Boundary-Driven Table-Filling for Aspect Sentiment Triplet Extraction

Yice Zhang[1,2*], **Yifan Yang**[1,2*], **Yihui Li**[1,2], **Bin Liang**[1,2†], **Shiwei Chen**[1,3],
Yixue Dang[4], **Ming Yang**[5], **and Ruifeng Xu**[1,2,3†]
[1] Harbin Insitute of Technology, Shenzhen, China
[2] Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies
[3] Peng Cheng Laboratory, Shenzhen, China
[4] Joint Lab of HITSZ and China Merchants Securities
[5] SIAT, Chinese Academy of Sciences, Shenzhen, China
{zhangyc_hit,evanyfyang,liyihui0413}@163.com
bin.liang@stu.hit.edu.cn, chenshw@pcl.ac.cn
dangyixue@cmschina.com.cn, min.yang@siat.ac.cn
xuruifeng@hit.edu.cn

Abstract

Aspect Sentiment Triplet Extraction (ASTE) aims to extract the aspect terms along with the corresponding opinion terms and the expressed sentiments in the review, which is an important task in sentiment analysis. Previous research efforts generally address the ASTE task in an end-to-end fashion through the table-filling formalization, in which the triplets are represented by a two-dimensional (2D) table of word-pair relations. Under this formalization, a term-level relation is decomposed into multiple independent word-level relations, which leads to relation inconsistency and boundary insensitivity in the face of multi-word aspect terms and opinion terms. To overcome these issues, we propose Boundary-Driven Table-Filling (BDTF), which represents each triplet as a relation region in the 2D table and transforms the ASTE task into detection and classification of relation regions. We also notice that the quality of the table representation greatly affects the performance of BDTF. Therefore, we develop an effective relation representation learning approach to learn the table representation, which can fully exploit both word-to-word interactions and relation-to-relation interactions. Experiments on several public benchmarks show that the proposed approach achieves state-of-the-art performances[1].

1 Introduction

As a fine-grained task, Aspect-Based Sentiment Analysis (ABSA) focuses on the attitudes expressed on specific aspect terms (Pontiki et al., 2014). Opinion terms refer to words or phrases expressing subjective sentiments. Intuitively, opinion terms are important clues when determining the sentiment polarity and could provide a more detailed sentiment description for the aspect terms. Previous work has generally concentrated on extracting aspect terms and opinion terms and classifying the sentiment expressed on the aspect term without explicitly considering the relations between aspect terms and opinion terms (He et al., 2019; Li et al., 2019; Chen and Qian, 2020). Therefore, Peng et al. (2020) propose the Aspect Sentiment Triplet Extraction (ASTE) task, which is exemplified in Figure 1. In the ASTE task, a triplet consists of an aspect term, the corresponding opinion term, and the expressed sentiment.

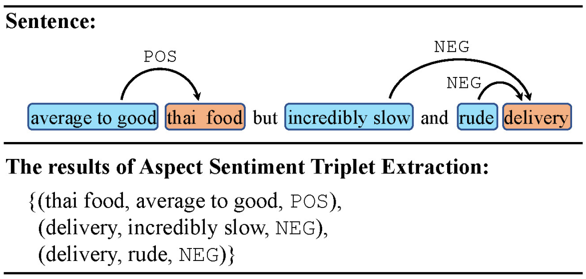

Figure 1: An example of the ASTE task. The aspect terms and the opinion terms are marked with orange and blue, respectively.

Peng et al. (2020) adopt the pipeline approach to address the ASTE task. They first decompose the ASTE task into several subtasks and then learn models separately for each subtask. A more advanced alternative is learning a joint model to exploit the interactions between different subtasks (Xu et al., 2020; Wu et al., 2020; Chen et al., 2021a; Mao et al., 2021; Xu et al., 2021; Jing et al., 2021). Among these works, Wu et al. (2020) and Jing

* The first two authors contribute equally to this work.
† Corresponding Authors
[1]We make our code publicly available at https://github.com/HITSZ-HLT/BDTF-ABSA.

et al. (2021) tackle the ASTE task through a table-filling approach, where the triplets are represented by a two-dimensional (2D) table of word-pair relations. In this approach, aspect terms and opinion terms are extracted through the diagonal elements of the table, and sentiments are treated as relation tags that are represented by the non-diagonal elements of the table. This formalization enables joint learning of different subtasks in ASTE, achieving superior performance over the pipeline approach.

However, the previous table formalization suffers from *relation inconsistency* and *boundary insensitivity* when dealing with multi-word aspect terms and opinion terms. It decomposes the relation between an aspect term and an opinion term into the relations between the corresponding aspect words and opinion words. In other words, a term-level relation is represented by several word-level relation tags. The relation tags in the table are assigned independently, which leads to potential inconsistencies in the predictions of the word-level relations. In addition, when there are minor boundary errors in the aspect term or opinion term, the voting result for the term-level relation may stay unchanged, encouraging the model to produce wrong predictions. Xu et al. (2021) try to solve this problem through a span-based method, but their method discards fine-grained word-level information, which is the advantage of the table-filling approach.

In this paper, we propose a Boundary-Driven Table-Filling (BDTF) approach for ASTE to overcome the above issues. Instead of decomposing ASTE into term extraction and relation classification, it extracts triplets by directly detecting and classifying the relation regions in a 2D table. Specifically, we first detect all possible relation regions in the table through a region detection layer, which is enabled by predefined boundary tags. Then we employ a region classifier to determine the sentiment label for each relation region. Classification over the entire relation region ensures relation consistency, and those relation regions with boundary errors can be removed by being classified as invalid.

To support the proposed BDTF, we also develop an effective relation representation learning approach to learn the table representation. We first learn the word-level contextualized representations of the input review through a pre-trained language model. Then we adopt a tensor-based operation to

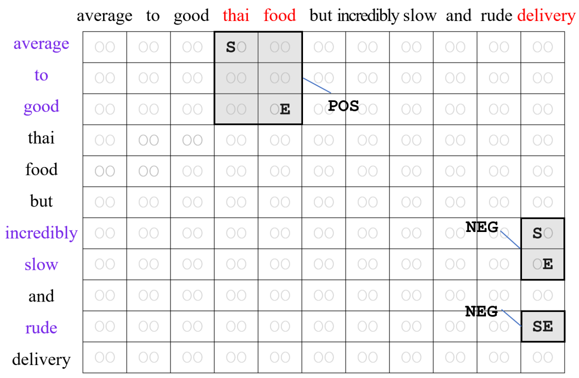

Figure 2: An example with BDTF for the ASTE task.

construct the relation-level representations to fully exploit the word-to-word interactions. Finally, we model relation-to-relation interactions through a multi-layer convolution-based encoder to enhance the relation-level representations. The relation representations of each two words in the review together form a 2D relation matrix, which serves as the table representation for BDTF.

Our contributions can be summarized as follows:

- We propose a Boundary-Driven Table-Filling (BDTF) approach for the ASTE task. It extracts aspect sentiment triplets from reviews by directly detecting and classifying the relation regions, overcoming relation inconsistency and boundary insensitivity of the previous methods.
- We develop an effective relation representation learning approach to learn the table representation, which fully exploits both word-to-word interactions and relation-to-relation interactions.
- Extensive experiments are conducted on several aspect-opinion tasks including ASTE, and the results demonstrate that our approach significantly outperforms the state-of-the-art methods.

