Enhancing Aspect Sentiment Quad Prediction through Dual-Sequence Data Augmentation and Contrastive Learning

Shihao Li 1556574783@qq.com
School of Computer Science and Technology,
Guangdong University of Technology, Guangzhou, 510006, Guangdong, China

Nankai Lin neakail@outlook.com
School of Computer Science and Technology,
Guangdong University of Technology, Guangzhou, 510006, Guangdong, China

Pinmo Wu wupm@mail2.sysu.edu.cn
School of Mathematics, Sun Yat-Sen University, 510006, Guangdong, China

Dong Zhou✉ dongzhou@gdufs.edu.cn
School of Information Science and Technology,
Guangdong University of Foreign Studies, Guangzhou, 510006, Guangdong, China

Aimin Yang✉ amyang18@163.com
School of Computer Science and Intelligence Education,
Lingnan Normal University, Zhanjiang, 524000, Guangdong, China

Abstract

Aspect sentiment quad prediction (ASQP) endeavors to analyze four sentiment elements in sentences. Recent studies utilize generative models to achieve this task, yielding commendable outcomes. However, these studies often fall short of fully leveraging the relationships between sentiment elements and have difficulty effectively handling implicit sentiment expressions. Furthermore, this task also confronts the obstacle of data scarcity stemming from the substantial expenses involved in data annotation. To address these limitations, we propose dual-sequence data augmentation to achieve diverse input and target expressions, while we incorporate contrastive learning to instigate the model to distinctly represent the presence or absence of these pivotal features pertaining to implicit aspects and opinion terms. Additionally, we introduce a prediction normalization strategy to refine the produced results. Empirical findings from experiments on four publicly available datasets show the superiority of our method, surpassing multiple baseline approaches and achieving state-of-the-art performance on the benchmark.

Keywords: Aspect sentiment Quad Prediction; Dual-sequence Data Augmentation; Contrastive Learning; Prediction Normalization.

1. Introduction

The primary goal of aspect-based sentiment analysis (ABSA) is to extract the required sentiment element tuples from a particular sentence. Recently, the aspect sentiment quad prediction (ASQP) (Zhang et al., 2021) task has been introduced, also known as ACOS in (Cai et al., 2021). It involves extracting four sentiment elements, comprising aspect term (at), opinion term (ot), sentiment polarity (sp), and aspect category (ac) (Zhang et al., 2022). As shown in Figure 1, regarding the sentence "Inexpensive, and the food is delicious, but I wish the seating space was a little bigger.", "food" is recognized as the aspect term, "delicious" functions as the opinion term delineating that aspect term, "positive" signifies the expressed sentiment polarity, and "food quality" characterizes the corresponding aspect category. These four elements amalgamate into an explicit quad (food, food quality, delicious, positive). Furthermore, two additional quads necessitate extraction: (NULL, restaurant prices, Inexpensive, positive) containing implicit aspect term, (seating space, ambience general, NULL, negative) incorporating an implicit opinion term.

Existing work highlights generative methods as a promising research direction. A prevalent approach entails the generation of sentiment element sequences within a predetermined framework, with the objective of leveraging tag semantics. To illustrate, Paraphrase (Zhang et al., 2021) transforms quad extraction into a generation task. Employing predefined rules, they initially map four elements (ac, at, ot, sp) to semantic values ($m_{ac}, m_{at}, m_{ot}, m_{sp}$). These values are then inserted into a predefined template to produce a natural language target sequence. As depicted in Figure 1, the original sentence is "rewritten" into a target sequence through paraphrase. Following the fine-tuning of the pre-trained language model using the sequence-to-sequence learning paradigm, decoding quads from the target sequence emerges as a straightforward process.

Source Text:	Inexpensive, and the food is delicious, but I wish the seating space was a little bigger.
Quads(*ac, at, ot, sp*):	(food, food quality, delicious, positive) (NULL, restaurant prices, Inexpensive, positive) (seating space, ambience general, NULL, negative)
Mapped Quads(m_{ac}, m_{at}, m_{ot}, m_{sp}):	(food, food quality, delicious, great) (it, restaurant prices, Inexpensive, great) (seating space, ambience general, NULL, bad)
paraphrased template:	m_{ac} is m_{sp} because m_{at} is m_{ot}
Single-order Prediction:	food quality is great because food is delicious restaurant prices is great because it is Inexpensive ambience general is bad because seating space is NULL

Figure 1: An example sentence is rewritten into a single-order target sequence using a paraphrased template.

However, as exemplified above, the generation of a sentiment element sequence in a single-order $\{m_{ac} \rightarrow m_{sp} \rightarrow m_{at} \rightarrow m_{ot}\}$ ignores the impact of language expression diversity

