CACA: Context-Aware Cross-Attention Network for Extractive Aspect Sentiment Quad Prediction

Bingfeng Chen[1,2], **Haoran Xu**[1], **Yongqi Luo**[1], **Boyan Xu**[1*], **Ruichu Cai**[1], **Zhifeng Hao**[1,3]
[1]School of Computer Science, Guangdong University of Technology
[2]Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ)
[3]College of Science, Shantou University
chenbf@gdut.edu.cn
{ml319xhr, lyongqi001, hpakyim, cairuichu}@gmail.com
haozhifeng@stu.edu.cn

Abstract

Aspect Sentiment Quad Prediction(ASQP) enhances the scope of aspect-based sentiment analysis by introducing the necessity to predict both explicit and implicit aspect and opinion terms. Existing leading generative ASQP approaches do not modeling the contextual relationship of the review sentence to predict implicit terms. However, introducing the contextual information into the pre-trained language models framework is non-trivial due to the inflexibility of the generative encoder-decoder architecture. To well utilize the contextual information, we propose an extractive ASQP framework, CACA, which features with *Context-Aware Cross-Attention Network*. When implicit terms are present, the *Context-Aware Cross-Attention Network* enhances the alignment of aspects and opinions, through alternating updates of explicit and implicit representations. Additionally, contrastive learning is introduced in the implicit representation learning process. Experimental results on three benchmarks demonstrate the effectiveness of CACA. Our implementation will be open-sourced at https://github.com/DMIRLAB-Group/CACA.

1 Introduction

Aspect Sentiment Quad Prediction (ASQP) (Cai et al., 2021a; Zhang et al., 2021a) is a fine-grained text sentiment analysis technique that extracts sentiment information from text, including aspect, opinion, category and sentiment polarity. Compared with the Aspect Sentiment Triple Extraction (ASTE) task (Peng et al., 2020; Wan et al., 2020), the ASQP task introduces the necessity to predict both explicit and implicit aspect and opinion terms, which is more aligned with practical needs and more challenging.

As shown in Figure 1, in the given sentence, we can easily extract the quadruples that are explicitly

Review Sentence:
It ' s **faster** , the **screen** is much **nicer** , and it has twice the **memory** and four times the **drive space** .

Quadruple Extraction:
QUAD#1 (screen - design_features - nicer – positive)
QUAD#2 (drive space - design_features - null – positive)
QUAD#3 (memory - design_features - null – positive)
QUAD#4 (null - operation_performance - faster – positive)

Figure 1: An example of ASQP Task's input and output. From left to right, they represent aspect term, aspect category, opinion term, and sentiment polarity respectively. 'NULL' represents the implicit target term.

present and obtain one target tuple {screen, nicer, Design_features, Positive}. However, in addition to this explicit quadruple, there are some implicit quadruples that need to be extracted in combination with complex contextual information, like {NULL, Operation_performance, faster, Positive} which the aspect term is implicit. These samples with implicit targets need to be predicted through more complex contextual relationships, which places higher requirements on the model's context understanding ability.

Leading generative-based ASQP approaches adopt the generative paradigm, constructing training samples from corresponding quadruples to fine-tune pre-trained language models under encoder-decoder architectures such as T5 (Raffel et al., 2020). For instance, Zhang et al. (2021b,c) constructs predefined templates to convert quadruples into natural language text to fine-tune the pre-train language model. Gou et al. (2023) generates quadruples by defining multiple templates in different views, allowing the generative model to learn the connections between different elements without changing the encoder-decoder structure in some extent.

Although promising results have been reported, we observe that they are still prone to incorrect predictions on challenging samples containing implicit

*Corresponding author, hpakyim@gmail.com

Proceedings of the 31st International Conference on Computational Linguistics, pages 9472–9484
January 19–24, 2025. ©2025 Association for Computational Linguistics

targets. Therefore, we conclude that existing approaches in the ASQP task leading to the following two main limitations: (i) Generative frameworks based on pre-trained models rely on a unified learning process, which cannot effectively distinguish between learning explicit and implicit targets. As shown in the Figure 1 for the case of implicit terms, there can be various combinations of missing entities, with QUAD#2 and QUAD#3 missing opinion terms, and QUAD#4 missing aspect terms. Relying solely on generative models may fail to handle these scenarios, leading to incorrect predictions. (ii) Existing solutions cannot effectively learn the alignment between aspect terms and opinion terms in cases where terms are missing. This deficiency prevents them from constructing a substantial connection between implicit and explicit terms, significantly affecting performance on such data in the ASQP task.

In this paper, we propose an extractive ASQP framework CACA, *Context-Aware Cross-Attention Network*. To well handle the implicit targets, our CACA introduce a Implicit Target Extraction module, which not only incorporates certain explicit target contextual information but also uses contrastive learning to differentiate between the representations of explicit and implicit targets. Meantime, we can extract implicit representations with complex contextual information from this module for use in other sentiment elements extraction. Next, we design a cross-attention mechanism to align aspect terms and opinion terms from both directions. On one hand, we can model the relationship between explicit aspect terms and opinion terms. On the other hand, when one of them is implicit, we can use the implicit representations learned from Implicit Target Extraction module to accurately capture other sentiment elements. As the QUAD#1 and QUAD#2 shown in Figure 1, we can not only effectively align "screen" and "nicer", but also using the learned implicit opinion representation to align with "drive space".

To sum up, our proposed CACA framework makes the following contributions:

- In contrast to leading generative paradigm for ASQP, our proposed CACA network is based on extraction and integrates the cross-attention mechanism to effectively align aspect terms and opinion terms.
- For the implicit targets, we devise an Implicit Target Extraction module which introduces contrastive learning to make explicit targets representations more distinguishable from implicit targets representations.
- By integrating the learned implicit representations with the CACA network, we significantly improved the performance on the implicit data subset in the ASQP task.
- We demonstrated the effectiveness of our CACA through extensive experiments, and it outperforms the latest baselines on three benchmarks.

2 Methodology

The overall architecture of CACA is shown in Figure 2, our network mainly consists of the following five parts: *Span Generation*, *Explicit Target Extraction*, *Implicit Target Extraction*, *Relation Alignment using Cross-Attention* and *Result Inference*. Next, we will describe each one in detail.