2 Our Approach

2.1 Task Formalization

Given a sentence $X = [x_1, x_2, \cdots, x_n]$ of length n, the goal of the ASTE task is to extract a set of aspect sentiment triplets. A triplet is defined as $(aspect, opinion, sentiment)$ where $sentiment \in \{\texttt{POS}, \texttt{NEU}, \texttt{NEG}\}$. As shown in Figure 2, we represent a triplet as a relation region in the 2D table. Its boundary is used to indicate

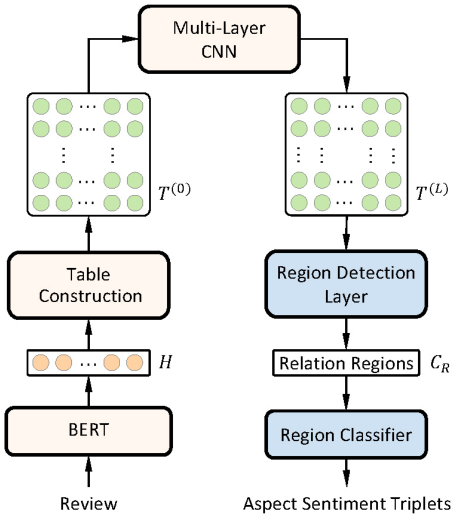

Figure 3: The proposed BDTF-ASTE approach.

the position of the aspect term and opinion term, and its type is used to indicate the sentiment. Relation regions are located by two boundary tags. Specifically, `S` denotes the upper left corner, and `E` denotes the lower right corner.

2.2 Model Overview

We briefly present the proposed approach in Figure 3. For an input review, we first learn the word-level contextualized representations via BERT and then learn a 2D table representation by constructing and encoding relation-level representations. Each element of this table representation is a vector representing a word-pair relation. Next, we detect all candidate relation regions in the table and predict the type of each relation region by a region classifier. Finally, we decode the aspect sentiment triplets based on the boundaries and types of the relation regions.

2.3 Representation Learning

2.3.1 Word-Level Representation Learning

We first employ a pre-trained language model such as BERT (Devlin et al., 2019) as the language encoder to obtain the word-level contextualized representations of the input sentence. This process can be formulated as follows:

$$\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_n = \mathrm{BERT}(x_1, x_2, \cdots, x_n). \quad (1)$$

2.3.2 Relation-Level Representation Construction

Most of the existing work constructs the relation representation between two words by feature concatenation (Wu et al., 2020; Jing et al., 2021; Xu et al., 2021). However, this method *underutilizes* word-to-word interactions because a relation is not a simple composition of two words. Inspired by Socher et al. (2013), we adopt a tensor-based operation to construct the relation-level representation. Given two words $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^d$, the tensor-based operation is defined as:

$$\mathcal{T}(\boldsymbol{h}_i, \boldsymbol{h}_j; V) = \boldsymbol{h}_i^\top V^{[1:t]} \boldsymbol{h}_j, \quad (2)$$

where $V \in \mathbb{R}^{d \times d \times t}$ is a tensor parameter. Specifically, for each slice $V^{[k]} \in \mathbb{R}^{d \times d}$, the tensor-based operation computes the inner product of two word representations in a certain vector space:

$$\mathcal{T}(\boldsymbol{h}_i, \boldsymbol{h}_j; V)_k = \boldsymbol{h}_i^\top V^{[k]} \boldsymbol{h}_j. \quad (3)$$

By introducing multiple vector spaces, the tensor-based operation can fully model word-to-word interactions.

In addition, we also exploit the context between two words, as the context often contain important indicators of the expressed relation (Eberts and Ulges, 2020). We obtain the context representation $\boldsymbol{c}_{ij} \in \mathbb{R}^d$ by the max-pooling operation (let $i \leq j$):

$$\boldsymbol{c}_{ij} = \mathrm{pooling}(\boldsymbol{h}_i, \boldsymbol{h}_{i+1}, \cdots, \boldsymbol{h}_j). \quad (4)$$

Finally, we construct the relation-level representation $\boldsymbol{r}_{ij}^{(0)} \in \mathbb{R}^d$ through a nonlinear projection:

$$\boldsymbol{r}_{ij}^{(0)} = f(\mathrm{Linear}([\boldsymbol{h}_i; \boldsymbol{h}_j; \boldsymbol{c}_{ij}; \boldsymbol{t}_{ij}])), \quad (5)$$
$$\boldsymbol{t}_{ij} = \mathcal{T}(\boldsymbol{h}_i, \boldsymbol{h}_j; V) \in \mathbb{R}^t. \quad (6)$$

where $f(\cdot)$ is an activation function, and we empirically choose `gelu` (Hendrycks and Gimpel, 2016) in this paper.

2.3.3 Relation-Level Representation Encoding

The relation representations of each two words in the sentence together form a 2D relation matrix, *i.e.*, the table representation. There are some potential dependencies among the elements of this table representation. For example, an element with boundary tag `S` always has an element with boundary tag `E` at its lower right, and elements in the same relation region have the same sentiment label.

To model these dependencies, we utilize the ResNet-style CNNs (He et al., 2016) to encode this relation matrix. Specifically, given the input table representation $T^{(l-1)} \in \mathbb{R}^{n \times n \times d}$, the l-th layer CNN produces a table representation $T^{(l)}$ of the same size by:

$$T' = \mathrm{ReLU}(\mathrm{LN}(\mathrm{Conv}_{1\times1}(T^{(l-1)}))), \tag{7}$$
$$T'' = \mathrm{ReLU}(\mathrm{LN}(\mathrm{Conv}_{3\times3}(T'))), \tag{8}$$
$$T''' = \mathrm{ReLU}(\mathrm{LN}(\mathrm{Conv}_{1\times1}(T''))), \tag{9}$$
$$T^{(l)} = T''' + T^{(l-1)}, \tag{10}$$

where LN denotes Layer Normalization (Ba et al., 2016).

2.4 Extraction Module

2.4.1 Region Detection Layer

For each element $\boldsymbol{r}_{ij}^{(L)}$ in the table representation $T^{(L)}$, the region detection layer utilizes two classifiers to calculate the probability of its boundary tag being `S` and `E`:

$$P_{ij}^{\mathtt{S}} = \mathrm{sigmoid}\left(\mathrm{Linear}\left(\boldsymbol{r}_{ij}^{(L)}\right)\right), \tag{11}$$
$$P_{ij}^{\mathtt{E}} = \mathrm{sigmoid}\left(\mathrm{Linear}\left(\boldsymbol{r}_{ij}^{(L)}\right)\right). \tag{12}$$

Instead of decoding `S` and `E` based on a certain threshold, we prune `S` and `E` through a top-k strategy and then combine them to get the candidate relation regions. Pruning allows the model to avoid potential exposure bias[2] (Schmidt, 2019). Specifically, we select the top-k candidates by $P_{ij}^{\mathtt{S}}$ and $P_{ij}^{\mathtt{E}}$, respectively. Then these valid `S`-`E` pairs of the selected candidates form the region candidate pool $C_R = \{\cdots, [\mathtt{S}(a,b), \mathtt{E}(c,d)], \cdots\}$, where an `S`-`E` pair is only valid if `E` is not on top or left of `S`, (*i.e.*, $a \le c$ and $b \le d$). The value of k is related to the length of the input sentence:

$$k = \max(n \cdot z, k_{min}), \tag{13}$$

where z and k_{min} are two hyper-parameters.

