# Making Better Use of Training Corpus: Retrieval-based Aspect Sentiment Triplet Extraction via Label Interpolation

**Guoxin Yu**[*], **Lemao Liu**[†], **Haiyun Jiang**, **Shuming Shi**, **Xiang Ao**[†]

Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China.
University of Chinese Academy of Sciences, Beijing 100049, China.
Tencent AI Lab, China.
Institute of Intelligent Computing Technology, Suzhou, CAS.
{yuguoxin20g,aoxiang}@ict.ac.cn
{redmondliu, haiyunjiang, shumingshi}@tencent.com

## Abstract

In this paper, we aim to adapt the idea of retrieval-based neural approaches to the Aspect Sentiment Triplet Extraction (ASTE) task. Different from previous studies retrieving semantic similar neighbors, the ASTE task has its specialized challenges when adapting, i.e., the purpose includes predicting the sentiment polarity and it is usually aspect-dependent. Semantic similar neighbors with different polarities will be infeasible even counterproductive. To tackle this issue, we propose a retrieval-based neural ASTE approach, named RLI (Retrieval-based Aspect Sentiment Triplet Extraction via Label Interpolation), which exploits the label information of neighbors. Given an aspect-opinion term pair, we retrieve semantic similar triplets from the training corpus and interpolate their label information into the augmented representation of the target pair. The retriever is jointly trained with the whole ASTE framework, and neighbors with both similar semantics and sentiments can be recalled with the aid of this distant supervision. In addition, we design a simple yet effective pre-train method for the retriever that implicitly encodes the label similarities. Extensive experiments and analysis on two widely-used benchmarks show that the proposed model establishes a new state-of-the-art on ASTE.

## 1 Introduction

As an emerging sub-task of Aspect-based Sentiment Analysis (ABSA), Aspect Sentiment Triplets Extraction (ASTE) extracts all sentimental triplets of a given sentence. Every triplet contains three elements, namely aspect terms, opinion terms, and their corresponding sentiment polarities. For example, in the sentence "*Great food but the service is horrible.*", ASTE attempts to identify (*food*, *great*, *positive*) and (*service*, *horrible*, *negative*).

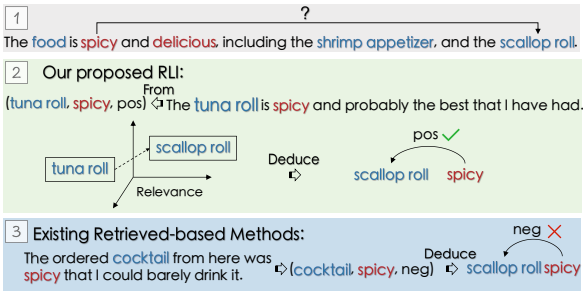



Figure 1: An example sentence (in gray block). RLI retrieves triplets considering both label similarity and semantic similarity (in green block). Existing Retrieval-based Methods fetch similar sentences only according to semantic similarities (in blue block).

Since the sentiment polarity of a triplet is aspect-dependent and determined by the corresponding opinion terms, establishing reciprocity among elements within the triplets could yield easier sentiment predictions. Following this idea, existing work devised advanced methods to explore the correlation between the aspect and the opinion terms. To name some, Xu et al. (2020b); Wu et al. (2020) proposed new tagging schemes to build connections among three elements within a sentence. Li et al. (2019); Zhang et al. (2020); Xu et al. (2021); Zhao et al. (2022) designed various end-to-end frameworks to explore relations among elements by sub-task interaction mechanisms. Chen et al. (2021a); Liu et al. (2022) matched the elements by machine reading comprehension.

Despite their effectiveness, existing methods may fail to clarify the intricate relationships among el-

[*]Work done while this author was an intern at Tencent.
[†]Corresponding authors.



*Findings of the Association for Computational Linguistics: ACL 2023*, pages 4914–4927
July 9-14, 2023 

ements in some challenging cases, e.g., sentences with uncommon aspect/opinion terms, or the aspect and opinion terms are distant from each other. For example, as shown in the first sentence in Fig. 1, "*scallop roll*" may be difficult to be extracted because "*scallop*" is an uncommon aspect word that rarely appeared in the training set. And it is intractable to connect "*scallop roll*" with "*spicy*" due to the long distance between them. These make it challenging to extract the triplet (*scallop roll*, *spicy*, *positive*).

To tackle the above issues, we attempt to apply retrieval-based models to ASTE, which have shown strength in several NLP tasks (Cai et al., 2022; Shang et al., 2021; Xu et al., 2020a) such as language model, machine translation, etc. Their basic idea is to retrieve semantic similar neighbors from training corpus or external data to improve the model's robustness towards infrequent data points (Meng et al., 2021; Li et al., 2022).

However, the ASTE task has its specialized challenges when adapting, i.e., its purpose includes predicting the sentiment polarities and it is usually aspect-dependent. For example, the two triplets (*battery*, *long*, *positive*) and (*boot-time*, *long*, *negative*) have the same opinion word but opposite aspect-level sentiment. Hence, it may derive a drawback of the conventional retrieval-based model: *the semantic similar neighbors with different sentiments may be infeasible even counterproductive*.

To remedy the challenges, we propose a retrieval-based neural ASTE approach, named RLI (Retrieval-based Aspect Sentiment Triplet Extraction via Label Interpolation), which can exploit the label information of neighbors. We first collect all triplets from the training set to construct a knowledge store and detect all candidate aspect-opinion pairs. For each pair, we retrieve semantic similar triplets from the constructed store. Then we interpolate their label information into the augmented representation of the candidate pair to predict the final sentiment. Unlike existing retrieval-based methods which retrieve neighbors only according to semantic similarities, we jointly train the retriever and the triplets extraction model such that the neighbors with both similar semantics and sentiment could be fetched. In addition, we propose a simple yet effective method to pre-train the proposed retriever, which could encode label information implicitly by using pseudo-labeled data before the joint training.

Exhibiting our idea by an example in Fig. 1, RLI could retrieve a relevant triplet (*tuna roll*, *spicy*, *positive*) for the candidate pair (*scallop roll*, *spicy*) (cf. the green block). By high relevance between "*tuna roll*" and "*scallop roll*", we can infer that (*scallop roll*, *spicy*) could be a valid pair and deduce its *positive* polarity. While, as shown in the blue block, the conventional retrieval-based methods may likely fetch a triplet (*cocktail*, *spicy*, *negative*), which has an opposite sentiment and may give false guidance.

Extensive experimental results and analysis on two standard datasets for ASTE show that the proposed model establishes a new state-of-the-art on the ASTE task and performs well on challenging examples.