2.1 Span Generation

For a input sentence $X = \{w_1, w_2, ..., w_n\}$, we employ the pretrained language model to serve as the contextual encoder to obtain the base contextual representation $H = \{h_1, h_2, ..., h_n\} \in R^{n\times d}$, where n is the length of the given sentence and d is the dimension of word representations. Then, we employ the slide window to obtain span-level representation with a maximum length span of L. The span representation from i to j words calculates as follows:

$$s'_{i,j} = h_i \oplus h_j \oplus Mean(h_i : h_j) \oplus (h_j - h_i) \quad (1)$$

$$s_{i,j} = W_s s'_{i,j} \quad (2)$$

where $Mean$ represents mean pooling. $\oplus$ denotes the concatenation operation. $h_i : h_j (i \leq j)$ is a sentence segment of $[h_i, h_{i+1}, ..., h_j]$ and $j-i \leq L$. $W_s \in R^{4d\times d}$ is the trainable paramters.

However, the span generated by the slide window contains a large amount of noise, which dilutes meaningful span information and hinders the model's learning. Therefore, we devise a pruning rule to remove irrelevant spans. For detail, we use the Natural Language Toolkit (NLTK[1]) to prune spans for each generated span that mainly meet the following conditions: (i) The length of stop words exceeds half of the span length. (ii) The span does not contain at least one adjective, noun, or adverb.

[1] https://www.nltk.org/

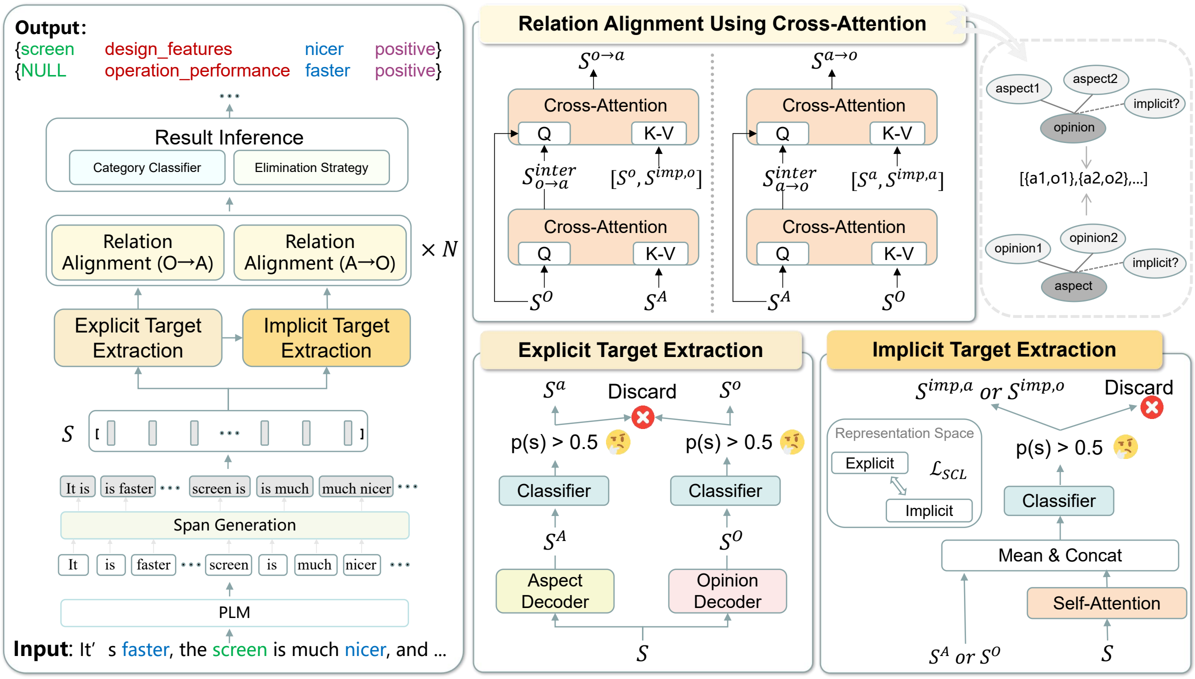

Figure 2: Overview of Context-Aware Cross-Attention Network. On the left side of the figure is the workflow of the CACA network, while the right side details the specific implementations of the three sub-modules: *Explicit Target Extraction*, *Implicit Target Extraction* and *Relation Alignment using Cross-Attention*.

After applying the pruning strategy, only M spans are left out and the final span representation can be expressed as:

$$S = \{S_1, S_2, \cdots, S_i, \cdots, S_M\} \tag{3}$$

where i represents the i^{th} representation of the span representation S.

2.2 Explicit Target Extraction

This section focuses on the extraction of explicit targets (aspect terms and opinion terms) and the extracted results will be stored in two sets: aspect term set Set^A and opinion term set Set^O. Firstly, we construct two decoders with the same structure to extract explicit targets. The aspect decoding process can be described as follows:

$$S' = Dropout(W_{down} \cdot \sigma(W_{up} \cdot S)) \tag{4}$$

$$S^A = LayerNorm(S' + S) \tag{5}$$

where W_{down} and W_{up} are the trainable parameters. σ is the activation function. Then, the probability distribution of each span is assessed:

$$p^a = Softmax(W_a S^A) \tag{6}$$

where $W_a \in \mathbb{R}^{M \times T}$, and T $\in$ {Valid, Invalid} is the set of target classes. Finally, we can obtain the valid aspect set Set^A which contain the span representations of explicit aspect terms.

Similarly, we can decode opinion terms using the same structure but with different parameters to obtain the opinion decoder embeddings S^O, the probability distribution p^o and the valid opinion set Set^O (refer Appendix A.1 for details). In this section, the loss is defined as:

$$\mathcal{L}_{A\&O} = -\sum_i y_i^a \log(p_i^a) - \sum_i y_i^o \log(p_i^o) \tag{7}$$

where y_i^a and y_i^o are the true labels of explicit aspect terms and opinion terms.