2.4.2 Region Classifier

Given a candidate relation region determined by $\mathtt{S}(a,b)$ and $\mathtt{E}(c,d)$, we concatenate the `S` representation, the `E` representation, and the max-pooling result of the relation matrix over this region as its feature representation $\boldsymbol{r}_{abcd} \in \mathbb{R}^{3d}$:

$$\boldsymbol{p}_{abcd}^{(L)} = \mathrm{pooling}\begin{pmatrix} \boldsymbol{r}_{ab}^{(L)} & \cdots & \boldsymbol{r}_{ad}^{(L)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{r}_{cb}^{(L)} & \cdots & \boldsymbol{r}_{cd}^{(L)} \end{pmatrix}, \tag{14}$$
$$\boldsymbol{r}_{abcd} = \left[\boldsymbol{r}_{ab}^{(L)}; \boldsymbol{r}_{cd}^{(L)}; \boldsymbol{p}_{abcd}^{(L)}\right]. \tag{15}$$

Then we use a classifier to predict its type $y_{\mathcal{T}} \in \{\mathtt{POS}, \mathtt{NEU}, \mathtt{NEG}, \mathtt{Invalid}\}$:

$$P_{abcd}(y_{\mathcal{T}}) = \mathrm{softmax}(\mathrm{Linear}(\boldsymbol{r}_{abcd})). \tag{16}$$

2.5 Training and Decoding

During training, we utilize the cross-entropy function to calculate the loss of boundary detection and region classification. Specifically, given the ground truth boundary label $y_{ij}^{\mathtt{S}}, y_{ij}^{\mathtt{E}} \in \{0,1\}$, the loss of region detection is calculated by:

$$\mathcal{L}_{\mathcal{B}} = \mathcal{L}_{\mathtt{S}} + \mathcal{L}_{\mathtt{E}}, \tag{17}$$
$$\mathcal{L}_{\mathtt{S}} = -\sum_{i,j\in[1,n]} y_{ij}^{\mathtt{S}} \log P_{ij}^{\mathtt{S}} + (1-y_{ij}^{\mathtt{S}}) \log(1-P_{ij}^{\mathtt{S}}),$$
$$\mathcal{L}_{\mathtt{E}} = -\sum_{i,j\in[1,n]} y_{ij}^{\mathtt{E}} \log P_{ij}^{\mathtt{E}} + (1-y_{ij}^{\mathtt{E}}) \log(1-P_{ij}^{\mathtt{E}}).$$

Given the ground truth region type $y_{\mathcal{T}}^*$, the loss of region classification is calculated by:

$$\mathcal{L}_{\mathcal{T}} = -\sum_{abcd\in C_R} \log P_{abcd}(y_{\mathcal{T}}^*). \tag{18}$$

The overall optimization objective is to minimize the summation of these two losses $\mathcal{L}_{\mathcal{B}} + \mathcal{L}_{\mathcal{T}}$.

During decoding, we first obtain the region candidate pool through the region detection layer and then utilize the region classifier to predict the type of each candidate region. We drop those relation regions whose predicted types are `Invalid` and generate the aspect sentiment triplets from the remaining relation regions. Suppose a relation region determined by $\mathtt{S}(a,b)$ and $\mathtt{E}(c,d)$ is predicted to be of type `POS`, then its corresponding triplet is:

$$aspect = [x_a, \cdots, x_c], \tag{19}$$
$$opinion = [x_b, \cdots, x_d], \tag{20}$$
$$sentiment = \mathtt{POS}. \tag{21}$$

3 Experiments

3.1 Datasets

We evaluate our approach on four public datasets from SemEval 2014[3] (Pontiki et al., 2014), SemEval 2015[4] (Pontiki et al., 2015), and SemEval 2016[5] (Pontiki et al., 2016). For these datasets, Fan et al. (2019) annotate opinion terms for each

[2] *Exposure bias* here means that for subsequent region classification, the true `S` and `E` are used during training, and the predicted `S` and `E` are used during inference, which will introduce discrepancies in the distribution.

[3] http://alt.qcri.org/semeval2014/task4/
[4] http://alt.qcri.org/semeval2015/task12/
[5] http://alt.qcri.org/semeval2016/task5/

Dataset	Split	#Sent	#A	#O	#T
Rest 14	Train	1266	2051	2061	2338
	Dev	310	500	497	577
	Test	492	848	844	994
Lap 14	Train	906	1280	1254	1460
	Dev	219	295	302	346
	Test	328	463	466	543
Rest 15	Train	605	862	935	1013
	Dev	148	213	236	249
	Test	322	432	460	485
Rest 16	Train	857	1198	1300	1394
	Dev	210	296	319	339
	Test	326	452	474	514

Table 1: Statistics of ASET-Data-v2 (Xu et al., 2020). #Sent, #A, #O, and #T represent the number of sentences, aspect terms, opinion terms, and triplets, respectively.

aspect term. Based on this, Peng et al. (2020) release ASTE-Data-v1. Later, it is found that not all triplets are annotated (Xu et al., 2020; Wu et al., 2020). Therefore, Xu et al. (2020) refine these datasets and release ASTE-Data-v2. We compare our approach with previous methods and perform the ablation study on ASTE-Data-v2. Its data statistics is detailed in Table 1.

We also evaluate our approach on ASTE-Data-v1. Additionally, we run our approach on the Aspect-Opinion Pair Extraction (AOPE) task, which is a similar task to ASTE. We compare our approach with previous methods on two AOPE-Data (Fan et al., 2019; Chen et al., 2020). These results are presented in Appendix A.2 and A.3.

3.2 Implementation Details

We adopt `BERT-base-uncased` (Devlin et al., 2019) as the default language encoder, which consists of 12 Transformer blocks with a hidden size of 768. The number of layers of the table encoder is set to 2. We set $t = 64$, $z = 0.3$, and $k_{min} = 5$. We train the model for 10 epochs and select the best model according to the performance on the development set. We run our approach five times with different random seeds and report the average results (F_1-score).

3.3 Baselines

We categorize the baselines into four groups: table-filling methods, span-based methods, generative methods, and other methods.

Table-Filling methods represent aspect terms and opinion terms along with their sentiment relations as word-pair relations. Wu et al. (2020) propose Grid Tagging Scheme (GTS) and design an inference strategy to exploit mutual indication between different opinion factors. Zhang et al. (2020) propose a multi-task learning framework (OTE-MTL) to jointly extract terms and parse sentiment dependencies. Dual-Encoder (Jing et al., 2021) and TGA +SFI (Wang et al., 2021a) learn the sequence representation and the table representation via *table-sequence* encoders. Chen et al. (2022) propose an Enhanced Multi-Channel Graph Convolutional Network model (EMC-GCN) to utilize linguistic features.

