMMs for SA [8]. A comprehensive study of sentiment analysis using LLMs, including Flan-T5, FLanUL2, T5, and GPT-3.5. Since it does not involve a specific ICL design, this method can serve as a direct baseline for using LLMs alone.

THOR[10]. A Three-hop reasoning (THOR) CoT framework for addressing implicit sentiment analysis issues. In this study, its prompt templates are adaptively modified to function as a comparative baseline for CoT.

The experimental results for these supervised methods are obtained based on the respective pre-trained models (BERT or T5-base) to ensure a fair comparison.

3.4 Zero-shot and Few-shot Results

The experimental results are shown in Table 2. Notably, due to the significantly inferior performance of the BERT-based model in low-resource scenarios relative to the baseline models, it was not included in the primary comparison. In the zero-shot scenario, the T5-base generative model struggled with ASQP tasks and failed to generate effective results. However, the performance gradually improved as the number of samples increased, highlighting the importance of high-quality labeled data for generative models.

Table 2. The experimental results under zero-shot and few-shot.

Methods	**Rest15**				**Rest16**			
	0-shot	**1-shot**	**5-shot**	**10-shot**	**0-shot**	**1-shot**	**5-shot**	**10-shot**
GAS	-	4.43	10.65	13.82	-	2.24	16.04	19.03
Paraphrase	-	7.78	11.53	18.60	-	2.36	12.85	16.34
DLO	-	6.79	13.07	18.92	-	1.90	17.68	28.95
ILO	-	7.25	14.85	20.99	-	2.41	15.71	21.32
MvP	-	9.33	18.54	22.82	-	3.62	21.51	29.24
w/ Llama-3.1								
LMMs for SA	18.20	20.79	27.81	32.84	29.01	30.99	35.04	37.10
THOR	17.62	22.49	26.78	31.47	27.61	30.04	34.34	35.91
RCR	19.37	26.32	8.50	33.14	28.19	33.12	38.11	39.96
w/ GPT-3.5								
LMMs for SA	7.77	27.83	26.86	25.74	10.06	28.45	38.63	37.13
THOR	10.21	22.13	27.04	23.51	14.11	26.62	37.26	36.04
RCR	13.82	28.46	31.44	28.59	19.00	30.60	41.51	42.09
w/ GPT-4o								
LMMs for SA	32.40	35.63	37.65	36.72	35.87	40.56	42.07	40.32
THOR	30.57	35.01	36.22	35.81	36.37	38.02	41.58	39.97
RCR	**33.01**	**39.78**	**42.26**	**44.33**	**38.07**	**40.97**	**48.08**	**51.23**

Compared to the best generative model baseline MvP, the ICL-based LLMs (LLMs for SA) showed significant performance improvements in both zero-shot and few-shot scenarios. For GPT-3.5, under few-shot conditions, the average F1 score gained on the Rest15 and Rest16 datasets was 9.91% and 16.25%, respectively. This demonstrates the great potential of LLMs in ASQP tasks. Furthermore, our proposed Representative Chain-of-Reasoning (RCR) framework achieved the best performance with Llama-3.1, GPT-3.5 and GPT-4o compared to the original ICL baselines. Specifically, with Llama-3.1, the average F1 scores gained on the Rest15 and Rest16 datasets were 1.92% and 1.81%, respectively. With GPT-3.5, the average F1 scores increased by 3.53% and 4.73%. With GPT-4o, the F1 scores improved by an average of 4.24% and 4.88%. This indicates that the RCR framework provides sufficient prior information for LLMs, fully leveraging their reasoning capabilities in ASQP tasks.

Table 3. The results of ablation study.

Methods	**Rest15**			**Rest16**		
	Pre	**Rec**	**F1**	**Pre**	**Rec**	**F1**
RCR	**27.99**	**35.85**	**31.44**	**36.82**	**47.56**	**41.51**
RCR w/o [RepDR]	23.39	34.21	27.78	24.53	32.42	27.92
RCR w/o [ChaPT]	21.83	27.30	24.26	28.16	36.29	31.71
RCR w/o [RepDR,ChaPT]	21.25	26.68	23.66	25.10	30.16	27.40

3.5 Ablation Study

We conducted ablation experiments to further validate our RCR framework’s effectiveness. In a 5-shot scenario, we analyzed the impact on the results by removing individual modules, with results shown in Table 3. ChaPT decomposes the ASQP task into subtasks, reducing the complexity of LLM reasoning. RepDR is responsible for providing more accurate prior semantic knowledge to LLMs through demonstration retrieval. The results indicate that removing any module significantly reduces RCR performance, demonstrating the effectiveness of ChaPT and RepDR in stimulating the reasoning capabilities of LLMs. Furthermore, We observed performance differences across the Rest15 and Rest16 datasets when removing specific modules. For instance, removing the RepDR module resulted in a 13.59% decrease in F1 score for Rest16, but only a 3.66% decrease for Rest15. This indicates that different datasets have varying dependencies on the ChaPT and RepDR modules, reflecting the distinct knowledge support these two components provide to LLMs.