2.3 Implicit Target Extraction

In this section, we need to predict whether the sentence contains implicit targets. We capture the association between spans through self-attention mechanisms.

$$S^{context} = Softmax(\frac{QW_q^c(KW_k^c)^T}{\sqrt{d_k}}) \cdot VW_v^c \tag{8}$$

where Q=K=V=S and W_q^c, W_k^c, W_v^c are the trainable parameters. d_k represents the scaling factor. Depending on the type of prediction task, we integrate different target decoders. Then we perform binary classification on the sentence to determine the presence of implicit targets.

To ensure that the representations of sentences with implicit targets are farther apart from those without implicit targets in the representation space, we introduce Supervised Contrastive Learning (Gunel et al., 2020). Specifically, within the same batch, texts containing implicit targets are treated as positive samples, while those without implicit targets are considered negative samples. The process can be represented as:

$$S^{imp,a} = FNN_a(Concat(Mean(S^{context}, S^A)) \quad (9)$$

$$p^{imp,a} = Softmax(W^a_{imp} S^{imp,a}) \quad (10)$$

where $W^a_{imp} \in R^{b\times 2}$. For each sample in mini-batch B, our implicit contrastive learning loss for aspect is defined as:

$$\mathcal{L}^a_i = -\frac{1}{M(i)} \sum_{j\in M(i)} log \frac{exp(sim(u^a_i, u^a_j)/\tau)}{\sum_{k=1}^{B} exp(sim(u^a_i, u^a_k)/\tau)} \quad (11)$$

where $M(i)$ denotes the set of examples with the same label and k $\neq$ i. u^a_i is the i^{th} representation of $S^{imp,a}$, τ is the temperature coefficient. $sim(u_i, u_j)$ is the formula for calculating cosine similarity.

Similarly, we can use the same approach to predict whether a sentence has implicit opinion terms (refer Appendix A.2 for details). And the total loss for *Implicit Target Extraction* module can be defined as:

$$\mathcal{L}_{IMP} = \mathcal{L}_{imp,a\&o} + \alpha \mathcal{L}^a_{CL} + \beta \mathcal{L}^o_{CL} \quad (12)$$

$$\begin{aligned} \mathcal{L}_{imp,a\&o} = &-\sum_i y_i^{imp,a} \log (p_i^{imp,a}) \\ &-\sum_i y_i^{imp,o} \log (p_i^{imp,o}) \end{aligned} \quad (13)$$

$$\mathcal{L}^a_{CL} = \sum \mathcal{L}^a_i, \mathcal{L}^o_{CL} = \sum \mathcal{L}^o_i \quad (14)$$

where α and β are hyper parameters. $y_i^{imp,a}, y_i^{imp,o}$ are the ground truth labels whether the sentence has implicit targets. For sentences that the model identifies as containing implicit targets, we save their representations for subsequent extraction of other sentiment elements as shown in Figure 2.

2.4 Relation Alignment Using Cross-Attention

In Section 2.2 and 2.3, we obtain the explicit and implicit representation. The key to enhancing prediction performance lies in effectively aligning and learning the relationship between aspect terms and opinion terms, especially the explicit and implicit alignment level. Therefore, we align their relation from two directions based on cross-attention.

For the direction of aspect to opinion, we use a cross-attention mechanism to incorporate S^O as supplementary information, which helps in better capturing the opinion terms corresponding to the specific aspect terms:

$$S^{inter}_{a\to o} = Softmax(\frac{S^A W^1_q (S^O W^1_k)^T}{\sqrt{d_k}}) \cdot S^O W^1_v \quad (15)$$

$$S^{A'} = W^1 \cdot (LayerNorm(S^A + S^{inter}_{a\to o}) \quad (16)$$

where W^1_q, W^1_k, W^1_v and W^1 are the trainable parameters. Then we fuse the Set^A from Section 2.2 into $S^{A'}$ by a Cross-Attention Block to obtain the opinion span most relevant to a specific aspect.

$$A^{a\to o} = Softmax(\frac{S^{A'} W^2_q (Set^A W^2_k)^T}{\sqrt{d}}) \quad (17)$$

$$q^{a\to o} = W_{a\to o}(A^{a\to o} \cdot S^{A'} W^2_v) \quad (18)$$

$$p(o|a) = Softmax(q^{a\to o}) \quad (19)$$

where $W_{o\to a} \in R^{M\times s}$, and $s \in$ {Positive, Negative, Neural, Invalid} is the sentiment class of every span. $p(o|a)$ is a probability distribution indicating the likelihood of all possible opinion terms given the aspect term.

When the existence of implicit aspect terms is confirmed in Section 2.3, we utilize the implicit representations $S^{imp,a}$ to effectively align the relationship with the opinion terms as illustrated on the right side of the module in Figure 2. For the issue of missing both aspect and opinion terms, we set implicit placeholders in the sequence, so that the model can also align the relationship between them by an complex contextual semantic understanding.

For the opinion to aspect direction, we calculate $q^{o\to a}$ and $p(a|o)$ in the same manner (refer Appendix A.3 for details). We can define the loss of this section:

$$\begin{aligned} \mathcal{L}_{A\leftrightarrow O} = &-\lambda \sum_i \sum_j y^{a\to o} \log (p^{a\to o}) \\ &-(1-\lambda) \sum_i \sum_j y^{o\to a} \log (p^{o\to a}) \end{aligned} \quad (20)$$

where $y^{a\to o}$ and $y^{o\to a}$ are the true labels for the sentiment of every span. λ is the loss weight for the two directions.

2.5 Result Inference

Category Classifier For aspect categories, we abandon the method of predicting categories solely based on aspect terms. Instead, we utilize representations formed by concatenating aspect terms and opinion terms to predict the category for each tuple. This method can not only be applied to multi-category aspect terms, but also well reflect their category information through opinion terms when the aspect terms are implicit. In such implicit cases, the opinion term can better reflect the category information of the quadruple.

Given the previous Set^A and Set^O, we perform a similar Cartesian product to classify each tuple:

$$p^c_{i,j} = FNN_c(Concat(Set^A_i, Set^O_j)) \tag{21}$$

where i and j represent the i^{th} aspect term and j^{th} opinion term, $W_c \in R^{k \times c}$, k is the number of combinations of aspect terms and opinion terms, c is the number of category classes. The loss of this subsection can be defined as:

$$\mathcal{L}_C = -\sum_i \sum_j y^c_{i,j} \log\left(p^c_{i,j}\right) \tag{22}$$

where $y^c_{i,j}$ is the true labels of the aspect category. Then we can obtain the (a,o,c) set as V_c.