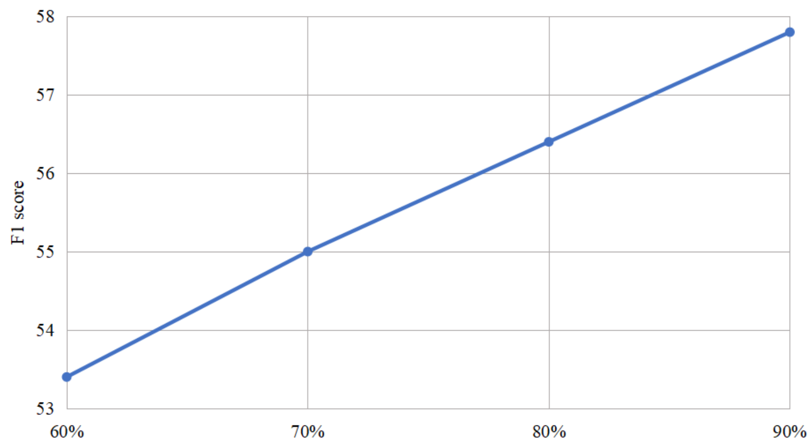

Figure 2: F1 scores of Paraphrase with training data ratios of 60%, 70%, 80%, and 90%.

and the interdependence of elements within aspect sentiment tuples on the target sequence. The aspect sentiment tuple possesses the order-free property, implying that different orders, such as $\{m_{ac} \rightarrow m_{sp} \rightarrow m_{at} \rightarrow \ m_{ot}\}$ and $\{m_{at} \rightarrow m_{ot} \rightarrow m_{sp} \rightarrow m_{ac}\}$, are valid, and the arrangement of the four elements will affect the performance of the generation-based pre-trained language model (Hu et al., 2022). In addition, the presence of implicit *at* and/or *ot* in sentences consistently worsens model generalization, with more than 30% of sentences containing implicit language (Cai et al., 2021). These cases, such as "We waited for a long time.", are challenging for models due to the absence of explicit *at* and *ot*, e.g., "the service" and "slow". Concurrently, owing to the scarcity of annotation data, for instance, the Rest15 and Rest16 ASQP datasets (Zhang et al., 2021) only encompass 834 and 1264 training samples, respectively. We adjust the ASQP-Rest16 data ratio for Paraphrase training and present the results in Figure 2. This figure illustrates a steady enhancement in performance with the growth of data, suggesting that the effect is still far from saturation. Training a robust model with a limited dataset proves challenging, and the conventional method of augmenting training samples by mere sentence rephrasing restricts the semantic diversity and quality of the generated data (Wang et al., 2023).

To address the above challenges, we introduce dual sequence data augmentation along with contrast learning. Initially, we integrate multiple template orders for target-side data augmentation. The inclusion of different template orders facilitates various perspectives of the quad, fostering diverse target representations by receiving information from multiple templates. Simultaneously, we propose a quads-to-sentences (Q2S) generation method as input-side data augmentation to generate augmented data characterized by both robust diversity and high quality. Additionally, we introduce a supervised contrastive learning objective designed to enhance the model's efficacy in representing crucial features, such as aspect and opinion terms. This is achieved by maximizing the distance between inconsistent examples and minimizing the distance between consistent examples through a supervised contrastive loss. Furthermore, we implement a prediction normalization strategy to handle issues arising from the generated sentiment elements being outside their corresponding label vocabulary set and the generated quads being one-sided.

To conclude, the contributions of this study are outlined below:

(1) We introduce a dual-sequence data augmentation that simultaneously augments the input and output sequences and adopts a prediction normalization strategy to refine produced results.

(2) We incorporate a task-specific supervised contrastive learning approach, thereby improving the learning of example representations and yielding downstream benefits.

(3) Our approach exhibits enhanced performance on ASQP and ACOS benchmarks. Specifically, on the Rest15 and Rest16 datasets of ASQP, F1 scores improved by 0.67% and 0.51%, respectively. Similarly, on the Restaurant and Laptop datasets of ACOS, F1 scores show enhancements of 0.92% and 0.49%, respectively.

2. Related Work

Early studies on ABSA were mainly directed towards predicting separate sentiment elements, including *at* extraction (Ma et al., 2019), *ot* extraction (Mensah et al., 2021), *ac* detection (Brauwers and Frasincar, 2022), and predicting the *sp* of a given *at* (Zhang and Qian, 2020) or *ac* (Hu et al., 2019). Subsequently, researchers (Zhang et al., 2022) recognized the interconnected nature of these sentiment elements, prompting a shift towards simultaneous recognition. This involved tasks like identifying aspect-sentiment pairs (Cai et al., 2020) or triples (Mao et al., 2021). Recently, there has been a rising focus on predicting all four sentiment elements at once, with two main approaches gaining attention: pipeline methods (Cai et al., 2021) and generation-based methods (Zhang et al., 2021). Due to their ability to alleviate the accumulation of errors in pipeline processes and make better use of the rich semantic information contained in labels (Yu et al., 2023), generative methods have become a prominent research focus. Promising works have devised innovative approaches based on contrastive learning (Peper and Wang, 2022), tree structures (Mao et al., 2022), impossibility learning (Hu et al., 2023), and data augmentation (Wang et al., 2023; Gou et al., 2023).

Data augmentation is a prevalent technique in the language domain aimed at enhancing model performance. Previous methods for data augmentation fall into three categories. The first category concentrates on input augmentation, including techniques like text modification (Wei and Zou, 2019) and back-translation (Sugiyama and Yoshinaga, 2019) of natural language. The second category focuses solely on augmenting the output, as demonstrated by sequence-to-sequence learning with virtual sequences employed as target-side data augmentation (Xie et al., 2021). The third category involves both input and output augmentation, exemplified by mixup (Zhang et al., 2017), which creates virtual training samples by linearly combining feature vectors with their corresponding targets. In this work, we follow the generative approach and achieve enhanced performance on test data through the incorporation of dual-sequential data augmentation and task-specific supervised contrastive learning objectives.