Span-Based methods perform term extraction and relation classification through the shared span representations. Span-ASTE (Xu et al., 2021) introduces a dual-channel span pruning strategy to ease the high computational cost caused by span enumeration. SSJE (Li et al., 2022a) utilizes a Graph Convolutional Network (GCN) on the syntactic dependency tree of the sentence to enhance the span representations.

Generative methods generally convert the ASTE task into the index generation problem, including PASTE (Mukherjee et al., 2021), Span-BART (Yan et al., 2021), GAS (Zhang et al., 2021c), Paraphrase (Zhang et al., 2021b), and UIE (Lu et al., 2022).

Others Peng et al. (2020) divide the triplet extraction into two stages and then learn two separate models. Xu et al. (2020) present a position-aware tagging scheme for ASTE and accordingly propose a joint approach, JET. Chen et al. (2021a) transform the ASTE task into the multi-turn machine reading comprehension (MRC) task and address it through a bidirectional MRC (BMRC) framework. Yu Bai Jian et al. (2021) present ASTE-RL by treating the aspect and opinion terms as arguments of the expressed sentiment in a hierarchical reinforcement learning (RL) framework.

3.4 Main Results

Table 2 lists the comparison results on the ASTE task. According to these results, our approach consistently attains the best performance, demonstrating its effectiveness. More specifically, we have the following observations. (1) Our approach achieves F_1-score improvements of 2.50%, 2.36%, 2.85%, and 2.01% on the four datasets compared with the previous best baseline model without in-

Model	Rest 14			Lap 14			Rest 15			Rest 16		
	$P.$	$R.$	F_1	$P.$	$R.$	F_1	$P.$	$R.$	F_1	$P.$	$R.$	F_1
Two-stage†(Peng et al., 2020)	43.24	63.66	51.46	37.38	50.38	42.87	48.07	57.51	52.32	46.96	64.24	54.21
JET$^{o}_{M=6}$(BERT)(Xu et al., 2020)	70.56	55.94	62.40	55.39	47.33	51.04	64.45	51.96	57.53	70.42	58.37	63.83
BMRC*(Chen et al., 2021a)	72.17	65.43	68.64	65.91	52.15	58.18	62.48	55.55	58.79	69.87	65.68	67.35
ASTE-RL(Yu Bai Jian et al., 2021)	70.60	68.65	69.61	64.80	54.99	59.50	65.45	60.29	62.72	67.21	69.69	68.42
PASTE$_{AF}$(Mukherjee et al., 2021)	66.70	66.50	66.60	61.20	53.60	57.10	61.70	60.80	61.30	66.10	69.80	67.90
Table-Filling Approaches												
OTE-MTL$^{\natural}$(Zhang et al., 2020)	62.70	57.10	59.71	49.62	41.07	44.78	55.63	42.51	47.94	60.95	53.35	56.82
GTS-BERT$^{\natural}$(Wu et al., 2020)	67.76	67.29	67.50	57.82	51.32	54.36	62.59	57.94	60.15	66.08	66.91	67.93
Double-Encoder(Jing et al., 2021)	67.95	71.23	69.55	62.12	56.38	59.11	58.55	60.00	59.27	70.65	70.23	70.44
TGA+SFI(Wang et al., 2021a)	71.75	70.52	71.13	65.25	53.79	58.98	62.77	59.79	61.25	68.20	69.26	68.73
EMC-GCN(Chen et al., 2022)	71.21	72.39	71.78	61.70	56.26	58.81	61.54	62.47	61.93	65.62	71.30	68.33
Span-Based Approaches												
Span-ASTE(Xu et al., 2021)	72.89	70.89	71.85	63.44	55.84	59.38	62.18	64.45	63.27	69.45	71.17	70.26
SSJE(Li et al., 2022a)	73.12	71.43	72.26	67.43	54.71	60.41	63.94	66.17	65.05	70.82	72.00	71.38
SSJE w/o GCN(Li et al., 2022a)	73.45	69.32	71.33	62.70	56.56	59.48	61.43	63.71	62.55	69.01	70.62	69.81
BDTF (Ours)	**75.53**	**73.24**	**74.35**	**68.94**	55.97	**61.74**	**68.76**	63.71	**66.12**	**71.44**	**73.13**	**72.27**

Table 2: Results on ASTE-Data-v2 (Xu et al., 2020) (%). The results with † are retrieved from Xu et al. (2020). The results with ♮ and $*$ are reproduced by Xu et al. (2021) and Yu Bai Jian et al. (2021). All the above results except Peng et al. (2020) and Zhang et al. (2020) are obtained with `BERT-base-uncased` as the language encoder.

Model	PreTrained Model	#Params	Rest 14	Lap 14	Rest 15	Rest 16	AVG
Span-BART(Yan et al., 2021)	`BART-base`	139M	65.25	58.69	59.26	67.62	61.71
GAS(Zhang et al., 2021c)	`T5-base`	223M	72.16	60.78	62.10	70.10	66.29
Paraphrase(Zhang et al., 2021b)	`T5-base`	223M	72.03	61.13	62.56	71.70	66.86
Double-Encoder(Jing et al., 2021)	`BERT-base-uncased`	109M	69.55	59.11	59.27	70.44	64.59
	`ALBERT-xxlarge-v1`	223M	74.82	63.30	67.67	72.01	69.45
SSI+SEL(Lu et al., 2022)	`T5-v1.1-base`	223M	71.27	58.69	59.60	70.24	64.95
	`UIE-base`	223M	72.55	62.94	64.41	72.86	68.19
	`UIE-large`	750M	74.52	63.88	67.15	75.07	70.16
BDTF (Ours)	`BART-base.encoder`	82M	74.79	62.46	64.64	70.45	68.09
BDTF (Ours)	`BERT-base-uncased`	109M	74.35	61.74	66.12	72.27	68.62
BDTF (Ours)	`RoBERTa-base`	125M	75.20	64.08	67.64	74.15	70.27
BDTF (Ours)	`DeBERTa-v3-base`	184M	**75.48**	**66.71**	**68.22**	**75.36**	**71.44**

Table 3: Comparison results with different pre-trained models (F_1-score, %).

troducing the syntactic dependency tree (Xu et al., 2021). (2) Although Li et al. (2022a) introduce the additional word dependency information in representation learning through Graph Convolutional Network (GCN), our approach still outperforms their approach. (3) Compared with the previous Table-Filling approaches (Zhang et al., 2020; Wu et al., 2020; Jing et al., 2021; Wang et al., 2021a; Chen et al., 2022), our approach shows substantial improvements in F_1-score. These improvements in F_1-score are more attributable to the improvements in *Precision*. For example, compared with Jing et al. (2021), our approach obtains F_1-score gains of 4.80%, 2.63%, and 6.80% for Rest14, Lap14, and Rest15, while the corresponding *Precision* gains reach 7.58%, 6.82%, and 10.21%. This suggests that our approach produces fewer wrong predictions due to its boundary sensitivity.