Elimination Strategy After generating (a, o, s) pairs based on the Section 2.4, tuple conflicts (i.e., the same a, o but different s) inevitably arise. To address the issue of model consistency, previous methods often resolved conflicts by selecting results with higher sentiment probabilities.

$$V' = \begin{cases} (a, o, s_{a\to o}) & s_{a\to o} > s_{o\to a} \\ (a, o, s_{o\to a}) & s_{a\to o} < s_{o\to a} \end{cases} \tag{23}$$

where $s_{a\to o}$ and $s_{o\to a}$ represent the sentiment polarity probabilities in both directions, respectively.

However, they did not consider the impact of erroneous answers caused by sample uncertainty and noise, which also affects performance. Therefore, we set an additional confidence threshold to eliminate incorrect or uncertain answers from the model. The threshold calculation formula is as follows:

$$p(a, o) = \begin{cases} p(a)p(o|a) & if\ a \to o \\ p(o)p(a|o) & if\ o \to a \end{cases} \tag{24}$$

$$V'' = \{(a, o, s) \mid p(a, o) > \delta\} \tag{25}$$

$$V_s = V' \cap V'' \tag{26}$$

where δ represents the threshold. V'' is the set obtained by threshold filtering from two directions. V_s is the (a,o,s) set after elimination. Then,we can merge sets V_c and V_s to finally obtain the (a, o, c, s) quad set V_{quad}.

2.6 Training

The training objective is to minimize the total loss function, which is defined as follows:

$$\mathcal{L}(\Theta) = \mathcal{L}_{A\&O} + \mathcal{L}_{IMP} + \mathcal{L}_{A\leftrightarrow O} + \mathcal{L}_C \tag{27}$$

where Θ denotes all trainable parameters of the model. The four losses originate from the four main components of our model.

3 Experiments

3.1 Set up

Datasets To validate the effectiveness of our model, we conducted experiments separately on Restaurant, Laptop datasets (Cai et al., 2021b) and Phone dataset (Zhou et al., 2023). The last dataset has the larger size, more words per sample and higher density compared to the other two datasets. Restaurant and Laptop datasets contain implicit aspect and opinion terms while Phone dataset contains no samples with implicit aspect terms. The data distribution is shown in Table 1.

	Samples	Categories	Quadruples			
			EA&EO	EA&IO	IA&EO	IA&IO
Restaurant	2286	13	2429 (66.40%)	350 (9.57%)	530 (14.49%)	349 (9.54%)
Laptop	4076	121	3269 (56.77%)	1237 (21.48%)	910 (15.80%)	342 (5.94%)
Phone	7115	88	13160 (82.86%)	2724 (17.14%)	N/A	N/A

Table 1: The data distribution of three datasets, where EA, EO, IA, and IO respectively represent explicit aspect and opinion terms, implicit aspect and opinion terms.

Evaluation Metrics We utilize the F1 score as the primary evaluation metric and report the corresponding precision and recall scores. During the experiment, a quadruple prediction is considered correct only when all predicted elements match the gold labels entirely.

Implementation Details We use the pre-trained model of DeBERTaV3-base[2] and DeBERTaV3-large[3] (He et al., 2021), which has a richer corpus than BERT (Devlin et al., 2018). The AdamW optimizer (Loshchilov and Hutter, 2017) is employed during the model training process with a learning rate set to 10^{-4} and the tempertature τ set to 0.07. The loss weights (α, β, λ) are set to (0.5,0.5,0.7)

[2] https://huggingface.co/microsoft/deberta-v3-base

[3] https://huggingface.co/microsoft/deberta-v3-large

Model	Restaurant			Laptop			Phone		
	P	R	F1	P	R	F1	P	R	F1
Extract-Classify(Cai et al., 2021b)	38.54	52.96	44.61	45.56	29.48	35.80	31.28	33.23	32.23
GAS(Zhang et al., 2021c)	60.69	58.52	59.59	41.60	42.75	42.17	50.72	48.15	49.40
Paraphrase(Zhang et al., 2021b)	58.98	59.11	59.04	41.77	**45.04**	43.34	46.72	49.84	48.32
GEN-SCL-NET†(Peper and Wang, 2022)	59.46	58.78	59.12	44.13	43.11	43.61	45.16	51.56	48.15
OTP(Bao et al., 2023)	**71.13**	56.08	62.71	45.12	37.91	41.20	-	-	-
MVP(Gou et al., 2023)	60.86	59.84	60.35	44.38	43.34	43.85	52.00	**52.44**	52.22
One-ASQP(Zhou et al., 2023)	65.91	56.24	60.69	43.80	39.54	41.56	57.42	50.96	54.00
IACOS(Xu et al., 2024)	57.24	53.21	55.15	**49.59**	34.65	40.80	-	-	-
Llama2†(Touvron et al., 2023)	61.27	60.65	60.95	43.76	43.17	43.46	51.67	51.37	51.52
GPT-4o mini(15-shot)†	41.39	35.42	38.17	16.95	15.01	15.92	26.14	17.78	21.16
CACA(base)	66.31	61.24	63.16	45.26	41.37	43.22	**66.40**	46.39	54.59
CACA(large)	67.45	**62.53**	**64.67**	46.30	43.19	**44.69**	63.78	49.35	**55.65**

Table 2: Results of Restaurant, Laptop and Phone Datasets compared to other baselines. Most results are from their original papers, with those marked with an "†" indicating our own reproduced results.

and the threshold δ is set to 0.9. We train our framework in a total of 200 epochs on the NVIDIA 3090 GPU. We utilize the validation set to select the best checkpoint for final testing. The experiment was conducted under five different random seeds and the final score was based on the average of these five runs.