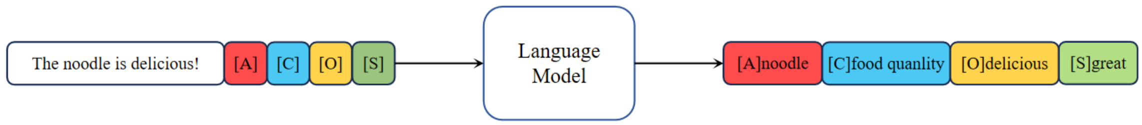

Figure 3: Example of an input-target pair.

3. Methodology

3.1. Task Definition

The objective of aspect sentiment quad prediction (ASQP) is to forecast all aspect-level quads $\{(at, ot, sp, ac)\}$ within a given sentence x. Following previous work (Zhang et al., 2021), we define a projection function to correlate quad (at, ot, sp, ac) with semantic values $(m_{at}, m_{ot}, m_{sp}, m_{ac})$. For example, we map the "NULL" label of aspect terms to "it", and map the sentiment polarity $sp \in$ {positive, neutral, negative} to words {great, ok, bad}, respectively. Applying the aforementioned rules, the values are input into the template T to construct the target sequence. To signify different sentiment elements, distinct markers "[A]m_{at}, [C]m_{ac}, [O]m_{ot}, [S]m_{sp}" (Hu et al., 2022) are employed. The markers for m_{at}, m_{ac}, m_{ot} and m_{at} are denoted as [A], [C], [O] and [S], respectively. Each element is prefixed with its corresponding marker and concatenated in the specified order o_i to construct the target sequence. In cases where the input sentence contains multiple quads, a special symbol [SSEP] is employed to link their corresponding templated sequences, resulting in the final target sequence y_{o_i}. To convey the desired ordering of sentiment elements o_i, we incorporate input element order prompts (Gou et al., 2023). The prompt (e.g., "[A][C][O][S]" signifying prediction in the order $\{m_{at} \rightarrow m_{ac} \rightarrow m_{ot} \rightarrow m_{sp}\}$) is appended to the end of each input sentence, yielding the ultimate input x_{o_i}. Therefore, the input-target pairs for training are obtained, as shown in Figure 3.

3.2. Dual-Sequence Data Augmentation

3.2.1. Multiple-Order Data Augmentation.

We propose sequences of sentiment elements in various orders as target-side data augmentation. We conceptualize different template orders akin to examining a picture from distinct angles, providing diverse viewpoints. Therefore, combining multiple template orders serves to counteract potential biases towards surface patterns, fostering a comprehensive understanding of the task's essence.

Template Order Selection. Since the overhead increases linearly with the number of template orders, employing all 24 sequence permutations significantly amplifies training time, and performance fluctuates among different template orders. Consequently, judicious template order selection becomes imperative. After studying prompt ordering (Gou et al., 2023), we select the order deemed likely to perform better by taking into account the average entropy of possible permutations on the training set.

The procedure unfolds is below: Initially, we enlist all conceivable order permutations o_i as candidates. Subsequently, for a given input sentence x and its target quad, we formulate

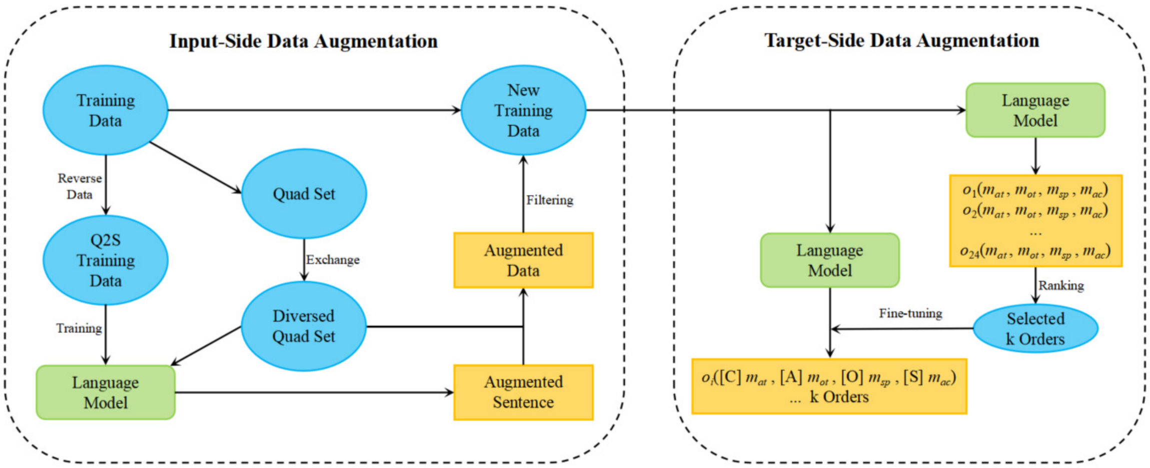

Figure 4: Overview of dual-sequence data augmentation. The o_i represents the i-th order permutation of the four sentiment elements.

the ordered target pattern y_{o_i} corresponding to the o_i, as described in §3.1. During this process, spaces replace the element markers to minimize noise. The entropy $p(y_{o_i} \mid x)$ is then obtained by conducting a query to the pre-trained language model. Finally, we compute the average entropy of o_i across the training set D:

$$S_{o_i} = \frac{\sum_D p(y_{o_i} \mid x)}{|D|}. \tag{1}$$

Consequently, we rank each permutation by S_{o_i} and select the top k permutations with smaller values for training.