Many works (especially generative methods) adopt Pre-trained Language Models (PLMs) other than `BERT-base-uncased` as the backbone. Therefore, we implement our approach with different PLMs and present the results in Table 3. These results show that PLM has a significant impact on model performance. For our approach, replacing BERT with RoBERTa (Liu et al., 2019) can directly bring about 1.5% F_1-score improvement. Besides, we observe that, even with the

Model	Rest 14				Lap 14				Rest 15				Rest 16			
	$B.$	R_S	R_M	$O.$	$B.$	R_S	R_M	$O.$	$B.$	R_S	R_M	$O.$	$B.$	R_S	R_M	$O.$
TF$_{GTS}$	10.16	2.35	6.02	12.98	11.75	5.18	7.97	16.14	14.37	2.52	6.02	18.25	9.21	4.24	4.42	14.36
BDTF	6.77	2.19	3.33	11.34	6.03	5.36	6.70	12.95	10.50	3.06	4.16	12.69	5.34	3.36	4.35	13.04
Δ	-3.39	-0.16	-2.69	-1.64	-5.75	+0.18	-1.27	-3.19	-3.87	+0.54	-1.86	-5.56	-3.87	-0.88	-0.07	-1.32

Table 4: Analysis of the wrong predictions produced by different table-filling approaches. $B.$, R_S, R_M, and $O.$ denote the proportion of boundary error, single-word relation error, multi-word relation error, and other errors in all predictions, respectively.

Decoding	Rest 14	Lap 14	Rest 15	Rest 16	AVG-Δ
TF$_{GTS}$	<u>70.00</u>	<u>56.60</u>	59.58	<u>68.75</u>	-
TF$_{Double}$	69.04	54.98	<u>59.75</u>	67.61	-
BDTF(Ours)	**74.35**	**61.74**	**66.12**	**72.27**	**+4.85**

Table 5: Comparison of the table-filling approaches (F_1-score, %).

encoder of `BART-base`, our approach can outperform most methods using `base`-size PLMs. When using `DeBERTa-v3-base` (He et al., 2021) as the backbone, our approach consistently achieves the best performance, even surpassing the previous methods using the `large`-size PLMs. These results show the superiority of the proposed approach.

Additionally, we evaluate our approach on ASTE-Data-v1 (Peng et al., 2020) and two AOPE-Data (Fan et al., 2019; Chen et al., 2020). These results are detailed in the Appendix A.2 and A.3. Briefly, our approach also achieves the best performance on these datasets.

3.5 Ablation Studies

3.5.1 Comparison of Table-Filling Approaches

To verify the effectiveness of our proposed table-filling approach, we replace our extraction module with TF$_{GTS}$ (Wu et al., 2020) and TF$_{Double}$[6] (Jing et al., 2021) while keeping the rest of the model unchanged.

As shown in Table 5, our table-filling approach significantly outperforms the previous table-filling approaches on all four datasets. The average F_1-score improvement is 4.85%. Notice that there are a few overlapping cases between aspect terms and opinion terms in the ASTE-Data-v2. Both TF$_{GTS}$ and TF$_{Double}$ suffer from this issue[7], but our approach can successfully solve it. This is one reason why our approach outperforms them.

To verify boundary sensitivity of our approach, we perform a detailed analysis of the wrong predictions. We first categorize the wrong predictions into three types: *boundary errors*, *relation errors*, and other errors. When the aspect term or opinion term of an extracted triplet is boundary-misspecified, we categorize it as a boundary error. When the boundary of a triplet is completely correct, but the sentiment relation is incorrectly identified, we categorize it as a relation error. Relation errors are further divided into single-word and multi-word errors. We count the proportion of each type of error in the predictions and list the results in Table 4. It can be observed that, compared with the previous table-filling approaches, our approach significantly reduces the number of boundary errors in the predictions. The proportion of boundary errors on Lap14 is reduced by about half. This shows that our approach effectively filters those boundary-misspecified triplets. Furthermore, our approach also reduces the proportion of relation errors, especially on multi-word relations. This may be due to its relation consistency.

3.5.2 Relation Learning Analysis

We conduct an ablation experiment to explore the effects of the components of relation representation learning. As shown in Table 7, the model performs poorly when the relation representation is obtained only by feature concatenation. Adding context, tensor-based operation, or CNN results in significant performance improvements. Adding these three components together improves the average performance by 8.71%. This demonstrates the necessity of each component and the overall effectiveness of our relation representation learning

[6]TF$_{Double}$ utilizes two distinct encoders to learn sequence representation and table representation separately. In our implementation, we take the contextualized representation output by BERT as the sequence representation.

[7]Neither TF$_{GTS}$ nor TF$_{Double}$ account for overlapping cases in their original implementations, and thus some triplets are ignored when calculating metrics, which makes their published performance inflated. Our implementation fixes this issue.

Review	Ground-truth	$\textbf{TF}_{Double}$	**BDTF**(Ours)
The downstairs bar scene is very cool and chill.	{downstairs bar scene, cool, POS} {downstairs bar scene, chill, POS}	{downstairs, cool, POS} {downstairs, chill, POS} {bar scene, cool, POS} {bar scene, chill, POS}	{downstairs bar scene, cool, POS} {downstairs bar scene, chill, POS}
new hamburger with special sauce is ok - at least better than big mac!	{new hamburger with special sauce, ok, POS} {big mac, better than, NEG}	{hamburger, ok, POS} {big mac, better, POS}	$\emptyset$
The menu is interesting and quite reasonably priced.	{menu, interesting, POS} {menu, reasonably priced, POS} {priced, reasonably, POS}	{menu, interesting, POS} {menu, reasonably priced, POS}	{menu, interesting, POS} {menu, reasonably priced, POS} {priced, reasonably, POS}
However , I can refute that OSX is "FAST".	{OSX, FAST, NEG}	{OSX, FAST, POS}	{OSX, FAST, POS}
i love their chicken pasta cant remember the name but is sooo good.	{chicken pasta, love, POS}	{chicken pasta, love, POS} {chicken pasta, good, POS}	{chicken pasta, love, POS} {chicken pasta, good, POS}

Table 6: Case study.

Model	**Rest 14**	**Lap 14**	**Rest 15**	**Rest 16**	**AVG-Δ**
Concat	68.13	52.31	55.11	64.09	-
Concat + Context	72.96	59.25	64.96	72.12	+7.41
Concat + Tensor	71.63	58.78	63.77	70.17	+6.18
Concat + CNN	71.91	55.01	64.81	69.39	+5.37
Full Model	**74.35**	**61.74**	**66.12**	**72.27**	**+8.71**
Concat + Add	67.78	53.43	57.78	65.63	+1.24
Concat + Sub	69.56	52.49	55.97	66.27	+1.16
Concat + Mul	69.80	53.34	58.08	66.76	+2.08

Table 7: Ablation study on relation-level representation learning (F_1-score, %). *Concat* refers to performing a nonlinear projection on the concatenation of two word-level representations.

approach.

In addition, we also compare some element-wise operations (Bordes et al., 2013): *addition*, *subtraction*, and *multiplication*. As shown in Table 7, their performance improvements are not significant enough and sometimes even negative. This suggests that simple vector operations are insufficient to learn the relation-level information between words.