Baseline We compare our method against several strong baselines for ASQP as follows:

- **The extractive-based methods:** Extract-Classify (Cai et al., 2021b), One-ASQP (Zhou et al., 2023), IACOS (Xu et al., 2024);
- **The generative-based methods:** GAS (Zhang et al., 2021c), Paraphrase (Zhang et al., 2021b), MVP (Gou et al., 2023), GEN-SCL-NET (Peper and Wang, 2022), OTP (Bao et al., 2023);
- We alse benchmark popular LLMs like LLaMa-7B (Touvron et al., 2023) which we fine-tune it on the three datasets and GPT-4o mini[4]. Detailed setups and more results for GPT-4o mini are described in Appendix A.4.

3.2 Main Results

The results of all baselines on the experimental datasets are shown in Table 2. We can see that our model outperforms all baselines, particularly in terms of F1 score. It is evident that our model has certain advantages compared to the latest generative models. Regarding precision, our confidence-based elimination strategy filters out a batch of incorrect predictions, resulting in outstanding precision performance across all baselines. For the F1 scores, our model surpasses the best-performing baseline by nearly 2 F1-score on the Restaurant and Phone dataset. We also apply CACA to the Pair-wise Aspect and Opinion Terms Extraction (PAOTE) task and achieve excellent results (refer section 4.2 for details).

In addition, to validate the model's performance on different types of data, we divided the test set into multiple subsets based on explicit and implicit targets. We calculated the F1 score for each subset and the results are shown in Table 3. By observing the data, our method generally achieved better results on targets with implicit aspects. In the Restaurant dataset, our method improved by nearly 19% over the best baseline on EA&IO. This indicates that we have effectively aligned the relationship between implicit and explicit targets.

3.3 Ablation Study

To validate the effectiveness of each module, we conducted ablation experiments on three datasets, focusing on the bidirectional interactive decoding model, contrastive learning in the implicit module and the elimination strategy. As shown in Table 4, the effectiveness of our modules was confirmed. The aspect term to opinion term direction of decoding had the most significant impact on the

[4] https://openai.com/

Model	Restaurant				Laptop				Phone	
	EA&EO	EA&IO	IA&EO	IA&IO	EA&EO	EA&IO	IA&EO	IA&IO	EA&EO	EA&IO
Extract-Classify	45.0	23.9	34.7	N/A	35.4	16.8	39.0	N/A	35.2	24.2
Paraphrase	65.4	45.6	53.3	45.6	<u>45.7</u>	33.0	51.0	<u>39.6</u>	49.1	45.6
GEN-SCL-NAT	<u>66.5</u>	<u>46.2</u>	56.5	<u>50.7</u>	**45.8**	<u>34.3</u>	<u>54.0</u>	<u>39.6</u>	50.1	45.4
One-ASQP	66.3	31.1	**64.2**	N/A	44.4	26.7	53.5	N/A	<u>54.8</u>	<u>52.9</u>
CACA	**69.5**	**55.2**	<u>60.3</u>	**55.8**	44.9	**35.3**	**54.8**	**41.2**	**57.1**	**54.1**

Table 3: The F1 scores on testing subsets with the combinations of the explicit and implicit targets.

Ablation Study	Restaurant	Laptop	Phone	Avg.Δ
CACA	**64.67**	**44.69**	**55.65**	-
w/o A→O Direction	63.07	43.51	54.33	-1.37
w/o O→A Direction	64.03	43.78	54.58	-0.87
w/o L^a_{CL} & L^o_{CL}	63.47	43.60	55.03	-0.91
w/o Elimination Strategy	63.26	43.36	54.67	-1.24

Table 4: The ablation experiments on individual modules.

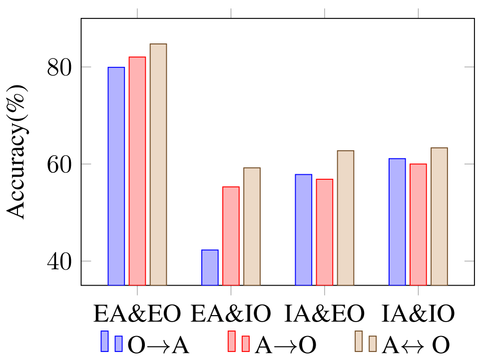

Figure 3: Accuracy of different implicit and explicit target combinations on the restaurant dataset

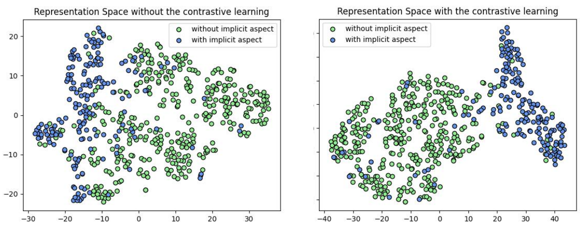

Figure 4: The representation space with the contrastive learning on the Restaurant dataset.

model. While opinion terms do guide aspect terms to some extent, aspect terms are the core of the entire quadruple, and decoding from opinion terms to aspect terms is more of an auxiliary role. Secondly, the elimination strategy proved effective, indicating that the model produced a considerable number of low-confidence answers, which were often incorrect. In general, we believe that the structure we devised is effectively validated for quadruple application.

4 Discussion and Analysis

4.1 The performance on implicit level

We have already discussed the effects under different implicit combination subsets in Section 3.2. Additionally, we similarly divided the data of Restaurant dataset as shown in Table 1 and calculated the accuracy of aspect terms and opinion terms from both unidirectional and bidirectional perspectives without considering the categories and sentiment polarity as shown in Figure 3. We observed that when we consider results from both directions, the accuracy of implicit aspect-opinion pairs shows a significant improvement. This validates the effectiveness of addressing implicit target recognition from both directions.

For the implicit representation, we employed contrastive learning in implicit target extraction. Figure 4 illustrates the implicit representation space under aspect terms in the Restaurant dataset. It is evident that after incorporating contrastive learning, the boundary between explicit and implicit representation spaces becomes more distinct in the right image.

4.2 The performance on PAOTE task

To demonstrate our CACA's ability to effectively integrate the relationship between aspect and opinion, and to better model their correlation, we apply the model to the PAOTE task with widely used datasets, which focus on modeling between aspect and opinion. We compared with the latest baseline methods (SpanMlt (Zhao et al., 2020), GTS (Wu et al., 2020), LAGCN (Wu et al., 2021), MAIN (Liu et al., 2022)) and the results are shown in Table 5.