Multiple Template Order Training. We employ the selected k ordered permutations to construct k different input-target pairs for each sentence, training multiple template sequences simultaneously. For a given input-target pair (x, y), we choose the pre-trained sequence-to-sequence language model T5 (Raffel et al., 2020) to initialize the parameters θ and fine-tune it through the minimization of the cross-entropy loss:

$$\mathcal{L}_{CE} = -\sum_{t=1}^{n} \log p_\theta(y_t \mid x, y_{<t}). \tag{2}$$

Here, t is the span of the target sequence y, and $y_{<t}$ signifies the prior produced token.

3.2.2. Quads-to-Sentences Data Augmentation

As stated above, we purely incorporate target sequences during model training, while the inclusion of both input and output sequences holds the potential for further performance enhancements, as shown in Figure 4. Therefore, we propose quads-to-sentences data augmentation as input-side data augmentation to augment the training dataset, with the objective of creating sentences describing given quads. Specifically, given n quads $\{q_1, q_2, \ldots, q_n\}$,

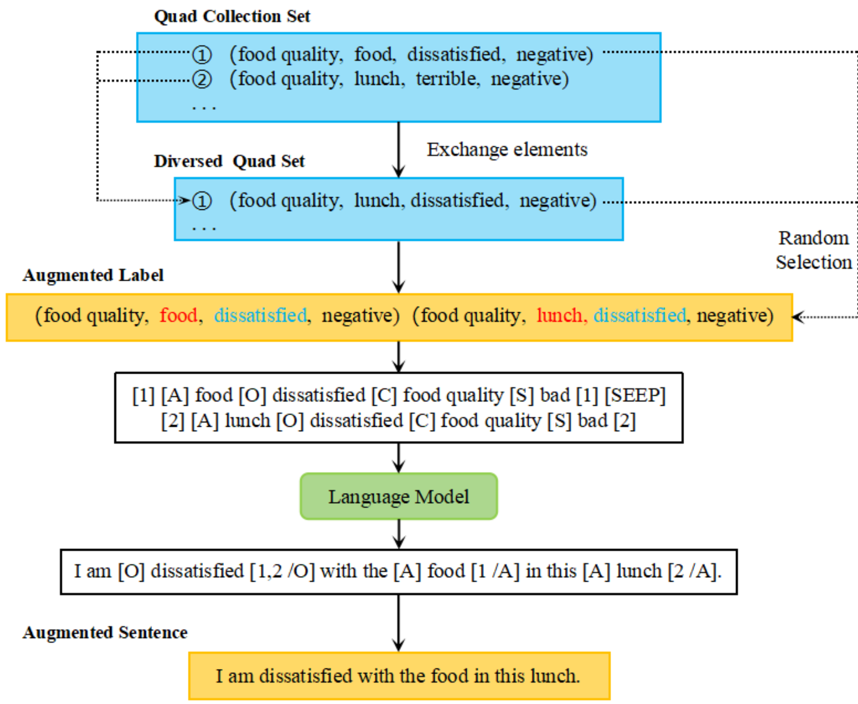

Figure 5: Example of generating an augmented sentence.

where $q_i = (at_i, ac_i, ot_i, sp_i)$, it is imperative to generate a sentence x that includes exclusively the input quads. Figure 5 presents the proposed method.

Quad Diversification. In the initial phase, we aggregate all quads from the training dataset into a set, referred to $S_{\text{origin}} = \{(at_i, ot_i, sp_i, ac_i)\}$. Subsequently, for quads sharing the same ac, we swap their at and (ot, sp) pairs at random to generate new quads. To ensure coherence, the ot, and sp from the same original quad are not separated, preventing the creation of new quads with conflicting elements. For instance, given two quads: (salads, delicious, positive, food quality) and (sandwiches, dry, negative, food quality), the new quads obtained are (sandwiches, delicious, positive, food quality) and (salads, dry, negative, food quality). Ultimately, the quantity of new quads for each ac is balanced to create the set of quads after swapping labels, denoted as S_{exchange} . By combining S_{origin} and S_{exchange} , we derive a diverse set of quads S_{diverse} , which is then utilized to train the quads-to-sentences model.