Table 4. Comparison results with supervised learning baselines.

Model	**Methods**	**Rest15**			**Rest16**		
		Pre	**Rec**	**F1**	**Pre**	**Rec**	**F1**
BERT	HGCN-TFM	25.55	22.01	23.65	27.4	26.41	26.9
	HGCN-Linear	24.43	20.25	22.15	25.36	24.03	24.68
	TASO-CRF	44.24	26.34	34.78	48.65	39.68	43.71
	TASO-Linear	41.86	26.5	32.46	49.73	40.7	44.77
	Extract-Classify	35.64	37.25	36.42	38.4	50.93	43.77
T5	GAS	45.31	46.7	45.98	54.54	57.62	56.04
	PARAPHRASE	46.16	47.93	47.03	56.63	59.37	57.91
	ILO	47.78	50.38	50.35	57.58	61.17	59.32
	DLO	47.08	43.93	48.18	57.92	61.8	59.79
	MvP	-	-	**51.04**	-	-	**60.39**
GPT-4o	LMMs for SA	36.25	39.18	37.65	39.86	44.54	42.07
	THOR	34.3	38.37	36.22	39.27	44.18	41.58
	RCR	42.00	46.92	44.33	48.18	54.69	51.23

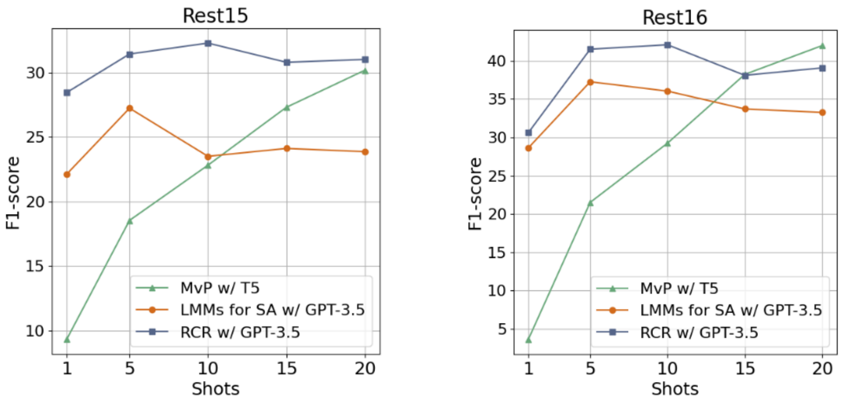
3.6 Performance Comparison with Supervised Learning Baselines

We compare the best performance of the ICL method under GPT-4o with the results of the supervised learning baselines. The comparison results are shown in the Table 4. Experimental results indicate a significant performance gap between ICL-based methods and generative models, suggesting that training generative models with domain-labeled data remains the best approach for modeling ASQP tasks at this stage. However, it is noteworthy that the proposed RCR framework demonstrates significantly better performance in low-resource scenarios compared to the fully fine-tuned BERT baseline. When compared with the Extract-Classify method, the F1 score increases by 7.91% and 7.46% on the Rest15 and Rest16 datasets, respectively. This strongly demonstrates the substantial application potential of the RCR framework for ASQP tasks in low-resource settings.

3.7 Influence of Different Sample Sizes

Our preliminary research reveals that the LLMs' reasoning capabilities for ASQP tasks improve significantly with an increased sample size. However, this raises the question of whether this improvement is always directly proportional to the number of samples. To explore this issue, we further increased the sample size, as shown in Fig. 4.

We found that the T5-large MvP model's performance steadily improved with more samples, indicating that the generative-based models rely on sufficient high-quality labeled data. Surprisingly, for ICL-based methods, performance tends to decline after reaching a certain sample size threshold. Our analysis suggests two main reasons for this decline. First, a large number of examples provides excessive prior semantic information, causing LLMs to become confused and lose focus on core aspects. Second, lower-ranked samples are poorer in quality and contain more redundancy. Notably, compared to previous ICL methods, the RCR framework mitigates this performance degradation, indicating that RepDR retrieves higher-quality demonstrations and introduces fewer errors.