As indicated by the results, our model outperforms all methods by approximately 4 F1 points across four benchmark datasets. This demonstrates CACA's effectiveness in simultaneously extracting aspect and opinion terms, enabling a better cap-

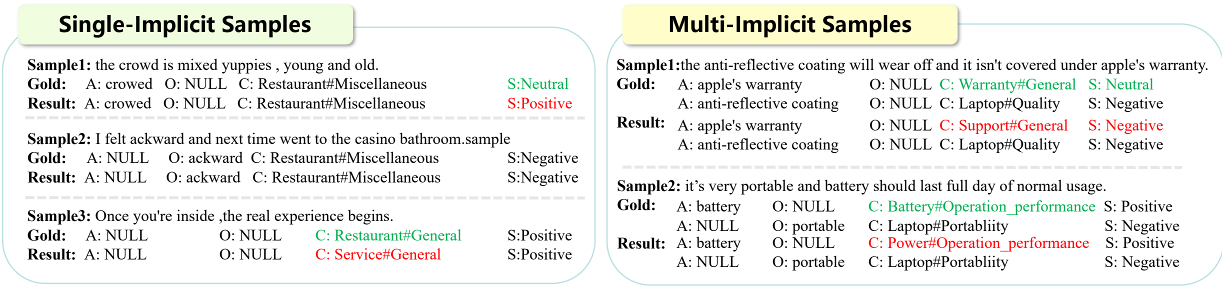

Figure 5: The case study of our model.

ture of fine-grained sentiment information in text. And by the alignment of aspect and opinion terms, CACA fully leverages the relationship between aspect terms and opinion terms, significantly enhancing extraction accuracy and realation consistency.

Model	14lap	14res	15res	16res
SpanMlt(Zhao et al., 2020)	68.66	75.60	64.68	71.78
GTS(Wu et al., 2020)	65.67	75.53	67.53	74.62
LAGCN(Wu et al., 2021)	68.88	76.62	68.91	76.59
MAIN(Liu et al., 2022)	69.86	77.54	70.92	77.97
CACA	**73.87**	**81.42**	**75.15**	**82.70**

Table 5: The F1 scores on PAOTE task with our alignment network and the results are cited from their original publications.

4.3 Effect of threshold δ and loss weight λ

Th threshold δ represents the confidence boundary in the model's elimination strategy. When δ is set to 0.9, the model's performance is at its best. This indicates that the model indeed has a portion of low-confidence incorrect answers and validates the effectiveness of the Elimination Strategy. The λ value denotes the loss weigths in different directions in aspect and opinion terms. By adjusting λ, our CACA can focus more on a specific direction, thereby improving overall performance. As the Table 6 is shown, when λ is set to 0.3, the model's performance decreases to some extent. However, when set to 0.7, the model's performance improves. This indicates that focusing on the alignment direction from aspect terms to opinion terms plays a crucial role in improving the extraction of quadruples. In other words, aspect terms have a stronger directional influence on opinion terms. Nonetheless,observing the ablation experiments also reveals that another direction contributes to a little performance improvement.

Hyperparamters		Restaurant	Laptop	Phone
$\delta =$	0.6	63.92	43.78	54.39
	0.7	63.87	43.97	54.56
	0.8	64.09	44.35	55.43
	0.9	**64.67**	**44.69**	**55.65**
$\lambda =$	0.3	63.75	44.06	54.34
	0.5	64.13	44.29	55.13
	0.7	**64.67**	**44.69**	**55.65**

Table 6: Model performance under different δ and λ hyperparameters.

4.4 The Effect of T5-Encoder

To validate the indivisibility of the encoder-decoder architecture in generative models, we extracted the T5 encoder and applied it to our model. The results, as shown in the Table 7, fully demonstrate the inherent limitations of generative models.

PLM	Parameters	Restaurant	Laptop
T5-base encoder	110M	59.27	41.42
DebertaV3-base	86M	63.16	43.22
DebertaV3-large	304M	64.67	44.69

Table 7: The T5-Encoder's Effect of Our Structure.

4.5 Case Study

We conduct a case study on CACA with a few examples as shown in Figure 5. We divided the samples into two categories: single-implicit and multi-implicit, and displayed some errors that occurred. In single implicit samples, our first sample made an error in polarity judgment, while in the third sample, there was an error in category judgment, which may be due to issues arising from the imbalance in the dataset distribution. In multi-implicit samples, we also encountered some errors in predicting sentiment polarity and aspect category classification. However, despite the complexity of multiple implicit targets, our CACA still performs

Datasets	Single Element				Pair Elements		Triple Elements			
	A	O	C	S	A&O	C&S	O&C&S	A&C&S	A&O&S	A&O&C
Restaurant	29.32%	27.16%	35.16%	47.38%	12.78%	30.73%	14.64%	15.51%	9.90%	5.30%
Laptop	54.68%	52.39%	32.86%	63.81%	40.31%	29.01%	11.76%	18.80%	36.33%	4.80%
Phone	19.36%	26.16%	26.75%	63.65%	9.12%	22.84%	7.46%	9.91%	7.24%	2.07%

Table 8: The error rate of the remaining elements after excluding the corresponding element, where A, O, C, and S represent Aspect Term, Opinion Term, Aspect Category, and Sentiment Polarity, respectively.

well in accurately predicting the presence of multi-implicit targets. This strongly demonstrates the classification capability of our model in handling implicit issues.

4.6 Further Error Analysis

To further explore which element the model performs best on and where there is still room for improvement, we calculated the overall error rate of the remaining elements when one, two, or three elements are correctly predicted. The error rates for various cases are shown in the Table 8.

Specifically, when only one element is correctly predicted, the overall error rate for the remaining three elements is higher. Among these, when sentiment is correct, the model exhibits the highest error rate, indicating that predicting the other three elements is more challenging compared to sentiment polarity. When two elements are correctly predicted, the error rates for the other two elements are below 30%, especially for the category and sentiment polarity. This demonstrates the effectiveness of our model in extracting and aligning aspect terms and opinion terms. When aspect terms and opinion terms are correct, the error rate is around 10% except for the laptop dataset, which is due to the specific nature of its categories. When all three elements are correctly predicted, it can be observed that sentiment polarity has the least impact, while category has the greatest impact on the laptop dataset.