Quads-to-Sentences Model. ASQP endeavors to predict quads from a given sentence and Q2S can be perceived as the converse of ASQP. To handle the Q2S challenge, aligned with the previously detailed work on ASQP text generation described previously, we leverage pre-trained sequence-to-sequence models. Each time we randomly select 1 or 2 quads from S_{diverse} and input them into the Q2S model to perform data augmentation. In our approach, the primary emphasis is on the model's input and output design. For the input sequence, similar to §3.1, we convert the given quad into a templated sequence. Notably, special index markers are inserted at the beginning and end of each sequence to differentiate multiple quads. Specifically, the i-th quad (at_i, ot_i, sp_i, ac_i) is converted into a templated sequence: $[i]$[A]at_i[C]ot_i[O]sp_i[S]$ac_i[i]$. In cases where the input contains multiple quads, a special symbol [SSEP] is employed to connect their corresponding templated sequences. Concerning the output sequence, the Q2S model utilizes unique markers to annotate the aspect and opinion term ranges during sequence generation. Additionally, annotations encapsulate information regarding the association between at and ot. This enables the gathering of (at, ot) pairs from the output sequence, facilitating subsequent verification of the coherence between the detected pairs and the input quad. Then instances of incongruent at and ot are filtered out. Specifically, special markers "[A]", "[i /A]", "[O]" and "[i/O]" are used to annotate the at and ot of the i-th quad in the sentence. The special markers [A] and [O] signify the start of at and ot, while [i/A] and [i/O] indicate the terminus. In scenarios where multiple at in a sentence are described by the same ot, or vice versa, a list of numbers separated by commas within square brackets is employed to group them together. For example, [1, 2 A] indicates that the 1st and 2nd perspectives refer to the same at.

3.3. Contrastive Learning

3.3.1. Supervised Contrastive Loss

We incorporated a supervised contrastive learning objective into the fine-tuning process of downstream generation tasks for ASQP quad extraction. This addition aims to enhance the discriminative representation of crucial input features by the encoder-decoder model. Specifically, our objective is to facilitate the learning of representations for example-level at and ot, as illustrated by the labels in Figure 6.

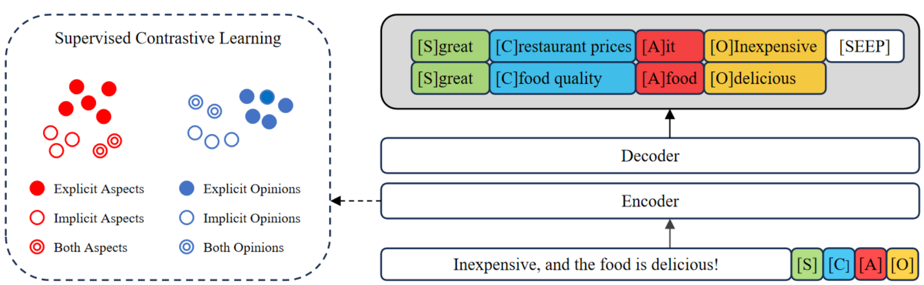

Figure 6: Overview of contrastive learning on ABSA. The trained model distinctly captures two key features: (1) at type and (2) ot term type.

In our approach, we produce representations of training examples x_i in training mini-batch X, similar to Peper and Wang (2022), by initially summing and pooling the output of the encoder as $\text{Mean}\,(\text{Encode}\,(x_i))$, which is next input to the dedicated fully-connected layer FC_f, where the feature $f \in \{at, ot\}$. Consequently, this process yields the representation r_{fi}. In our experimental setup, the fully connected layer has both input and output dimensions set at 768. Subsequently, by employing a dropout probability of p $= 0.1$, we introduce perturbations to each example representation r_{fi} without altering the original labels. This process generates modified views to expand X, resulting in X'. We follow Khosla et al. (2020), where the supervised contrastive loss is defined as:

$$\mathcal{L}_i^f = \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{e^{\operatorname{sim}\left(r_{fi}, r_{fp}\right)/\tau}}{\sum_{b \in B(i)} e^{\operatorname{sim}\left(r_{fi}, r_{fb}\right)/\tau}}, \tag{3}$$

where $B(i) \equiv X'/x_i$ denotes the collection of all other examples in the extended small batch X' except for the example x_i. $P(i) \equiv \{p \in B(i) : y_{fp} = y_{fi}\}$ symbolizes the set in $B(i)$ with the same label as in the example x_i. Additionally, τ signifies the temperature scaling parameter. The function sim(.) represents any similarity metric, such as cosine similarity or inner product.

3.3.2. Joint Training Objectives

The ultimate training goal is to incorporate our two feature-specific losses into the existing decoder cross-entropy loss $\mathcal{L}_{CE}$. The weights α and β are used for adjustments:

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha \mathcal{L}_i^{at} + \beta \mathcal{L}_i^{ot}. \tag{4}$$

3.4. Prediction Normalization

3.4.1. Element Correction

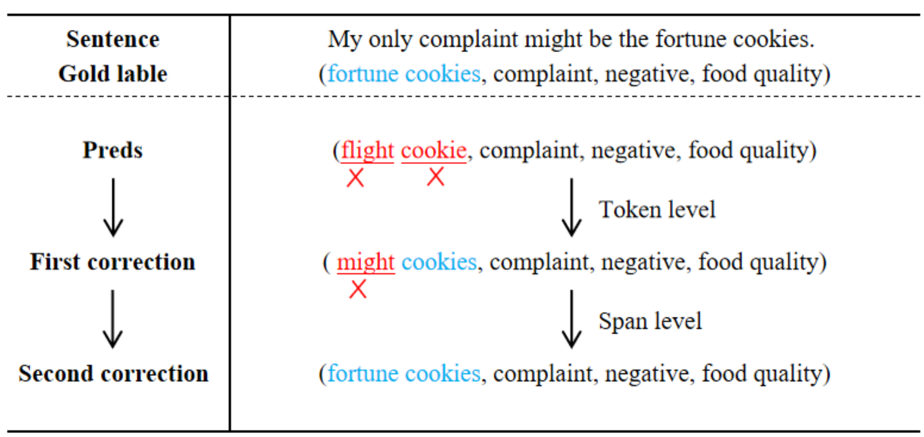

Figure 7: Example of element correction.