3.6 Case Study

We analyze and discuss our approach through several representative examples from the test set, presented in Table 6.

The first two examples indicate that our approach produces fewer boundary errors than the previous table-filling method. Especially in the $2nd$ example, the aspect terms and opinion terms are difficult to identify correctly. TF_{Double} outputs boundary-misspecified triplets given boundary-misspecified aspect terms and opinion terms, while our approach rejects these boundary errors as expected. We believe future work could attempt to correct these boundary errors. The $3rd$ example presents an overlapping case of aspect terms and opinion terms. The previous table-filling method fails to deal with the overlapping case, but our approach successfully solves it. In the $4th$ example, although *FAST* is a positive word, *refute* reverses the corresponding sentiment polarity. Both our approach and TF_{Double} make the wrong prediction, suggesting that more sentiment knowledge is required to improve the modeling ability for sentiment expressions. The $5th$ example reflects the worrying problem of incomplete annotation in the existing datasets.

4 Related Work

4.1 Aspect-Opinion Co-Extraction

In recent years, Aspect-Based Sentiment Analysis has attracted lots of researchers' interest (He et al., 2019; Yan et al., 2021; Zhang et al., 2021c; Liang et al., 2021; Mao et al., 2021; Liang et al., 2022a,b; Cao et al., 2022). As one of the most fundamental tasks in ABSA, aspect term extraction has been studied in many prior works (Hu and Liu, 2004; Yin et al., 2016; Xu et al., 2018; Hu et al., 2019; Wei et al., 2020; Wang et al., 2021b). The sentiment expression of aspect terms often depends on opinion terms, and thus opinion terms can be applied as clues to extract aspect terms and determine corresponding sentiment polarity. As a result, the amount of related aspect and opinion co-extraction work has been gradually increasing (Wang et al., 2016, 2017; Li and Lam, 2017; Li et al., 2018; Fan et al., 2019; Chen et al., 2020; Zhao et al., 2020).

To explicitly capture the relation between aspect and opinion terms, Fan et al. (2019) introduce a new task, Target-oriented Opinion Words Extraction (TOWE), which aims to extract the corresponding opinion words (*i.e.*, opinion terms) for a given target (*i.e.*, aspect term). Furthermore, Peng et al. (2020) present the Aspect Sentiment Triplet Extraction (ASTE) task.

Subsequent works address the ASTE task by transforming it into the position-aware tagging problem (Xu et al., 2020), the machine reading comprehension task (Chen et al., 2021a; Mao et al., 2021), the table-filling problem (Wu et al., 2020; Zhang et al., 2020; Chen et al., 2021b; Jing et al., 2021; Chen et al., 2022), the span-relation extraction problem (Xu et al., 2021; Li et al., 2022a), and the sequence generation task (Yan et al., 2021; Zhang et al., 2021c; Mukherjee et al., 2021; Lu et al., 2022).

4.2 Table-Filling Approach

The table-filling approach is initially proposed for joint entity and relation extraction. Miwa and Sasaki (2014) first cast joint entity and relation extraction as a table-filling problem. Gupta et al. (2016) improve it by a bi-RNN structure. Zhang et al. (2017) introduce global normalization and syntactic features. Similarly, Adel and Schütze (2017) jointly normalize all predictions of table-filling through the extension of the linear-chain CRF. The above methods first encode the text sequence and then obtain the table representation by feature concatenation. For a stronger table representation, Tran and Kavuluru (2019) and Wang and Lu (2020) employ CNNs and Multi-Dimensional RNNs (MDRNNs) as the table encoder.

5 Conclusion

This paper proposes a Boundary-Driven Table-Filling (BDTF) approach for the Aspect Sentiment Triplet Extraction (ASTE) task. BDTF transforms the ASTE task into detection and classification of relation regions in a two-dimensional table, solving the problems of *relation inconsistency* and *boundary insensitivity* in previous table-filling methods. In addition, to support BDTF, this paper develops an effective relation learning approach to learn the table representation, which can fully exploit word-to-word interactions and relation-to-relation interactions. Experiments on several public datasets show that our approach significantly outperforms existing methods. Further analysis shows that our approach produces fewer boundary errors and can solve the overlapping issue that previous table-filling approaches suffer from. Ablation study demonstrates the effectiveness of each component in the relation representation learning approach.

Acknowledgments

We thank the anonymous reviewers for their valuable suggestions to improve the quality of this work. This work was partially supported by the National Natural Science Foundation of China 62006062 and 62176076, Shenzhen Foundational Research Funding JCYJ20200109113441941, JCYJ20210324115614039, The Major Key Project of PCL2021A06, Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies 2022B1212010005.

Limitations

Even though the proposed approach significantly outperforms previous methods on several public benchmarks, it suffers from the following limitations:

- The table-filling methods need to construct a two-dimensional table representation of word-pair relations. If the vector dimensions are the same, the size of the table representation will be significantly larger than that of the sequence representation. Therefore, compared to other methods, the table-filling methods take up more memory. This problem also appears in our approach. Detailed memory usage and training time are presented in Appendix A.4.

- The proposed approach has lower recall than precision. This is because our approach significantly reduces wrong predictions but does not increase correct predictions by much.

We believe that addressing the above limitations without compromising the original advantages can further improve the model.

References

Heike Adel and Hinrich Schütze. 2017. Global normalization of convolutional neural networks for joint entity and relation classification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1723–1729, Copenhagen, Denmark. Association for Computational Linguistics.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Jiahao Cao, Rui Liu, Huailiang Peng, Lei Jiang, and Xu Bai. 2022. Aspect is not you need: No-aspect differential sentiment framework for aspect-based sentiment analysis. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1599–1609, Seattle, United States. Association for Computational Linguistics.

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.

Shaowei Chen, Jie Liu, Yu Wang, Wenzheng Zhang, and Ziming Chi. 2020. Synchronous double-channel recurrent network for aspect-opinion pair extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6515–6524, Online. Association for Computational Linguistics.

Shaowei Chen, Yu Wang, Jie Liu, and Yuelin Wang. 2021a. Bidirectional machine reading comprehension for aspect sentiment triplet extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):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, Online. Association for Computational Linguistics.

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3694, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Markus Eberts and Adrian Ulges. 2020. Span-based joint entity and relation extraction with transformer pre-training. In *ECAI 2020*, pages 2006–2013. IOS Press.

Zhifang Fan, Zhen Wu, Xin-Yu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2509–2518, Minneapolis, Minnesota. Association for Computational Linguistics.

Yuhao Feng, Yanghui Rao, Yuyao Tang, Ninghua Wang, and He Liu. 2021. Target-specified sequence labeling with multi-head self-attention for target-oriented opinion words extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1805–1815, Online. Association for Computational Linguistics.