5 Related Work

With the rapid development of ASQP, the quadruple extraction task is mainly divided into two categories: generative-based methods and the other based on extractive methods.

Generative methods: Previous works design novel approaches based on tree structure (Mao et al., 2022; Bao et al., 2022, 2023); Some introduce contrastive learning to the generative models (Peper and Wang, 2022; Li et al., 2024); Moreover, Some researchers enhance their generative models through data augmentation techniques (Hu et al., 2022; Yu et al., 2023; Wang et al., 2023; Zhang et al., 2024b,c).

However, although all generative models have rich pre-trained knowledge, they are limited by the inherent encoder-decoder structure and cannot well perceive the implicit existence by building unique structural modeling complex context information.

Extractive methods: There are relatively few studies focusing on the extraction of all four elements. Cai et al. (2021b) first used an extractive approach to solve the ASQP task. Zhou et al. (2023); Zhang et al. (2024a) decompose the quadruple extraction task into multiple subtasks. Xu et al. (2024) leverages informative and adaptive negative examples to jointly train the multi-label classifier and the other two classifiers on categories and sentiments by multi-task learning.

However, in all quadruple works recent years, there are few researchers investigating how to better align implicit and explicit terms, and well utilize the contextual information to improve the performance of the implicit target. Our proposed method of using constrastive learning to obtain the implicit representation and Context-Aware Cross-Attention Network can enhance the alignment of aspect terms and opinion terms effectively.

6 Conclusions

In this paper, we proposed the CACA network to tackle the issue of implicit target and enhance the alignment of the aspect terms and opinion terms. We introduce contrastive learning to make explicit target representations more distinguishable from implicit target representations. By integrating with the CACA network, the relationship between aspect terms and opinion terms is effectively aligned, whether explicit or implicit. Experiments on three benchmark datasets demonstrate that our model outperforms the baselines.

Limitations

Our CACA model also has some limitations. Our model often makes errors when dealing with numerous aspect categories that have few labeled examples. Therefore, it is crucial to investigate more resilient approaches for identifying aspect categories in low-resource settings. Moreover, our model makes a bad performance when handling neutral text. This could be due to the presence of certain sentiment words in the samples, which may affect the judgment of sentiment polarity, causing entities that originally lack polarity to be assigned a certain sentiment. This is also the main research direction in the future. We can consider integrating data augmentation to improve the performance on these data.

Acknowledgments

This research was financially supported by the Open Research Fund from Guangdong Laboratory of Artificial Intelligence and Digital Economy (SZ), under Grant No. GML-KF-24-23, National Science and Technology Major Project (2021ZD0111501), National Science Fund for Excellent Young Scholars (62122022), Natural Science Foundation of China (U24A20233, 62476163, 62206064, 62206061, 62406078), the major key project of PCL (PCL2021A12), Guangdong Basic and Applied Basic Research Foundation (2023B1515120020), Collaborative Education Project of the Ministry of Education (202407). This research was enabled by the computational resources and support of the High Performance Computing Platform at the School of Computer Science, Guangdong University of Technology.

References

Xiaoyi Bao, Xiaotong Jiang, Zhongqing Wang, Yue Zhang, and Guodong Zhou. 2023. Opinion tree parsing for aspect-based sentiment analysis. *arXiv preprint arXiv:2306.08925*.

Xiaoyi Bao, Zhongqing Wang, Xiaotong Jiang, Rong Xiao, and Shoushan Li. 2022. Aspect-based sentiment analysis with opinion tree generation. In *IJCAI*, volume 2022, pages 4044–4050.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021a. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 340–350, Online. Association for Computational Linguistics.

Hongjie Cai, Rui Xia, and Jianfei Yu. 2021b. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 340–350.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

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

Yonghyun Jun and Hwanhee Lee. 2024. Dynamic order template prediction for generative aspect-based sentiment analysis. *Preprint*, arXiv:2406.11130.

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

Yijiang Liu, Fei Li, Hao Fei, and Donghong Ji. 2022. Pair-wise aspect and opinion terms extraction as graph parsing via a novel mutually-aware interaction mechanism. *Neurocomputing*, 493:268–280.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

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

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

Joseph Peper and Lu Wang. 2022. Generative aspect-based sentiment analysis with contrastive learning and expressive structure. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6089–6095, Abu Dhabi, United Arab Emirates. 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.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9122–9129.

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

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. *arXiv preprint arXiv:2105.02520*.

Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for aspect-oriented fine-grained opinion extraction. *arXiv preprint arXiv:2010.04640*.

Xiancai Xu, Jia-Dong Zhang, Lei Xiong, and Zhishang Liu. 2024. iacos: Advancing implicit sentiment extraction with informative and adaptive negative examples. *Preprint*, arXiv:2311.03896.

Yongxin Yu, Minyi Zhao, and Shuigeng Zhou. 2023. Boosting aspect sentiment quad prediction by data augmentation and self-training. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Hua Zhang, Xiawen Song, Xiaohui Jia, Cheng Yang, Zeqi Chen, Bi Chen, Bo Jiang, Ye Wang, and Rui Feng. 2024a. Query-induced multi-task decomposition and enhanced learning for aspect-based sentiment quadruple prediction. *Engineering Applications of Artificial Intelligence*, 133:108609.

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

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

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021c. Towards generative aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 504–510.

Wenyuan Zhang, Xinghua Zhang, Shiyao Cui, Kun Huang, Xuebin Wang, and Tingwen Liu. 2024b. Adaptive data augmentation for aspect sentiment quad prediction. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11176–11180.

Yice Zhang, Jie Zeng, Weiming Hu, Ziyi Wang, Shiwei Chen, and Ruifeng Xu. 2024c. Self-training with pseudo-label scorer for aspect sentiment quad prediction. *Preprint*, arXiv:2406.18078.

He Zhao, Longtao Huang, Rong Zhang, Quan Lu, and Hui Xue. 2020. Spanmlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 3239–3248.

Junxian Zhou, Haiqin Yang, Yuxuan He, Hao Mou, and Junbo Yang. 2023. A unified one-step solution for aspect sentiment quad prediction. *arXiv preprint arXiv:2306.04152*.