Ideally, a predicted element e for sentiment type t should exclusively pertain to its designated vocabulary set V_t. For example, the expected aspect and opinion terms should be explicitly present in the input sentences s. However, this may not always align with reality, given that each e is produced using a vocabulary set that includes all tokens instead of its specific vocabulary set. To tackle this issue, we introduce a corrective strategy using the most similar method. We initially construct its corresponding vocabulary set V_t. Then, if the e does not belong to the V_t, we find the most similar token replacement e from V_t via Levenshtein distance (Levenshtein et al., 1966). It should be noted that when the token length of the predicted aspect or opinion term exceeds 1, each token in e is traversed for correction. Subsequently, a new predicted element e' is obtained. If e' does not yet constitute a subsequence of the s, a secondary correction process is initiated to find the most similar subsequence in the s to replace e. An example of the element correction is shown in Figure 7.

3.4.2. Quad Correction

The model is capable of generating multiple quads through different template orders. Certain template orders result in the same correct quad, while others are less efficient and, consequently, may be incorrect. However, the likelihood of these less efficient orders causing the same error is low. In essence, when different template orders give the same quad, it is more likely to be correct. Building on this rationale, we propose a majority strategy that amalgamates information from multiple orders to rectify errors in a single order. Specifically, for the input sentence, when instructing the training model to generate from k selected order permutations, each predicting one or more quads, we begin by aggregating the outcomes of all order permutations. Subsequently, we determine the quads that appear in the majority of order permutations, i.e., those with a count greater than or equal to $\frac{k}{2}$, as the final prediction. Our final predictions gain enhanced reliability with the support of multiple orders. An example of the quad correction is shown in Figure 8.

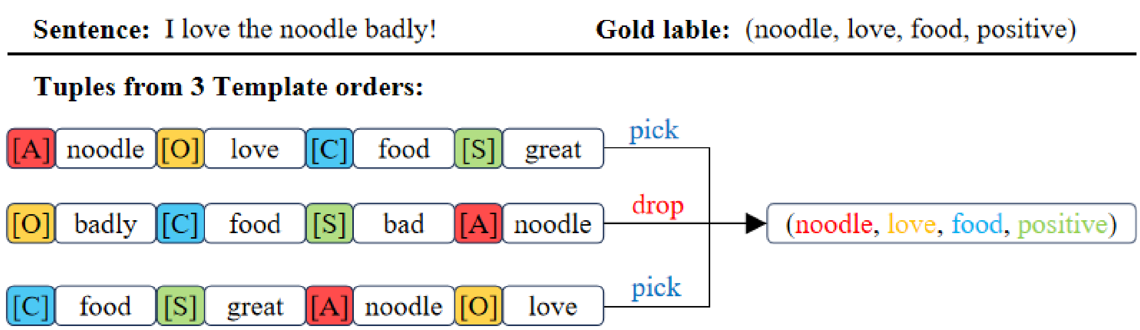

Figure 8: Example of quad correction.

4. Experiments

4.1. Dataset

We assess the performance of our approach using two ASQP datasets: Rest15 and Rest16 (Zhang et al., 2021), as well as two ACOS datasets: Laptop and Restaurant (Cai et al., 2021). Unlike ASQP, the ACOS dataset focuses on implicit aspects and opinion terms, providing a more thorough assessment of our method. To maintain consistency, we adopt the same data split as employed in prior studies. Table 1 shows the dataset statistics.

Table 1: Dataset statistics. E: explicit, I: implicit, A: aspect, O: opinion. E.g., EA&IO represents "explicit aspect, implicit opinion".

	Rest15	Rest16	Restaurant	Laptop
Sentences (Train/Dev/Test)	1580 (834/209/537)	2124 (1264/316/544)	2286 (1531/170/585)	4076 (2934/326/816)
EA&EO Quads	1389(55.66%)	2566(77.88%)	2429(66.40%)	3269(56.77%)
IA&EO Quads	1107(44.34%)	729(22.12%)	530(14.49%)	910(15.80%)
EA&IO Quads	-	-	350(9.57%)	1237(21.48%)
IA&IO Quads	-	-	349(9.54%)	342(5.94%)

4.2. Compared Methods

For extensive evaluation, we have selected multiple robust baseline methods, which can be divided into two types: BERT-based (Devlin et al., 2018) methods and T5-based methods. BERT-based methods include JET (Cai et al., 2021), TAS-BERT (Cai et al., 2021), TASO-BERT (Zhang et al., 2021) and Extract-Classify (Cai et al., 2021). T5-based methods encompass Paraphrase (Zhang et al., 2021), Seq2Path (Mao et al., 2022), DLO (Hu et al., 2022), MVP (Gou et al., 2023) and UAUL (Hu et al., 2023).