## 2 Related Work

### 2.1 Aspect Sentiment Triplet Extraction

Recall that the key of resolving ASTE is to establish reciprocity among three elements within the triplets. Early studies (Peng et al., 2020; Huang et al., 2021) designed pipeline models to extract these elements successively and group them into triplets, which suffered from error propagation and aggregation. To avoid such obstacles, Xu et al. (2020b); Wu et al. (2020); Chen et al. (2020) proposed novel tagging schemes to connect the elements and train the models in an end-to-end fashion. Zhang et al. (2020); Zhao et al. (2022); Huan et al. (2022) devised multi-task frameworks to exploit the interactions among various sub-tasks. Chen et al. (2021b, 2022) constructed the given text to different graphs and fully utilized the relations between words.

Besides, some studies gradually put forward new paradigms for ASTE. Yu et al. (2021) regarded the aspect and opinion terms as arguments of the expressed sentiment in a reinforcement learning framework. Chen et al. (2021a); Liu et al. (2022) converted ASTE to a machine reading comprehension problem. Xu et al. (2021) considered ASTE as a span prediction problem.

Recently, a series of generative methods (Zhang et al., 2021a; Yan et al., 2021; Zhang et al., 2021b; Gao et al., 2022) come to the fore, which regarded ASTE as a text generation problem achieved superior performance. Nevertheless, all the above methods may become fragile in sentences with multiplex triplets, where aspect terms or opinion terms are uncommon, correlations are complicated or sentiments are unclear.

### 2.2 Retrieval Augmented Methods

Prior studies have proved that retrieval-based methods could improve performance across a variety of NLP tasks. They retrieved similar neighbors from external knowledge to facilitate the model's robustness towards infrequent data points, which has been applied in question answering (Li et al., 2020; Karpukhin et al., 2020), neural machine translation (Tu et al., 2018; Xu et al., 2020a; Shang et al., 2021; Cai et al., 2022), language modeling (Guu et al., 2017; Khandelwal et al., 2019), dialog generation (Fan et al., 2020; Thulke et al., 2021), and etc. Due to the considerable computational cost of retrieving from large-scale corpora, Wang et al. (2022) proposed to fetch data most similar to the input only from the training data. They simply concatenated them with input to achieve significantly better performance on many natural language processing tasks. Apart from their effectiveness, these methods only consider semantic information but ignore the label similarity, which may retrieve triplets with similar semantic yet opposite sentiments. Hence they might render ineffective for solving ASTE.

## 3 Methodology Overview

### 3.1 Task Definition

Suppose $\mathbf{X} = \{x_1, x_2, \cdots, x_n\}$ is a sentence with $n$ words, and each span is represented by $S = (S_1, S_2)$ where $S_1$ and $S_2$ denote the start and end positions of the span. Typically, ASTE is treated as a span extraction task: given a sentence $\mathbf{X}$, ASTE aims to extract a triplet set $\mathcal{T} = \{\langle A, O, y\rangle\}$, where $A = (A_1, A_2)$ and $O = (O_1, O_2)$ respectively denote the spans of an aspect term and an opinion term, and $y \in \{\text{positive}, \text{neutral}, \text{negative}\}$ is the sentiment polarity of the triplet. It is worth noting that each triplet $\langle A, O, y\rangle$ is dependent on a sentence $\mathbf{X}$, but we only mention $\langle A, O, y\rangle$ and skip its corresponding sentence $\mathbf{X}$ for brevity.

### 3.2 Model Overview

The proposed approach consists of four distinct modules namely: *triplets store construction*, *candidate aspects and opinions detection*, *triplet-based retrieval*, and *triplets extraction*, which are shown in Figure 2. The first module constructs a triplets store for triplets-level retrieval (§4.1). The second one extracts the candidate aspect-opinion span pairs based on a span-level sequence labeling method (§4.2). The third phase retrieves neighbors for each candidate aspect-opinion pair from the constructed store(§4.3). Moreover, we interpolate the representations and label information of the retrieved triplets with candidate pairs and further predict their final sentiment polarities(§4.4). Finally, we present how to pre-train the retriever to implicitly encode label information and jointly train the retrieval model and ASTE model(§5).

## 4 RLI Model

### 4.1 Triplet Store Construction

ASTE pays more attention to the aspect terms and opinion terms than other words in the given sentence $\mathbf{X}$. To this end, we construct a knowledge store $\mathcal{M}$ containing all the triplets in the training set, instead of accommodating all the sentences.

To represent each triplet $\langle A, O, y\rangle$ in $\mathcal{M}$, we employ BERT to define its key and value as follows. We first use BERT to get the representations $\mathbf{H} = \{h_1, h_2, \cdots, h_n\}$ for each word in the sentence $\mathbf{X}$. Then we define the representations of aspect $A$ and opinion terms $O$ by $E_A$ and $E_O$:

$$\begin{aligned} E_A &= h_{A_1} \oplus h_{A_2} \oplus f_{\text{span}}(A_2 - A_1 + 1), \\ E_O &= h_{O_1} \oplus h_{O_2} \oplus f_{\text{span}}(O_2 - O_1 + 1), \end{aligned} \tag{1}$$

where $\oplus$ is the concatenation of two vectors, $f_{\text{span}}$ works as a trainable feature extractor related to the span width following Xu et al. (2021). Afterward, we concatenate the above spans and a trainable sentiment embedding together to represent each triplet $\langle A, O, y\rangle$ as a key-value $\langle K, V\rangle$ pairs:

$$\begin{aligned} K &= E_A \oplus E_O, \\ V &= f_{\text{sentiment}}(y), \end{aligned} \tag{2}$$

where $f_{\text{sentiment}}$ is a learnable conversion function of a sentiment polarity $y$. Note that $K$ and $V$ encode representation information and label information of $\langle A, O, y\rangle$, respectively.

Finally, the triplet store $\mathcal{M} = \{\langle A^i, O^i, y^i\rangle | i \in [1, |\mathcal{M}|]\}$ can be actually represented by a set of key-value pairs $\mathcal{M} = \{\langle K^i, V^i\rangle | i \in [1, |\mathcal{M}|]\}$.