A Appendix

A.1 The Opinion Decoder

In this section, we provide a detailed explanation of the process for obtaining opinion terms. The opinion decoding process and the loss can be described as follows:

$$S' = Dropout(W'_{down} \cdot \sigma(W'_{up} \cdot S)) \quad (28)$$

$$S^O = LayerNorm(S' + S) \quad (29)$$

$$p^o = Softmax(W_o(S^O)) \quad (30)$$

$$\mathcal{L}_{\mathcal{O}} = -\sum_i y_i^o \log(p_i^o) \quad (31)$$

where W'_{down}, W'_{up} and W_o are the trainable parameters. σ is the activation function. y_i^o is the ground truth labels of the opinion terms. p_i^o is the probability distribution of opinion terms.

A.2 The Implicit Opinion Module

We use a similar approach to predict whether implicit opinion terms exist in the text by building a classifier and incorporating contrastive learning to distinguish between texts containing implicit terms and those containing only explicit terms. The specific formula representation and loss are as follows:

$$S^{imp,o} = FNN_a(Concat(Mean(S^{context}, S^O)) \quad (32)$$

$$p^{imp,o} = Softmax(W_{imp}^o S^{imp,o}) \quad (33)$$

$$\mathcal{L}^{imp,o} = -\sum_i y_i^{imp,o} \log(p_i^{imp,o}) \quad (34)$$

where $S^{context}$ is given by the formula equation 8. $W_{imp}^o \in R^{b\times 2}$ is the trainable parameters. $y_i^{imp,o}$ is the label that indicates whether the text contains implicit opinion terms. $p_i^{imp,o}$ is the corresponding probability distribution.

For each sample in mini-batch B, our implicit contrastive learning loss for opinion is defined as:

$$\mathcal{L}_i^o = -\frac{1}{M(i)} \sum_{j\in M(i)} log \frac{exp(sim(u_i^o, u_j^o)/\tau)}{\sum_{k=1}^{B} exp(sim(u_i^o, u_k^o)/\tau)} \quad (35)$$

where $M(i)$ denotes the set of examples with the same label and k $\neq$ i. u_i^o is the i^{th} representation of $S^{imp,o}$, τ is the temperature coefficient.

A.3 Alignment from Opinion to Aspect

For the another direction from opinion to aspect, the specific formula is as follows:

$$S_{o\to a}^{inter} = Softmax(\frac{S^O W_q^3 (S^A W_k^3)^T}{\sqrt{d}}) \cdot S^A W_v^3 \quad (36)$$

$$S^{O'} = W^3 \cdot (LayerNorm(S^O + S_{o\to a}^{inter}) \quad (37)$$

where W_q^3, W_k^3, W_v^3 and W^3 are the trainable parameters.

Similarly, we use the module shown in the Figure 2 to identify the aspect term information most relevant to the specific opinion term.

$$A^{o\to a} = Softmax(\frac{S^{O'} W_q^4 (Set^O W_k^2)^T}{\sqrt{d_k}}) \quad (38)$$

$$q^{o\to a} = W_{o\to a}(A^{o\to a} \cdot S^{O'} W_v^4) \quad (39)$$

$$p(a|o) = Softmax(q^{o\to a}) \quad (40)$$

where $W_{o\to a} \in R^{M\times s}$, and $s \in$ {Positive, Negative, Neural, Invalid} is the sentiment class of every span. $p(a|o)$ is a probability distribution indicating the likelihood of all possible opinion terms given the aspect term.

A.4 Experiments with ChatGPT

For the GPT model, we utilize In-Context Learning (Brown et al., 2020). For detail, we used the prompt provided in the DOT (Jun and Lee, 2024) and randomly sampled a specified number of instances. The prompt templates we used are shown in the Figure 6, and we conducted the corresponding experiments in a few-shot manner on ChatGPT-4o mini. The results are shown in the Table 9.

It is evident that GPT has certain limitations when handling the more complex quad extraction task. However, as the number of samples provided in the prompt increases, there can be some improvement in the model's performance. Yet, compared to smaller models, the performance gap is still significant, almost more than double. In our experiments, we conducted a sampling inspection of the samples extracted by the LLMs and found that the errors were mainly due to the predicted entity targets being overly complex, with some irrelevant information mixed in. Additionally, the predictions of implicit targets were often haphazard, frequently misidentifying texts without implicit entities as having them, which negatively impacted the model's performance.

Shot Numbers	Restaurant			Laptop			Phone		
	P	R	F1	P	R	F1	P	R	F1
0-shot	17.20	16.66	16.93	10.08	9.93	10.01	9.39	7.28	8.2
5-shot	33.33	29.34	31.21	12.26	10.98	11.58	20.85	14.14	16.85
10-shot	36.58	32.03	34.15	15.04	13.23	14.08	25.90	17.75	21.06
15-shot	41.39	35.42	38.17	16.95	15.01	15.92	26.14	17.78	21.26
CACA	**67.45**	**62.53**	**64.67**	**46.30**	**43.19**	**44.69**	**63.78**	**49.35**	**55.65**

Table 9: The F1 results on GPT-4o mini with different shot numbers and the same prompt template.

Prompt

According to the following sentiment elements definition :

- The 'aspect term' refers to a specific feature, attribute, or aspect of a productor service that a user may express an opinion about, the aspect term might be 'null' for implicit aspect.
- The 'opinion term' refers to the sentiment or attitude expressed by a user towards a particular aspect or feature of a product or service, the aspect term might be 'null' for implicit opinion.
- The 'aspect category' refers to the category that aspect belongs to, and the available categories includes: {dataset specific categories}.
- The 'sentiment polarity' refers to the degree of positivity, negativity or neutrality expressed in the opinion towards a particular aspect or feature of a product or service , and the available polarities includes: 'positive ', 'negative' and 'neutral ‘.

Recognize all sentiment elements with their corresponding aspect terms , aspect categories , opinion terms and sentiment polarity in the following text with the format of [('aspect term', 'opinion term', 'aspect category', 'sentiment polarity'), ...]:

<Example#1 Start>
Sentence: {Sentence}
Output: {Tuple List}
<End>
....
<Input>
Sentence: {Sentence}
Output:

Figure 6: The prompt template we used on GPT-4o mini.