4.3. Implement Details

The T5-base model (Raffel et al., 2020) is utilized as our pre-trained model. Basic Training Hyperparameters: epochs = 20, batch size = 16, learning rate = 1e-4. Greedy search is employed for decoding during inference. For dual-sequence data augmentation, we configure the number of templates k to 15, and the Q2S augmented data is twice the volume of the training data. The hyper-parameters in the joint objective are established as $\alpha = \beta = 0.1$, while in supervised contrastive learning $\tau = 0.07$. All experiments were performed using Nvidia RTX 3090 GPU.

5. Results

A Sentiment quad prediction is deemed correct only if every prediction element precisely matches the ground truth labels. Precision rate, recall rate, and F1 score served as the evaluation metrics, with the F1 score being the primary index. The reported results represent averages from five runs with diverse random seed initializations.

5.1. Overall Results

Tables 2 and 3 present the overall performance of the ASQP and ACOS tasks, respectively. Our method outperforms other methods across all four datasets. In particular, compared with the best results achieved by strong baselines, our method demonstrates enhancements of 0.67% and 0.51% on Rest15 and Rest16, respectively, within the ASQP dataset; on the ACOS dataset, our method showcases improvements of 0.92% and 0.49% for Laptop and Restaurant, respectively.

Table 2: Main comparative experimental results on the ASQP dataset. The superior score is in bold, while the next-best is underlined.

Method	Rest15			Rest16		
	P	R	F1	P	R	F1
TASO-BERT	44.24	28.66	34.78	49.73	40.70	44.77
Extract-Classify	35.64	37.25	36.42	38.40	50.93	43.77
GAS	45.31	46.70	45.98	54.54	57.62	56.04
Paraphrase	46.16	47.72	46.93	56.63	59.30	57.93
DLO	47.08	49.33	48.18	57.92	61.80	59.79
MVP	-	-	<u>51.04</u>	-	-	60.39
UAUL	49.12	50.39	49.75	59.02	62.05	<u>60.50</u>
Ours	51.23	52.20	**51.71**	59.98	62.08	**61.01**

Table 3: Main comparative experimental results on the ACOS dataset. The superior score is in bold, while the next-best is underlined.

Method	Laptop			Restaurant		
	P	R	F1	P	R	F1
JET	44.52	16.25	23.81	59.81	28.94	39.01
TAS-BERT	47.15	19.22	27.31	26.29	46.29	33.53
Extract-Classify	45.56	29.48	35.80	38.54	52.96	44.61
Seq2Path	-	-	42.97	-	-	58.41
MVP	-	-	43.92	-	-	<u>61.54</u>
UAUL	44.91	44.01	<u>44.45</u>	61.03	60.55	60.78
Ours	46.04	44.72	**45.37**	63.13	60.96	**62.03**

5.2. Ablation Study

To verify the effectiveness of each module, a systematic ablation study was performed. Table 4 shows the results.

Effect of dual-sequence data augmentation: We initially carried out a validation by employing a single template order as the output sequence while maintaining the original dataset as the input sequence, instead of using dual-sequence data. As indicated in Table 4, the advantages introduced by dual-sequence data augmentation are noteworthy. By integrating multiple template orders and expanding the diversity of inputs, dual-sequence data augmentation proves to be advantageous in enhancing the comprehension of the nature of ASQP tasks.

Effect of contrastive learning: We secondly confirmed the impact by eliminating all supervised contrastive losses and exclusively utilizing the cross-entropy loss of the decoder as the training target. The efficacy of the loss highlighting implicit linguistic phenomena emphasizes the benefits of integrating aspect and opinion terms into supervised contrastive learning objectives for modeling challenging examples.

Effect of prediction normalization strategy: We perform a final assessment to evaluate the impact of the prediction normalization strategy on the model. The assessment compared the performance of a generated model using element correction and quad correction against a model that directly takes the predicted elements and randomly selects a quad from multiple template orders. Experimental results substantiate that the prediction normalization strategy can refine the results.

Table 4: Results of ablation experiments for each component. F1 scores are reported

	Method	Rest15	Rest16	Laptop	Restaurant
	w/o Dual sequence data augmentation	48.17	59.96	44.32	58.26
ours	w/o Contrastive learning	50.44	60.31	43.42	60.83
	w/o Prediction normalization	49.39	59.99	44.96	61.19
	ours	**51.71**	**61.01**	**45.37**	**62.03**

Effect of the value of k: We further explore by varying the value of k, as detailed in Table 5. Initially, as k increases, the F1 score shows an upward trend. However, intriguingly, the F1 score slightly decreases after reaching a certain value. We believe that lower-ranked order permutations might be less effective. When k is too large, it can introduce noise or irrelevant information, which may interfere with the model's learning and lead to decreased performance. Therefore, selecting the appropriate k value is crucial.

Table 5: Effect of the number of views. F1 scores are reported

Dataset	k					
	1	3	8	15	19	24
ASQP-Rest15	48.20	48.85	51.32	51.71	51.47	51.13
ASQP-Rest16	59.02	59.61	60.97	61.01	60.75	60.58

5.3. Case Study

We further investigated some error cases, exemplified by two situations in Figure 9. As illustrated in Case 1, sentences containing connectives pose a challenge because some labels require splitting while others do not, leading to potential segmentation errors. This issue arises because the model may struggle to differentiate when a connective indicates a split versus when it does not, causing inaccuracies in segmentation.

In Case 2, detecting the exact span of an opinion term becomes challenging, particularly when the term is presented as a lengthy text span. This difficulty is due to the model's limitations in handling extended spans and distinguishing between relevant and irrelevant portions of the text, which can result in incorrect boundaries for opinion terms.