### 4.2 Candidate Aspects & Opinions Detection

Similar to Xu et al. (2021), during the inference stage, given a sentence $\mathbf{X}$, we firstly extract all possible candidate spans which may be either aspect or opinion span, and then we employ a classifier to predict whether a candidate span $S$ is an aspect, an opinion, or an invalid span. Specifically, we first

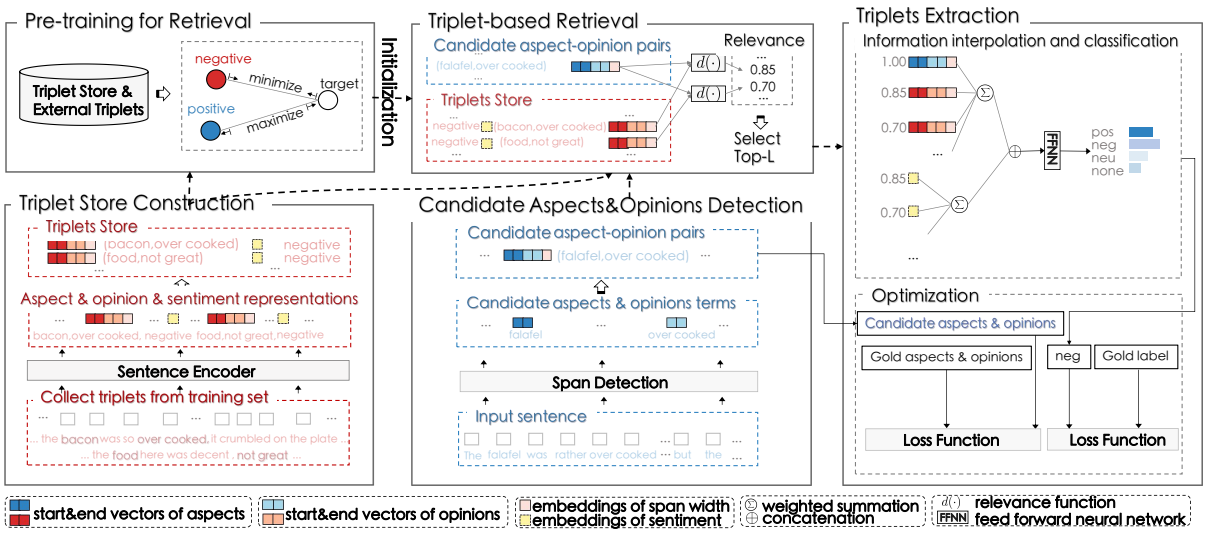



Figure 2: Model Overview. We first pre-train the retriever by using external unlabeled data and initialize the relevance scores. Then we jointly train all the modules on the standard ASTE datasets.

use BERT to obtain the representation $E_S$ for each candidate span $S$ as $E_A$ and $E_O$ in Eq. (1). Then a detection model $P_{\text{det}}$ is used to detect the type of the candidate span $S$: aspect, opinion, or invalid span.

$$\begin{aligned} E_S &= h_{S_1} \oplus h_{S_2} \oplus f_{\text{span}}(S_2 - S_1 + 1), \\ P_{\text{det}}(c|S, \mathbf{X}) &= \text{softmax}(g(E_S))[c], \\ c &\in \{\text{aspect}, \text{opinion}, \text{invalid}\}, \end{aligned} \tag{3}$$

where $g$ is a feed-forward neural network, and $[c]$ denotes taking the probability for the dimension corresponding to the type $c$. Theoretically, there are $\frac{n(n+1)}{2}$ spans in the sentence $\mathbf{X}$, but it is too slow to make predictions for all possible spans. In practice, we limit the maximum length of spans thus discarding some excessively long spans.

According to Eq. (3), we select the top $\mathcal{K}$ candidate aspect spans and opinion spans. Subsequently, we pair candidate aspect spans and opinion spans to create $\mathcal{K}^2$ candidate aspect-opinion pairs. Suppose $\langle A, O\rangle$ denotes each candidate aspect-opinion pair, and it can be represented by $K = E_A \oplus E_O$ as defined in Eq. (2).

### 4.3 Triplet-based Retrieval

For each candidate aspect-opinion pair $\langle A, O\rangle$, we retrieve the $L$ most relevant triplets from the constructed store by a relevance function between the pair $\langle A, O\rangle$ and each triplet $\langle A^i, O^i, y^i\rangle$ from triplet store $\mathcal{M}$. Formally, the relevance function $d$ between $\langle A, O\rangle$ and $\langle A^i, O^i\rangle$ is defined as:

$$d(A, O; A^i, O^i) = K^\top \mathbf{W} K^i, \tag{4}$$

where $\mathbf{W}$ is trainable parameters, $K$ is the representation of the candidate pair $\langle A, O\rangle$ and $K^i$ is the representation of each aspect-opinion pair $\langle A^i, O^i\rangle$ in $\mathcal{M}$.

According to the relevance function $d$, we select the top-$L$ triplets denoted by $\mathcal{M}(A, O) = \{\langle A^l, O^l, y^l\rangle | l \in [1, L]\}$ with the highest relevance scores in $\mathcal{M}$, which will be further used as memory to extract the triplet in the next subsection.

### 4.4 Triplets Extraction

So far, we have acquired the representations of $\mathcal{K}^2$ candidate aspect-opinion pairs and their similar triplets by retrieval. For each aspect-opinion pair $\langle A, O\rangle$, recall $\mathcal{M}(A, O) = \{\langle A^l, O^l, y^l\rangle | l \in [1, L]\}$ denotes the retrieved triplets. We interpolate both the representation and label information of the retrieved triplets to predict the polarity of $\langle A, O\rangle$. Specifically, we aggregate the dense representations of each candidate pair and its retrieved triplets by using an attention model defined by $d$.

$$\begin{aligned} h(A, O) &= (K + \sum_{l=1}^{L} \alpha_l K^l) \oplus \sum_{l=1}^{L} \alpha_l V^l, \\ \alpha_l &= \frac{\exp\left(d(A, O; A^l, O^l)\right)}{\sum_{j=1}^{L} \exp\left(d(A, O; A^j, O^j)\right)}, \end{aligned} \tag{5}$$

where $K$ and $K^l$ are the representations of $\langle A, O\rangle$ and $\langle A^l, O^l\rangle$ as defined in Eq. (2), respectively. $V^l$ is the sentiment label embedding of the retrieval triplets $\langle A^l, O^l, y^l\rangle$.

Next, we predict the final sentiment polarity of $\mathcal{K}^2$ pairs by a neural model. For each candidate aspect-opinion $\langle A, O\rangle$ pair,

$$\begin{aligned} P_{\text{ext}}(y|A, O, \mathbf{X}) &= \text{softmax}(F(h(A, O)))[y], \\ y &\in \{\text{positive}, \text{negative}, \text{neutral}, \text{none}\}, \end{aligned} \tag{6}$$

where $F$ is a feed-forward neural network, and "none" denotes the aspect-opinion pair is not a meaningful pair with definite sentiment polarity. In this way, we can not only achieve aspect-opinion pair extraction by judging whether the label is "none", but also extract triplets by identifying valid pairs with definite sentiments.

## 5 Training

### 5.1 Pre-training for Retrieval

To make the retriever memorize the sentiment similarity information in advance, we propose a simple yet effective method to pre-train the retriever by using external unlabeled data, which prompts the retrieved triplets to have similar sentiments.