(a) Rest15 result (b) Rest16 result

Fig. 4. The evaluation curve of the model with varying sample sizes.

4 Conclusion

In this paper, we introduce a novel RCR framework designed to solve the ASQP task in low-resource scenarios. To reduce complexity, the chain prompting module (ChaPT) is designed to decompose the ASQP task into three subtasks and enable LLMs to conduct step-by-step reasoning. Furthermore, a representative demonstration retriever (RepDR) is developed to provide ChaPT with demonstrations that balance diversity and similarity, maximizing the reasoning ability of LLMs at each step. Comprehensive experimental results substantiate the efficacy of the proposed RCR framework in both zero-shot and few-shot settings, leading to GPT-4 achieving state-of-the-art performance on the ASQP task in low-resource scenarios.

Acknowledgments. This study was supported by the Industry-University-Research Subsidy Projects of Xiamen City, Fujian Province, China (2023CXY0111 and 2024CXY0102).

References

1. Zhang W, Deng Y, Li X, et al. Aspect sentiment quad prediction as paraphrase generation[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. 2021: 9209-19.
2. Peper J, Wang L. Generative aspect-based sentiment analysis with contrastive learning and expressive structure[C]//Findings of the Association for Computational Linguistics: EMNLP 2022. 2022: 6089-95.
3. Zhang W, Li X, Deng Y, et al. Towards generative aspect-based sentiment analysis[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. 2021: 504-10.
4. Zhang Y, Guo Q, Kordjamshidi P. Towards navigation by reasoning over spatial configurations[C]//Proceedings of Second International Combined Workshop on Spatial Language Understanding and Grounded Communication for Robotics. 2021: 42-52.
5. Bao X, Wang Z, Jiang X, et al. Aspect-based sentiment analysis with opinion tree generation[C]//IJCAI. 2022, 2022: 4044-50.
6. Gou Z, Guo Q, Yang Y. Mvp: Multi-view prompting improves aspect sentiment tuple prediction[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 4380-97.
7. Hu M, Wu Y, Gao H, et al. Improving aspect sentiment quad prediction via template-order data augmentation[C]//Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. 2022: 7889-900.
8. Zhang W, Deng Y, Liu B, et al. Sentiment analysis in the era of large language models: A reality check[C]//Findings of the Association for Computational Linguistics: NAACL 2024. 2024: 3881-906.
9. Sun X, Li X, Zhang S, et al. Sentiment analysis through llm negotiations [J]. arXiv preprint arXiv:231101876, 2023.
10. Fei H, Li B, Liu Q, et al. Reasoning implicit sentiment with chain-of-thought prompting[C]//Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics. 2023: 1171-82.

11. Wang X, Zhu W, Wang W Y. Large language models are implicitly topic models: Explaining and finding good demonstrations for in-context learning [J]. arXiv preprint arXiv:230111916, 2023, 1: 15.
12. Reimers N, Gurevych I. Sentence-bert: Sentence embeddings using siamese bert-networks[C]//Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. 2019: 3982-92.
13. Shi Z, Wei J, Xu Z, et al. Why larger language models do in-context learning differently?[C]//International Conference on Machine Learning. PMLR, 2024: 44991-5013.
14. Xie S M, Raghunathan A, Liang P, et al. An explanation of in-context learning as implicit bayesian inference [J]. arXiv preprint arXiv:211102080, 2021.
15. Wu Z, Xiong Y, Yu S X, et al. Unsupervised feature learning via non-parametric instance discrimination[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 3733-42.
16. Pontiki M, Galanis D, Papageorgiou H, et al. Semeval-2016 task 5: Aspect based sentiment analysis[C]//International workshop on semantic evaluation. 2016: 19-30.
17. Peng H, Xu L, Bing L, et al. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34: 8600-7.
18. Cai H, Tu Y, Zhou X, et al. Aspect-category based sentiment analysis with hierarchical graph convolutional network[C]//Proceedings of the 28th international conference on computational linguistics. 2020: 833-43.
19. Li X, Bing L, Zhang W, et al. Exploiting bert for end-to-end aspect-based sentiment analysis[C]//Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019). 2019: 34-41.
20. Wan H, Yang Y, Du J, et al. Target-aspect-sentiment joint detection for aspect-based sentiment analysis[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34: 9122-9.
21. Cai H, Xia R, Yu J. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions[C]//Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing. 2021: 340-50.