Lei Gao, Yulong Wang, Tongcun Liu, Jingyu Wang, Lei Zhang, and Jianxin Liao. 2021. Question-driven span labeling model for aspect–opinion pair extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(14):12875–12883.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2537–2547, Osaka, Japan. The COLING 2016 Organizing Committee.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2019. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 504–515, Florence, Italy. Association for Computational Linguistics.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019. Open-domain targeted sentiment analysis via span-based extraction and classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 537–546, Florence, Italy. Association for Computational Linguistics.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 168–177, New York, NY, USA. Association for Computing Machinery.

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*.

Hongjiang Jing, Zuchao Li, Hai Zhao, and Shu Jiang. 2021. Seeking common but distinguishing difference, a joint aspect-based sentiment analysis model. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3910–3922, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019. A unified model for opinion target extraction and target sentiment prediction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6714–6721.

Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. 2018. Aspect term extraction with history attention and selective transformation. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4194–4200. AAAI Press.

Xin Li and Wai Lam. 2017. Deep multi-task learning for aspect term extraction with memory interaction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2886–2892, Copenhagen, Denmark. Association for Computational Linguistics.

You Li, Yongdong Lin, Yuming Lin, Liang Chang, and Huibing Zhang. 2022a. A span-sharing joint extraction framework for harvesting aspect sentiment triplets. *Knowledge-Based Systems*, 242:108366.

You Li, Chaoqiang Wang, Yuming Lin, Yongdong Lin, and Liang Chang. 2022b. Span-based relational graph transformer network for aspect–opinion pair extraction. *Knowl. Inf. Syst.*, 64(5):1305–1322.

Bin Liang, Xiang Li, Lin Gui, Yonghao Fu, Yulan He, Min Yang, and Ruifeng Xu. 2022a. Few-shot aspect category sentiment analysis via meta-learning. *ACM Transactions on Information Systems (TOIS)*.

Bin Liang, Hang Su, Lin Gui, Erik Cambria, and Ruifeng Xu. 2022b. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 235:107643.

Bin Liang, Rongdi Yin, Jiachen Du, Lin Gui, Yulan He, Min Yang, and Ruifeng Xu. 2021. Embedding refinement framework for targeted aspect-based sentiment analysis. *IEEE Transactions on Affective Computing*.

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

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

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. pages 5755–5772.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(15):13543–13551.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling joint entity and relation extraction with table representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1858–1869, Doha, Qatar. Association for Computational Linguistics.

Rajdeep Mukherjee, Tapas Nayak, Yash Butala, Sourangshu Bhattacharya, and Pawan Goyal. 2021. PASTE: A tagging-free decoding framework using pointer networks for aspect sentiment triplet extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9279–9291, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. volume 34, pages 8600–8607.

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

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

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Haris 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.

Florian Schmidt. 2019. Generalization in generation: A closer look at exposure bias. pages 157–167.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Tung Tran and Ramakanth Kavuluru. 2019. Neural metric learning for fast end-to-end relation extraction. *arXiv preprint arXiv:1905.07458*.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1706–1721, Online. Association for Computational Linguistics.

Peiyi Wang, Lianzhe Huang, Tianyu Liu, Damai Dai, Runxin Xu, Houfeng Wang, Baobao Chang, and Zhifang Sui. 2021a. Explicit interaction network for aspect sentiment triplet extraction. *arXiv preprint arXiv:2106.11148*.

Qianlong Wang, Zhiyuan Wen, Qin Zhao, Min Yang, and Ruifeng Xu. 2021b. Progressive self-training with discriminator for aspect term extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 257–268, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 616–626, Austin, Texas. Association for Computational Linguistics.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

Zhenkai Wei, Yu Hong, Bowei Zou, Meng Cheng, and Jianmin Yao. 2020. Don't eclipse your arts due to small discrepancies: Boundary repositioning with a pointer network for aspect extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3678–3684, Online. Association for Computational Linguistics.

Shengqiong Wu, Hao Fei, Yafeng Ren, Donghong Ji, and Jingye Li. 2021. Learn from syntax: Improving pair-wise aspect and opinion terms extraction with rich syntactic knowledge. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 3957–3963. International Joint Conferences on Artificial Intelligence Organization. Main Track.

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.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Double embeddings and CNN-based sequence labeling for aspect extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 592–598, Melbourne, Australia. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Lu Xu, Hao Li, Wei Lu, and Lidong Bing. 2020. Position-aware tagging for aspect sentiment triplet extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2339–2349, Online. Association for Computational Linguistics.

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, Online. Association for Computational Linguistics.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2979–2985. AAAI Press.

Samson Yu Bai Jian, Tapas Nayak, Navonil Majumder, and Soujanya Poria. 2021. *Aspect Sentiment Triplet*

Extraction Using Reinforcement Learning, page 3603–3607. Association for Computing Machinery, New York, NY, USA.

Chen Zhang, Qiuchi Li, Dawei Song, and Benyou Wang. 2020. A multi-task learning framework for opinion triplet extraction. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 819–828, Online. Association for Computational Linguistics.

Jingyuan Zhang, Zequn Zhang, Zhi Guo, Li Jin, Kang Liu, and Qing Liu. 2021a. Enhancement of target-oriented opinion words extraction with multiview-trained machine reading comprehension model. *Computational Intelligence and Neuroscience*, 2021.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2017. End-to-end neural relation extraction with global optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1730–1740, Copenhagen, Denmark. Association for Computational Linguistics.

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

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

Yue Zhang, Tao Peng, Ridong Han, Jiayu Han, Lin Yue, and Lu Liu. 2022. Synchronously tracking entities and relations in a syntax-aware parallel architecture for aspect-opinion pair extraction. *Applied Intelligence*, pages 1–16.

He Zhao, Longtao Huang, Rong Zhang, Quan Lu, and Hui Xue. 2020. SpanMlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3239–3248, Online. Association for Computational Linguistics.

A Additional Results

A.1 Datasets and Experiment Settings

We additionally evaluate our approach on ASTE-Data-v1 (Peng et al., 2020) and two AOPE-Data (Fan et al., 2019; Chen et al., 2020). Their statistics are detailed in Table 8.

On the two AOPE-Data, the validation set is not explicitly defined. For a fair comparison, we follow the settings of previous works (Chen et al., 2020; Wu et al., 2020). For the AOPE-Data provided by Fan et al. (2019), we randomly select 20% of the training set as the validation set; for the AOPE-Data provided by Chen et al. (2020), we publish the best results on the test set. All results are the average of 5 runs.

Provider	Dataset	Split	#Sent	#A	#O	#P/#T
Peng et al. (2020)	**Rest 14**	Train	1300	2079	2145	2145
		Dev	323	530	524	524
		Test	496	849	862	862
	Lap 14	Train	920	1283	1265	1265
		Dev	228	317	337	337
		Test	339	475	490	490
	Rest 15	Train	593	834	923	923
		Dev	148	225	238	238
		Test	318	426	455	455
	Rest 16	Train	842	1183	1289	1289
		Dev	210	291	316	316
		Test	320	444	465	465
Fan et al. (2019)	**Rest 14**	Train	1625	2539	2722	3062
		Test	500	864	888	1030
	Lap 14	Train	1151	1626	1625	1871
		Test	343	481	498	565
	Rest 15	Train	754	1076	1192	1871
		Test	325	436	469	493
	Rest 16	Train	1079	1512	1661	1770
		Test	329	456	485	524
Chen et al. (2020)	**Rest 14**	Train	3041	3693	3512	2809
		Test	800	1134	1014	936
	Lap 14	Train	3045	2359	2500	1535
		Test	800	653	677	380
	Rest 15	Train	1315	1205	1217	1231
		Test	685	542	516	516

Table 8: Statistics of ASTE-Data-v1 (Peng et al., 2020) and two AOPE-Data (Fan et al., 2019; Chen et al., 2020). #Sent, #A, #O, #P, and #T represent the number of sentences, aspect terms, opinion terms, pairs, and triplets, respectively.