Specifically, over the external unlabeled data, we first use the *candidate aspect & opinion detection* module to extract aspect-opinion pairs. Then a feed-forward neural network is used to predict if they are valid and further determine their sentiment polarities. In this way, we obtain a set of triplets $\{\langle A, O, y\rangle\}$ from the external data, where $y$ is the pseudo polarity predicted by the neural network. We call them pseudo-labeled data. Furthermore, for each triplet $\langle A, O, y\rangle$, we randomly select two triplets $\langle A', O', y\rangle$ and $\langle A'', O'', y'\rangle$ which suffice to the following constraints: the former is with the same polarity and the other is with a different sentiment polarity, i.e., $y' \neq y$. Inspired by contrastive learning, we optimize a ranking loss $\mathcal{L}_{\text{pre}}$ to maximize the relevance score between triplets with the same sentiments meanwhile minimize that between triplets with opposite sentiments.

$$\mathcal{L}_{\text{pre}} = d(A, O; A', O')^2 - \big(1 - d(A, O; A'', O'')\big)^2,$$

$d(\cdot)$ is the relevance function defined in Eq. (4) which measures the similarity between two triplets. After pre-training and initializing the relevance scores, we jointly train all the proposed modules. Due to the pre-training, the retriever encodes sentiment similarities and RLI can retrieve helpful triplets to assist the sentiment prediction of the candidate aspect-opinion pair from a warm start.

### 5.2 Joint Training

For a sentence $\mathbf{X}$, suppose $\mathcal{S}(\mathbf{X})$ denotes a span pool including $\mathcal{K}$ candidate spans for $\mathbf{X}$. The standard practice to train ASTE models relies on manually annotated data. That is, for each span $S \in \mathcal{S}(\mathbf{X})$, there is a golden label $c \in \{\text{aspect, opinion, invalid}\}$; and for each aspect-opinion pair $\langle A, O\rangle \in \mathcal{S} \times \mathcal{S}$, there is a golden label $y \in \{\text{positive, negative, neutral, none}\}$. We employ the standard cross entropy to train the candidate aspect terms and opinion terms detection model $P_{\text{det}}$, triplets extraction model $P_{\text{ext}}$ as well as the retrieval similarity in a joint manner as follows:

$$\begin{aligned}\mathcal{L}_{\text{det}} &= -\sum_{\mathbf{X}} \sum_{S \in \mathcal{S}(\mathbf{X})} \log P_{\text{det}}(c|S, \mathbf{X}), \\ \mathcal{L}_{\text{ext}} &= -\sum_{\mathbf{X}} \sum_{A,O \in \mathcal{S}(\mathbf{X})} \log P_{\text{ext}}(y|A, O, \mathbf{X}),\end{aligned} \tag{7}$$

where $\mathbf{X}$ is over a training set with manually annotated golden triplets. The overall loss is calculated as a weighted sum of the above two loss functions.

$$\mathcal{L} = \mathcal{L}_{\text{det}} + \alpha \cdot \mathcal{L}_{\text{ext}}, \tag{8}$$

where $\alpha > 0$ is a hyperparameter to trade off both loss terms. In each iteration, we first perform the triplets retrieval based on the last-time iteration. Next, we extract the triplets with the help of retrieved triplets and update the parameters in the current iteration. Note that the relevance scores are used to define the representation $h(A, O)$ through the attention in Eq. (5), on which the classifier $P_{\text{ext}}$ is based. Therefore, minimizing $\mathcal{L}$ actually optimizes the three models, i.e., the detection model in Eq. (3), the triplets extraction model in Eq. (6) and the retrieval similarity in Eq. (4).

It is notable that we don't use the external pseudo-labeled data to train the joint model but only pre-train the retriever in §5.1: the triplet store for retrieval consists of all those triplets created from the original training data for ASTE and the loss function in Eq. (8) is minimized over the original training data as well.

## 6 Experiment

### 6.1 Settings

**Datasets.**[1] To evaluate our method as comprehensively as possible, we conduct experiments on $\mathcal{D}_a$ (Peng et al., 2020) and $\mathcal{D}_b$ (Xu et al., 2020b). Both of them contain 3 datasets in the restaurants domain and 1 dataset in the electronics domain. For pre-training, we use two external datasets from He et al. (2018), one is from the Yelp domain, and the other is from the Amazon electronics domain.

[1] See Appendix A for more details of datasets.

| Model | Res14 | | Lap14 | | Res15 | | Res16 | |
|---|---|---|---|---|---|---|---|---|
| | *Pair* | *Triplet* | *Pair* | *Triplet* | *Pair* | *Triplet* | *Pair* | *Triplet* |
| WhatHowWhy$^{\diamondsuit}$ | 56.10 | 51.89 | 53.85 | 43.50 | 56.23 | 46.79 | 60.04 | 53.62 |
| CMLA+$^{\diamondsuit}$ | 48.95 | 43.12 | 44.10 | 32.90 | 44.60 | 35.90 | 50.00 | 41.60 |
| RINANTE+$^{\diamondsuit}$ | 46.29 | 34.03 | 29.70 | 20.00 | 35.40 | 28.00 | 30.70 | 23.30 |
| Unified+$^{\diamondsuit}$ | 55.34 | 51.68 | 52.56 | 42.47 | 56.85 | 46.69 | 53.75 | 44.51 |
| Dual-MRC$^{\diamondsuit}$ | 74.93 | 70.32 | 63.37 | 55.58 | 64.97 | 57.21 | 75.71 | 67.40 |
| Generative$^{\triangledown}$ | 77.68 | 72.46 | 66.11 | 57.59 | 67.98 | 60.11 | 77.38 | 69.98 |
| GAS$^{\sharp}$ | – | 70.20 | – | 54.50 | – | 59.10 | – | 65.00 |
| LEGO$^{\sharp}$ | – | 72.60 | – | 59.50 | – | 63.20 | – | 71.50 |
| JET$^{t}_{M=6}$ $^{\triangledown}$ | – | 60.41 | – | 46.65 | – | 53.68 | – | 63.41 |
| JET $^{o}_{M=6}$ $^{\triangledown}$ | – | 63.92 | – | 50.00 | – | 54.67 | – | 62.98 |
| SPAN* | 78.62 | 73.96 | 69.48 | 60.59 | 71.56 | 64.50 | 78.85 | 70.48 |
| RLI(Ours) | **79.92** | **74.98** | **70.27** | **61.97** | **72.66** | **65.71** | **81.29** | **73.33** |

Table 1: *F1 score for Pair* and *Triplet extraction* on $\mathcal{D}_a$. Results with $^{\diamondsuit}$, $^{\sharp}$, and $^{\triangledown}$ are taken from Mao et al. (2021), Gao et al. (2022) and their original papers. Results with * are reproduced by ourselves with the same experiment settings. All the models are based on BERT-base or BART-base (for generative methods). The best values are in bold numbers. Dashed lines separate baselines according to types.