For future work, one potential solution could involve enhancing the model's performance in recognizing and appropriately handling different types of connectives through additional training data or refined algorithms specifically targeting these linguistic features. Additionally, developing more sophisticated span detection techniques, such as incorporating context-aware mechanisms or leveraging more advanced natural language processing models, could help in better understanding and processing lengthy text spans.

case 1

Sentence: The chips and salsa are so yummy, and the prices are fabulous.

Gold Label: (chips and salsa, yummy, positive, food quality),
(it, fabulous, positive, restaurant prices)

Prediction: (chips, yummy, great, food quality)✗, (salsa, yummy, great, food quality)✗,
(it, fabulous, great, restaurant prices)✓

case 2

Sentence: The owners are great fun and the beer selection is worth staying for.

Gold Label: (owners, great fun, positive, service general),
(beer selection, worth staying fors, positive, drinks style_options)

Prediction: (owners, great fun, positive, service general)✓,
(beer selection, worth, positive, drinks style_options)✗

Figure 9: Two error cases predicted.

6. Conclusion

In this study, we initiate by introducing dual-sequence data augmentation to diversify input and target expressions, thereby fostering an elevated comprehension of the task by the model. Additionally, we further incorporate an innovative supervised contrastive learning approach developed for specific tasks to refine example representations to realize downstream benefits. Finally, we propose to optimize the generated output through a predictive normalization strategy. A sequence of experiments conducted on four datasets illustrates the better performance of our method in comparison to various baseline approaches, highlighting the effectiveness of each component within our methodology.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 62376062), the Ministry of Education of Humanities and Social Science Project (No. 23YJAZH220, No. 24YJAZH244), the Philosophy and Social Sciences 14th Five-Year Plan Project of Guangdong Province (No. GD23CTS03), and the Guangdong Basic and Applied Basic Research Foundation of China (No. 2023A1515012718).

References

Gianni Brauwers and Flavius Frasincar. A survey on aspect-based sentiment classification. *ACM Computing Surveys*, 55(4):1–37, 2022.

Hongjie Cai, Yaofeng Tu, Xiangsheng Zhou, Jianfei Yu, and Rui Xia. Aspect-category based sentiment analysis with hierarchical graph convolutional network. In *Proceedings of the 28th international conference on computational linguistics*, pages 833–843, 2020.

Hongjie Cai, Rui Xia, and Jianfei Yu. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Annual Meeting of the Association for Computational Linguistics*, pages 340–350, 2021.

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

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

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. Can: Constrained attention networks for multi-aspect 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 4601–4610, 2019.

Mengting Hu, Yike Wu, H. Gao, Yinhao Bai, and Shiwan Zhao. Improving aspect sentiment quad prediction via template-order data augmentation. In *Conference on Empirical Methods in Natural Language Processing*, pages 7889–7900, 2022.

Mengting Hu, Yinhao Bai, Yike Wu, Zhen Zhang, Liqi Zhang, Hang Gao, Shiwan Zhao, and Minlie Huang. Uncertainty-aware unlikelihood learning improves generative aspect sentiment quad prediction. *arXiv preprint arXiv:2306.00418*, 2023.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Vladimir I Levenshtein et al. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*, volume 10, pages 707–710. Soviet Union, 1966.

Dehong Ma, Sujian Li, Fangzhao Wu, Xing Xie, and Houfeng Wang. Exploring sequence-to-sequence learning in aspect term extraction. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 3538–3547, 2019.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. A joint training dual-mrc framework for aspect based sentiment analysis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13543–13551, 2021.

Yue Mao, Yi Shen, Jingchao Yang, Xiaoying Zhu, and Longjun Cai. Seq2path: Generating sentiment tuples as paths of a tree. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2215–2225, 2022.

Samuel Mensah, Kai Sun, and Nikolaos Aletras. An empirical study on leveraging position embeddings for target-oriented opinion words extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9174–9179, 2021.

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

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

Amane Sugiyama and Naoki Yoshinaga. Data augmentation using back-translation for context-aware neural machine translation. In *Proceedings of the fourth workshop on discourse in machine translation (DiscoMT 2019)*, pages 35–44, 2019.

An Wang, Junfeng Jiang, Youmi Ma, Ao Liu, and Naoaki Okazaki. Generative data augmentation for aspect sentiment quad prediction. In *STARSEM*, pages 128–140, 2023.

Jason Wei and Kai Zou. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388, 2019.

Shufang Xie, Ang Lv, Yingce Xia, Lijun Wu, Tao Qin, Tie-Yan Liu, and Rui Yan. Target-side input augmentation for sequence to sequence generation. In *International Conference on Learning Representations*, 2021.

Chengze Yu, Taiqiang Wu, Jiayi Li, Xingyu Bai, and Yujiu Yang. Syngen: A syntactic plug-and-play module for generative aspect-based sentiment analysis. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Mi Zhang and Tieyun Qian. Convolution over hierarchical syntactic and lexical graphs for aspect level sentiment analysis. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 3540–3549, 2020.

Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. Aspect sentiment quad prediction as paraphrase generation. In *Conference on Empirical Methods in Natural Language Processing*, pages 9209–9219, 2021.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 2022.