A.2 Results on ASTE-Data-v1

The experimental results on ASTE-Data-v1 are presented in Table 9. According to these results, our approach consistently achieves the best performance in F_1-score.

A.3 Results on Aspect-Opinion Pair Extraction

Aspect-Opinion Pair Extraction (AOPE) aims to extract aspect terms and opinion terms along with their relations. The only difference between AOPE and ASTE is that the sentiment is

Model	Rest 14			Lap 14			Rest 15			Rest 16		
	P.	*R.*	F_1	*P.*	*R.*	F_1	*P.*	*R.*	F_1	*P.*	*R.*	F_1
Two-stage(Peng et al., 2020)	44.18	62.99	51.89	40.40	47.24	43.50	40.97	54.68	46.79	46.76	62.97	53.62
OTE-MTL(Zhang et al., 2020)	66.04	56.25	60.62	50.52	39.71	44.31	57.51	43.96	49.76	64.68	54.97	59.36
JET$^o_{M=6}$(BERT)(Xu et al., 2020)	67.97	60.32	63.92	58.47	43.67	50.00	58.35	51.43	54.67	64.77	61.29	62.98
GTS-BERT(Wu et al., 2020)	70.92	69.49	70.20	57.52	51.92	54.58	59.29	58.07	58.67	68.58	66.60	67.58
S^{3}E^2(Chen et al., 2021b)	69.08	64.55	66.74	59.43	46.23	52.01	61.06	56.44	58.66	71.08	63.13	66.87
TOP(Huang et al., 2021)	63.59	73.44	68.16	57.84	<u>59.33</u>	58.58	54.53	<u>63.30</u>	58.59	63.57	71.98	67.52
Dual-MRC(Mao et al., 2021)	71.55	69.14	70.32	57.39	53.88	55.58	63.78	51.87	57.21	68.60	66.24	67.40
BMRC(Chen et al., 2021a)	71.32	<u>70.09</u>	70.69	<u>65.12</u>	54.41	59.27	63.71	58.63	61.05	67.74	68.56	68.13
Span-BART (Yan et al., 2021)	-	-	<u>72.46</u>	-	-	57.59	-	-	60.11	-	-	69.98
Double-Encoder(Jing et al., 2021)	73.96	67.87	70.78	65.13	57.03	<u>60.81</u>	<u>64.86</u>	<u>63.30</u>	<u>64.07</u>	<u>74.77</u>	<u>72.20</u>	<u>73.46</u>
TGA+SFI(Wang et al., 2021a)	<u>77.03</u>	67.46	71.92	62.71	54.53	58.33	64.62	60.62	62.55	68.45	70.61	69.51
BDTF(Ours)	76.71	**74.01**	**75.33**	**68.30**	55.10	**60.99**	**66.95**	**65.05**	**65.97**	73.43	**73.64**	**73.51**

Table 9: Results on ASTE-Data-v1 (Peng et al., 2020) (%).

Model	Rest 14	Lap 14	Rest 15
HAST+IOG*	63.14	58.97	58.84
JERE-MHS*	67.81	58.69	60.17
HAST+RD*	73.55	64.05	65.20
DE-CNN+RD*	71.02	61.11	64.19
IMN+RD*	73.69	62.98	65.56
SPAN+RD*	74.17	65.99	67.55
RINATE+RD*	74.34	64.17	65.42
SDRN(Chen et al., 2020)	76.48	67.13	70.94
MT-TSMSA(Feng et al., 2021)	<u>76.69</u>	<u>68.18</u>	<u>71.64</u>
BDTF (Ours)	**80.03**	**71.48**	**72.77**

Table 10: Results on Aspect-Opinion Pair Extraction on Chen et al. (2020) (F_1-score, %). The results with $*$ are retrieved from Chen et al. (2020).

not required in AOPE. Considering their similarity, we also evaluate our approach on the AOPE task. To make our approach applicable to the AOPE task, we modify the type space of relation regions from $\{\texttt{POS}, \texttt{NEU}, \texttt{NEG}, \texttt{Invalid}\}$ to $\{\texttt{Pair}, \texttt{Invalid}\}$. The corresponding experimental results are listed in Table 10 and 11. It can be observed that our approach also yields outstanding results on the AOPE task. On the dataset provided by Fan et al. (2019), many methods (Wu et al., 2021; Liu et al., 2022; Li et al., 2022a,b) introduce additional syntactic features to learn better feature representations. Although we do not use syntactic features, our approach still surpasses them, demonstrating its effectiveness.

A.4 Memory Usage & Training Time

In this section, we list the memory usage and training time for the comparison and improvement by subsequent work. We conduct experiments on Rest 14 of ASTE-Data-v2 with the batch size of 1. The results are shown in Table 12. We can observe that although the proposed BDTF substantially outperforms the previous method in F_1-score, it requires more memory to run. We believe that reducing the memory usage can further improve our approach.

Model	Rest14	Lap14	Rest15	Rest16
SpanMlt(Zhao et al., 2020)	75.60	68.66	64.48	71.78
GTS(Wu et al., 2020)	75.53	65.67	67.53	74.62
SDRN*(Chen et al., 2020)	74.91	68.50	70.08	76.92
LAGCN(Wu et al., 2021)	76.62	68.88	68.91	76.59
MRC-MVT(Zhang et al., 2021a)	77.02	67.35	68.63	75.99
QDSL(Gao et al., 2021)	<u>78.05</u>	70.20	71.22	77.28
MT-TSMSA(Feng et al., 2021)	78.37	69.33	69.13	78.39
STER(Zhang et al., 2022)	74.96	67.64	69.30	75.89
MAIN(Liu et al., 2022)	77.54	69.86	70.92	77.97
SRGT(Li et al., 2022b)	76.78	70.47	<u>71.92</u>	<u>79.36</u>
SSJE(Li et al., 2022a)	78.02	<u>72.51</u>	69.53	78.96
BDTF (Ours)	**79.39**	**72.92**	**72.62**	**79.65**

Table 11: Results of Aspect-Opinion Pair Extraction on Fan et al. (2019) (F_1-score, %). The results with $*$ are retrieved from Feng et al. (2021).

Model	Memory Usage	Training Time	F_1(%)
Span-ASTE	3173MB	108 seconds	71.62
BDTF(d=32)	6117MB	128 seconds	74.34
BDTF(d=64)	8103MB	135 seconds	74.73

Table 12: The comparison of memory usage and training time. We train these models on a single Tesla V100 32GB. *Training time* refers to the time it takes to train a model for one epoch on the training set.