**Baselines.** We compared our method [2] with various baselines, which are evaluated on $\mathcal{D}_a$ and $\mathcal{D}_b$.

- **Pipeline models**: WhatHowWhy (Peng et al., 2020), CMLA+ (Wang et al., 2017), RINANTE+ (Dai and Song, 2019), Unified+ (Li et al., 2019), and TOP (Huang et al., 2021).
- **MRC based methods**: Dual-MRC (Mao et al., 2021), BMRC (Chen et al., 2021a).
- **Reinforce learning based methods**: RL (Yu et al., 2021).
- **Generative models**: Generative (Yan et al., 2021), GAS (Zhang et al., 2021b), LEGO (Gao et al., 2022).
- **End-to-end models**: JET (Xu et al., 2020b), OTE-MTL (Zhang et al., 2020), GTS (Wu et al., 2020), SPAN (Xu et al., 2021), and EMC-GCN (Chen et al., 2022).

**Evaluation Metrics.** We implement five metrics to evaluate our proposed model: *F1 score for pair extraction*, *Precision*, *Recall*, *F1 score for triplet extraction*, and *Retrieval Accuracy*. Particularly, *Retrieval Accuracy* is the proportion of correct triplets retrieved, of which sentiment polarities are consistent with the gold label of the candidate aspect-opinion pair. We select the best model based on the *F1 score for triplet extraction* on the development set. The reported scores are the average of 5 runs with distinct random seeds.

[2] See appendix B for experimental details and parameters. We will release our code after the double-blind review.

### 6.2 Main Results

We compare our method with various baselines on $\mathcal{D}_a$ and $\mathcal{D}_b$ comprehensively. The results are reported in Table 1 and Table 2, respectively. Firstly, in Table 1, our model outperforms all the compared models on the *F1 score for pair extraction*. We speculate that our model could judge if a candidate aspect-opinion pair is valid or not by observing the relevance scores between a pair and its retrieved triplets. Secondly, as the two tables show, our model considerably improves *precision*, *recall*, and *F1 score for triplet extraction* compared to pipeline and end-to-end models over most datasets. This indicates that relevant triplets conduce to exploit the interactions between aspect terms and opinion terms. Thirdly, we observe that our model even achieves more competitive results than emerging generative methods, of which backbones may be stronger (T5 (Raffel et al., 2020) or BART (Lewis et al., 2019)). Such results suggest the superiority of retrieval-based methods.

### 6.3 Ablation Test

In Table 3, we perform an ablation study and report the results on the development and test set of $\mathcal{D}_b$ to investigate the effects of key modules.

On the one hand, we compute the relevance scores according to semantic similarity. Then we execute the model to retrieve triplets based on the fixed semantic similarity to get the results "w/o joint". It follows that the *F1 scores for triplets ex-*

| Model | Res14 | | | Lap14 | | | Res15 | | | Res16 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 |
| WhatHowWhy$^{\diamondsuit}$ | 43.24 | 63.66 | 51.46 | 37.38 | 50.38 | 42.87 | 48.07 | 57.51 | 52.32 | 46.96 | 64.24 | 54.21 |
| TOP$^{\sharp}$ | 63.59 | 73.44 | 68.16 | 57.84 | **59.33** | 58.58 | 54.53 | 63.30 | 58.59 | 63.57 | 71.98 | 67.52 |
| BMRC$^{\sharp}$ | 72.17 | 65.43 | 68.64 | **65.91** | 52.15 | 58.18 | 62.48 | 55.55 | 58.79 | 69.87 | 65.68 | 67.35 |
| RL$^{\sharp}$ | 70.60 | 68.65 | 69.61 | 64.80 | 54.99 | 59.50 | 65.45 | 60.29 | 62.72 | 67.21 | 69.69 | 68.41 |
| GAS$^{\flat}$ | – | – | 72.16 | – | – | 60.78 | – | – | 62.10 | – | – | 70.10 |
| OTE-MTL$^{\sharp}$ | 63.07 | 58.25 | 60.56 | 54.26 | 41.07 | 46.75 | 60.88 | 42.68 | 50.18 | 65.65 | 54.28 | 59.42 |
| GTS$^{\diamondsuit}$ | 67.76 | 67.29 | 67.50 | 57.82 | 51.32 | 54.36 | 62.59 | 57.94 | 60.15 | 66.08 | 69.91 | 67.93 |
| JET$^{t}_{M=6}$ $^{\diamondsuit}$ | 63.44 | 54.12 | 59.41 | 53.53 | 43.28 | 47.86 | **68.20** | 42.89 | 52.66 | 65.28 | 51.95 | 63.83 |
| JET$^{o}_{M=6}$ $^{\diamondsuit}$ | 70.56 | 55.94 | 62.40 | 55.39 | 47.33 | 51.04 | 64.45 | 51.96 | 57.53 | 70.42 | 58.37 | 63.83 |
| SPAN $^{\diamondsuit}$ | 72.89 | 70.89 | 71.85 | 63.44 | 55.84 | 59.38 | 62.18 | 64.45 | 63.27 | 69.45 | 71.17 | 70.26 |
| EMC-GCN $^{\triangledown}$ | 71.21 | **72.39** | 71.78 | 61.70 | 56.26 | 58.81 | 61.54 | 62.47 | 61.93 | 65.62 | 71.30 | 68.33 |
| RLI (Ours) | **77.46** | 71.97 | **74.34** | 63.32 | 57.43 | **60.96** | 60.08 | **70.66** | **65.41** | **70.50** | **74.28** | **72.34** |

Table 2: Results on $\mathcal{D}_b$. P, R, F1 represent *precision, recall, F1 scores for triplets extraction*. Results with $^{\diamondsuit}$, $^{\sharp}$, $^{\flat}$, $^{\triangledown}$ are from Xu et al. (2020b), Yu et al. (2021), Zhang et al. (2021b), and their original paper, respectively. All the results are based on BERT-base except that WhatHowWhy and OTE-MTL are based on Glove. GAS is based on T5 (Raffel et al., 2020). The best results are in bold numbers. Dashed lines separate baselines according to types.

| Dataset | Model | | Dev F1 | Test F1 |
|---|---|---|---|---|
| Res14 | w/o joint | | 66.85 | 72.07 |
| | joint | w/o sentiment | 67.55 | 73.70 |
| | | w/o pre-training | 67.12 | 72.58 |
| | | full model | **68.00** | **74.34** |
| Lap14 | w/o joint | | 60.03 | 60.33 |
| | joint | w/o sentiment | 61.90 | 60.54 |
| | | w/o pre-training | 61.06 | 60.02 |
| | | full model | **62.55** | **60.96** |
| Res15 | w/o joint | | 70.83 | 63.99 |
| | joint | w/o sentiment | 71.54 | 65.04 |
| | | w/o pre-training | 71.24 | 64.48 |
| | | full model | **72.21** | **65.41** |
| Res16 | w/o joint | | 70.47 | 70.30 |
| | joint | w/o sentiment | 71.44 | 71.39 |
| | | w/o pre-training | 70.75 | 71.69 |
| | | full model | **73.04** | **72.34** |

Table 3: Ablation test. The displayed scores are *F1 score for triplets extraction* on $\mathcal{D}_b$.

*traction* over most datasets decrease by $1\% - 2\%$, which proves that retrieving triplets only based on the semantic similarities is infeasible even counterproductive. However, joint training of the retriever and ASTE modules could dynamically optimize the retrieved triplets for better ASTE.

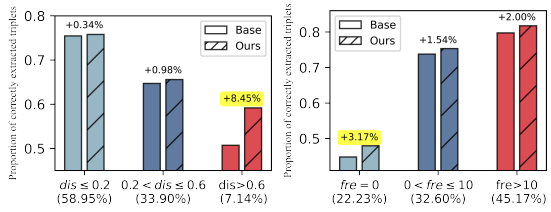
On the other hand, we evaluate two ablated models under joint training. First, we amputate the label information $\sum_{l=1}^{L} \alpha_l V^l$ in Eq. (5) to obtain model "w/o sentiment". Its degraded performance confirms the importance of sentiment label information. Second, we remove the retriever pre-training and jointly train the full model to obtain results "w/o pre-training". By comparison, we find that the pre-training increased the F1 scores, which verifies that pre-training encodes label similarity and improves the quality of retrieval. It makes the label information of retrieved triplets more similar to the gold sentiment polarities and thus achieves better sentiment prediction performance.

| Models | Base | Base+Aug | Ours |
|---|---|---|---|
| Avg. F1 | 66.19 | 66.99(+0.80) | **68.26**(+2.07) |

Table 4: Average *F1 score for triplets extraction* on $\mathcal{D}_b$.

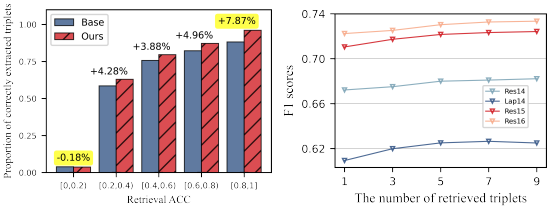
### 6.4 Auxiliary Experiment

In order to prove the improvement of our model derives from the triplets retrieval instead of external augmented data, we conduct an auxiliary experiment and display the average *F1 scores for triplet extraction* in Table 4. Specifically, we remove the retrieval module from our full model to obtain a **Base** model. Then we pre-train the Base model on the pseudo-labeled data and fine-tune it on the original $\mathcal{D}_b$ and the results are denoted as **Base+Aug**. As Table 4 shown, even if Base+Aug gets $0.8$ gain, our model achieves a higher $2.07$ improvement compared to Base. Since we didn't use external data for ASTE's joint training in our method, the results reveal that the capabilities of our model are not from the external data but mainly come from the assistance of triplets retrieval.




(a) (b)

Figure 3: Results on Res14 of $\mathcal{D}_b$.




(a) (b)

Figure 4: Effects of retrieval on $\mathcal{D}_b$.

# 7 Analysis

## 7.1 Inference Results Analysis

To prove the advantage of our method in dealing with challenging cases, we execute an in-depth study to analyze the results of triplets extraction from two perspectives: $dis$, the distance between aspect and opinion terms, and $fre$, the frequency of aspect/opinion terms appearing in the training set,

$$dis = \frac{|i_a - i_o|}{n}, \qquad fre = min(fre_a, fre_o), \tag{9}$$

where $i_a$ and $i_o$ represent the start indexes of the aspect and opinion term, $n$ is the length of the sentence, $fre_a$ and $fre_o$ are times of the aspect and opinion term appear in the training set.

Firstly, according to $dis$, we divided all the gold triplets into three groups and compared the proportion of triplets with different $dis$ correctly extracted by Base (declared in §6.4) and our model. As Fig. 3(a) shows, our model extracted more triplets with $dis > 0.6$ successfully. This means that the Base model may fail to connect the aspect term with a correct faraway opinion term. Nevertheless, our model could reduce the influence of long-distance by referring to relevant triplets.

Secondly, in Fig. 3(b) we categorized all the triplets into three groups by the frequency $fre$ and find that our model could extract more triplets containing aspect/opinion terms that never appear in the training set ($fre = 0$). We conjecture that our model could find them by imitating similar triplets. As a result, we conclude that our model can solve such tricky cases better.

## 7.2 Sensitivity Analysis

We perform a sensitivity analysis to determine the effects of retrieval accuracy and the number of retrieved triplets. According to the triplets' retrieval accuracy, we put all the triplets into different buckets and compare the proportion of triplets correctly extracted by Base and our model over them. In Fig. 4(a), when the accuracy is in $[0.8, 1]$, the improvements of our model are more significant. Unfortunately, when the accuracy falls into $[0, 0.2]$, our model is even slightly weak. This ensures that our model improves ASTE by retrieving triplets with the same sentiment polarities as the gold sentiments. The more triplets with the same sentiments retrieved, the greater their auxiliary function.

Besides, we investigate the effects of the number $L$ of the retrieved triplets in Fig. 4(b). It is noted that the *F1 score for triplets extraction* increases with $L$. But if $L$ is too large, the computational complexity will increase rapidly while the performance improvement is weak. So we set $L$ to 5 to obtain a trade-off between complexity and performance.

## 7.3 Case Study

To better understand the effectiveness of retrieved triplets, we empirically perform a case study on $\mathcal{D}_b$ in Fig. 5. BEFORE denotes extracted results of Base model (declared in §6.4), AFTER denotes the results of our full model, and RETRIEVED is our model's top-1 retrieved triplets and the sentences they come from. These cases demonstrate that the retrieved triplets could extract aspect-opinion pair with long-distance and overcome the problems of uncommon aspect/opinion terms with low frequency in the training set.

# 8 Conclusion

In this paper, we proposed a retrieval-based ASTE approach name RLI, which could exploit the sentiment information of neighbors to solve challenging cases in ASTE. A retriever fetching both semantic and sentiment-similar triplets is devised, and we jointly train the retriever with the ASTE framework to remedy the specialized challenges when adapting the retrieve-based methods in aspect-level sentiment analysis tasks. In addition, we proposed

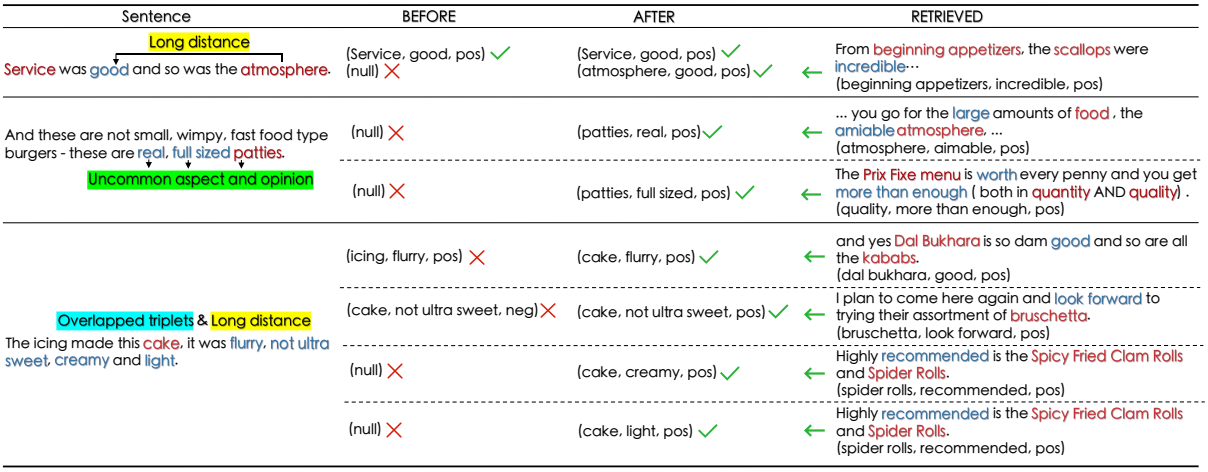



Figure 5: Case Study. "Long distance" denotes the distance between aspect and opinion terms is considerably long. "Uncommon aspect/opinion" represents that the term appears in the training set once or less. "Overlapped triplets" denotes there are multiple triplets containing the same aspect or opinion in the sentence.

a simple yet effective pre-train method for the retriever to implicitly encode the label similarities. Extensive experiments and analyses have proven the superiority of the proposed method.

## Limitations

Our method has three major limitations. First, the auxiliary data corpus with label information might be rare. Recall that the corpus we used in this paper is the training set of different benchmarks. However, large-scale labeled data as the auxiliary data source might be infeasible in practice, hence it may limit the model deployment in real-world scenarios. Second, our method is trained and evaluated on English datasets. Additional data processing as well as annotation is necessary for other linguistic settings. Third, external unlabeled data with the same domain as the ASTE datasets are needed for the pre-training of the retriever. In our experiment, we choose two external datasets in the restaurant and electronics domains. If our method is applied to other fields, we need to find additional external data in the corresponding domain for pre-training.

## Acknowledgements

The research work is supported by National Key RD Plan No. 2022YFC3303303, the National Natural Science Foundation of China under Grant (No.61976204), the Project of Youth Innovation Promotion Association CAS, Beijing Nova Program Z201100006820062. We would like to thank the anonymous reviewers for their insightful comments.

## References

Deng Cai, Yan Wang, Lemao Liu, and Shuming Shi. 2022. Recent advances in retrieval-augmented text generation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3417–3419.

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

Peng Chen, Shaowei Chen, and Jie Liu. 2020. Hierarchical sequence labeling model for aspect sentiment triplet extraction. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 654–666. Springer.

Shaowei Chen, Yu Wang, Jie Liu, and Yuelin Wang. 2021a. Bidirectional machine reading comprehension for aspect sentiment triplet extraction. In *Proceedings Of The AAAI Conference On Artificial Intelligence*, volume 35, pages 12666–12674.

Zhexue Chen, Hong Huang, Bang Liu, Xuanhua Shi, and Hai Jin. 2021b. Semantic and syntactic enhanced aspect sentiment triplet extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1474–1483.

Hongliang Dai and Yangqiu Song. 2019. Neural aspect and opinion term extraction with mined rules as weak supervision. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5268–5277.

Angela Fan, Claire Gardent, Chloe Braud, and Antoine Bordes. 2020. Augmenting transformers with

knn-based composite memory for dialogue. *arXiv preprint arXiv:2004.12744*.

Tianhao Gao, Jun Fang, Hanyu Liu, Zhiyuan Liu, Chao Liu, Pengzhang Liu, Yongjun Bao, and Weipeng Yan. 2022. LEGO-ABSA: A prompt-based task assemblable unified generative framework for multi-task aspect-based sentiment analysis. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 7002–7012, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Kelvin Guu, Panupong Pasupat, Evan Liu, and Percy Liang. 2017. From language to programs: Bridging reinforcement learning and maximum marginal likelihood. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1051–1062, Vancouver, Canada. Association for Computational Linguistics.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018. Exploiting document knowledge for aspect-level sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 579–585, Melbourne, Australia. Association for Computational Linguistics.

Hai Huan, Zichen He, Yaqin Xie, and Zelin Guo. 2022. A multi-task dual-encoder framework for aspect sentiment triplet extraction. *IEEE Access*.

Lianzhe Huang, Peiyi Wang, Sujian Li, Tianyu Liu, Xiaodong Zhang, Zhicong Cheng, Dawei Yin, and Houfeng Wang. 2021. First target and opinion then polarity: Enhancing target-opinion correlation for aspect sentiment triplet extraction. *arXiv preprint arXiv:2102.08549*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A survey on retrieval-augmented text generation. *arXiv preprint arXiv:2202.01110*.

Xiaoya Li, Yuxian Meng, Mingxin Zhou, Qinghong Han, Fei Wu, and Jiwei Li. 2020. Sac: Accelerating and structuring self-attention via sparse adaptive connection. *arXiv preprint arXiv:2003.09833*.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019. A unified model for opinion target extraction and target sentiment prediction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6714–6721.

Shu Liu, Kaiwen Li, and Zuhe Li. 2022. A robustly optimized bmrc for aspect sentiment triplet extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 272–278.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. 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.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. 2015. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52.

Yuxian Meng, Shi Zong, Xiaoya Li, Xiaofei Sun, Tianwei Zhang, Fei Wu, and Jiwei Li. 2021. Gnn-lm: Language modeling based on global contexts via gnn. In *International Conference on Learning Representations*.

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

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *International workshop on semantic evaluation*, pages 19–30.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 486–495.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Wei Shang, Chong Feng, Tianfu Zhang, and Da Xu. 2021. Guiding neural machine translation with retrieved translation template. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7. IEEE.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Learning semantic representations of users and products for document level sentiment classification. In *Proceedings of the 53rd annual meeting of the Association for Computational Linguistics and the 7th international joint conference on natural language processing (volume 1: long papers)*, pages 1014–1023.

David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2021. Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog. *arXiv preprint arXiv:2102.04643*.

Zhaopeng Tu, Yang Liu, Shuming Shi, and Tong Zhang. 2018. Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics*, 6:407–420.

Shuohang Wang, Yichong Xu, Yuwei Fang, Yang Liu, Siqi Sun, Ruochen Xu, Chenguang Zhu, and Michael Zeng. 2022. Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3170–3179.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Zhen Wu, Chengcan Ying, Fei Zhao, Zhifang Fan, Xinyu Dai, and Rui Xia. 2020. Grid tagging scheme for aspect-oriented fine-grained opinion extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2576–2585, Online. Association for Computational Linguistics.

Jitao Xu, Josep M Crego, and Jean Senellart. 2020a. Boosting neural machine translation with similar translations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1580–1590.

Lu Xu, Yew Ken Chia, and Lidong Bing. 2021. Learning span-level interactions for aspect sentiment triplet extraction. 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 4755–4766.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020b. Position-aware tagging for aspect sentiment triplet extraction. *arXiv preprint arXiv:2010.02609*.

Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2416–2429.

Samson Yu, Bai Jian, Tapas Nayak, Navonil Majumder, and Soujanya Poria. 2021. Aspect sentiment triplet extraction using reinforcement learning. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 3603–3607.

Chen Zhang, Qiuchi Li, Dawei Song, and Benyou Wang. 2020. A multi-task learning framework for opinion triplet extraction. In *EMNLP (Findings)*.

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.

Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, and Wai Lam. 2021b. 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.

Yichun Zhao, Kui Meng, Gongshen Liu, Jintao Du, and Huijia Zhu. 2022. A multi-task dual-tree network for aspect sentiment triplet extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 7065–7074, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

## A Statistics of Dataset

In order to quantitatively compare our method to prior work, we conduct our experiments on two widely used ASTE datasets $\mathcal{D}_a$ and $\mathcal{D}_b$, which are released by Peng et al. (2020) and Xu et al. (2020b) and originate from Semeval2014 (Pontiki et al., 2014), Semeval2015 (Pontiki et al., 2015), and Semeval2016 (Pontiki et al., 2016). The statistics are shown in Table 5. Each of them consists of three datasets: Restaurant14, Restaurant15, Restaurant16, and Laptop14. The first three datasets are from the restaurant domain, in which each sentence describes a customer's evaluation of restaurant service, environment, food, etc. The Laptop14

| Dataset | | Res14 | Lap14 | Res15 | Res16 |
|---|---|---|---|---|---|
| | | *s/pos/neu/neg* | *s/pos/neu/neg* | *s/pos/neu/neg* | *s/pos/neu/neg* |
| $\mathcal{D}_a$ | Train | 1300/1575/143/427 | 593/703/25/195 | 842/933/49/307 | 920/664/117/484 |
| | Dev | 323/377/32/115 | 148/179/9/50 | 210 /225/10/81 | 228/207/16/114 |
| | Test | 496/675/45/142 | 318/291/25/139 | 320 /362/27/76 | 339/335/50/105 |
| $\mathcal{D}_b$ | Train | 1266/1692/166/480 | 906/817/126/517 | 605/783/25/205 | 857/1015/50/329 |
| | Dev | 310/404/54/119 | 219/169/36/141 | 148/185/11/53 | 210/252/11/76 |
| | Test | 492/773/66/155 | 328/364/63/116 | 322/317/25/143 | 326/407/29/78 |

Table 5: Statistics of $\mathcal{D}_a$ and $\mathcal{D}_b$. *s*, *pos*/*neu*/*neg* denote the numbers of the sentence, positive/neutral/negative triplets, respectively.

datasets in $\mathcal{D}_a$ and $\mathcal{D}_b$ contain customer evaluations related to electronic products. All the datasets contain initialized training set, development set, and test set. Since existing popular methods are implemented on either $\mathcal{D}_a$ or $\mathcal{D}_b$, evaluating our model on two datasets can compare it with existing methods as comprehensively as possible and get more reliable experimental conclusions.

During the pre-training for retrieval, we adopt two document-level datasets named Yelp (Tang et al., 2015) and Amazon (McAuley et al., 2015) as external data, which are processed and released by He et al. (2018). For each dataset, we sort all the data according to the length of the document and select the shortest $10,000$ pieces of data for pre-training of our retriever. Specifically, Yelp is from the restaurant domain, which is used to generate pseudo-labeled data for Restaurant14, Restaurant15, and Restaurant16. Amazon is from the electronics domain, which is used to generate external data for Laptop14.

## B Experimental Settings

We adopt the BERT-base model from *huggingface* Transformer library [3] for all experiments. We pre-train the relevance scores for 10 epochs with batch size 8 and learning rate $1e-5$. We jointly train the full model for 30 epochs with batch size 1, and a learning rate of $1e-5$. We also use an early stopping and a linear warmup for $10\%$ of the training step during the joint learning. We adopt the Adam optimizer and accumulate gradients for each batch. We set the dropout rate, the maximum span width, the number of candidate aspect and opinion terms, the number $L$ of retrieved triplet, the loss coefficients $\alpha$ to $0.5$, $8$, half of the sentence length, $5$, and $5$.

In each iteration, we first extract candidate aspect terms and opinion terms and pair them into candidate aspect-opinion pairs. Then we retrieve relevant triplets for each pair and help them predict if the pair is valid and further determine their sentiment polarities. Finally, we update the parameters by gradient descent. The code is implemented with PyTorch 1.9.0 and transformers 4.1.1 and launched on an Ubuntu server with an NVidia Tesla V100 (32G). In addition, we will test our model with Mindspore, which is a new deep-learning framework[4].

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

[4] https://www.mindspore.cn/

## ACL 2023 Responsible NLP Checklist

### A For every submission:

☑ A1. Did you describe the limitations of your work?
*Section named limitations at the end of the paper.*

☑ A2. Did you discuss any potential risks of your work?
*Section named limitations at the end of the paper.*

☑ A3. Do the abstract and introduction summarize the paper's main claims?
*Section1*

☒ A4. Have you used AI writing assistants when working on this paper?
*Left blank.*

### B ☑ Did you use or create scientific artifacts?

*section 6*

☑ B1. Did you cite the creators of artifacts you used?
*section 6, appendix A, appendix B*

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
*section 6.1 and appendix A*

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
*section 6.1 and appendix A*

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
*Not applicable. Left blank.*

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
*appendix A*

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
*appendix A*

### C ☑ Did you run computational experiments?

*Section 6, Appendix B*

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
*Not applicable. Left blank.*

*The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.*

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
*Appendix B*

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
*Section 6 and Appendix B*

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
*Section 6 and Appendix B*

**D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?**

*Left blank.*

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
*No response.*

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
*No response.*

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
*No response.*

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
*No response.*

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
*No response.*