Seeking Common but Distinguishing Difference, A Joint Aspect-based Sentiment Analysis Model

Hongjiang Jing[1,2,3,†], **Zuchao Li**[1,2,3,†], **Hai Zhao**[1,2,3,*] **and Shu Jiang**[1,2,3]

[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, China
[3]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University

{jinghj,charlee,jshmjs45}@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn

Abstract

Aspect-based sentiment analysis (ABSA) task consists of three typical subtasks: aspect term extraction, opinion term extraction, and sentiment polarity classification. These three subtasks are usually performed jointly to save resources and reduce the error propagation in the pipeline. However, most of the existing joint models only focus on the benefits of encoder sharing between subtasks but ignore the difference. Therefore, we propose a joint ABSA model, which not only enjoys the benefits of encoder sharing but also focuses on the difference to improve the effectiveness of the model. In detail, we introduce a dual-encoder design, in which a pair encoder especially focuses on candidate aspect-opinion pair classification, and the original encoder keeps attention on sequence labeling. Empirical results show that our proposed model shows robustness and significantly outperforms the previous state-of-the-art on four benchmark datasets.

1 Introduction

Sentiment analysis is a task that aims to retrieve the sentiment polarity based on three levels of granularities: document level, sentence level, and entity and aspect level (Liu, 2012), which is under the urgent demands of several society scenarios (Preethi et al., 2017; Cobos et al., 2019; Islam and Zibran, 2017; Novielli et al., 2018). Recently, the aspect-based sentiment analysis (ABSA) task (Pontiki et al., 2014), focusing on excavating the specific aspect from an annotated review, has aroused much attention from researchers, in which this paper mainly concerns the aspect/opinion term extraction and sentiment classification task. The latest benchmark proposed by Peng et al. (2020) formulates the relevant information into a triplet: target aspect object, opinion clue, and sentiment polarity orientation. Thus, the concerned aspect term extraction becomes a task of Aspect Sentiment Triplet Extraction (ASTE). Similarly, the relevant information is formulated into a pair with aspect term and sentiment polarity, and the task is defined as Aspect Term Extraction and Sentiment Classification (AESC). Figure 1 shows an example of ASTE and AESC.

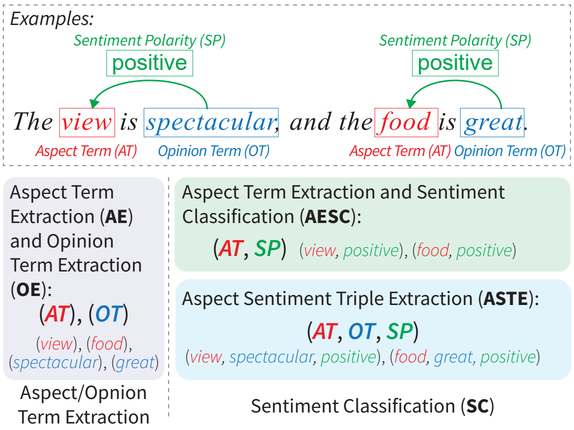

Examples:
Sentiment Polarity (SP)
positive
Sentiment Polarity (SP)
positive
The view is spectacular, and the food is great.
Aspect Term (AT)
Opinion Term (OT)
Aspect Term (AT)
Opinion Term (OT)
Aspect Term Extraction (AE) and Opinion Term Extraction (OE):
(AT), (OT)
(view), (food), (spectacular), (great)
Aspect Term Extraction and Sentiment Classification (AESC):
(AT, SP) (view, positive), (food, positive)
Aspect Sentiment Triple Extraction (ASTE):
(AT, OT, SP)
(view, spectacular, positive), (food, great, positive)
Aspect/Opnion Term Extraction
Sentiment Classification (SC)

Figure 1: The subtasks in our proposed model.

Two early methods handle the triplet extraction task efficiently (Zhang et al., 2020a; Huang et al., 2021). Both are typically composed of a sequence representation layer to predict the aspect/opinion term mentions and a classification layer to infer the sentiment polarity of the predicted mention pair of the last layer. However, as is often the case, such model design may easily result in that the errors of the upper prediction layer would hurt the accuracy of the lower layer during the training procedure.

To tackle the error cascading phenomenon on the pipeline model, a growing trend of jointly modeling these subtasks in one shot appears. Xu et al. (2020) proposed a joint model using a sequence tagging method, based on the bidirectional Long Short-Term Memory (LSTM) and Conditional Random Fields (CRF). However, they found that if a

[*] Corresponding author. [†] These authors made equal contribution. This work was supported by Key Projects of National Natural Science Foundation of China (U1836222 and 61733011).

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3910–3922
November 7–11, 2021. ©2021 Association for Computational Linguistics

tagged mention belongs to more than one triplet, this method will be ineffective. Zhang et al. (2020a) proposed a multi-task learning approach with the aid of dependency parsing on tail word pair of corresponding aspect-opinion pair. However, this non-strict dependency parsing may miss capturing structural information of term span. Meanwhile, the many-target to one-opinion issue is not effectively handled.

The promising results achieved by machine reading comprehension (MRC) frameworks on solving many other NLP tasks (Li et al., 2020, 2019a) also inspires the ASTE task. Mao et al. (2021) and Chen et al. (2021) attempted to design question-answer pair in terms of MRC to formulate the triplet extraction. Nevertheless, both need to make the converted question correspond one-to-one to the designed question manually, hence increasing computation complexity.

Among these joint models, Wu et al. (2020) transformed the sequence representation into the two-dimension space and argued that the word-pair under at least one assumption could represent the aspect-opinion pair as input of different encoders. Although this model indicated significant improvement, it treated the word-pair without taking span boundary of aspect term and opinion term into consideration and incorporated nonexistent pre-defined aspect-opinion pairs.

Considering the problems mentioned above, we propose a dual-encoder model based on a pre-trained language model by jointly fine-tuning multiple encoders on the ABSA task. Similar to prior work, our framework uses a shared sequence encoder to represent the aspect terms and opinion terms in the same embedding space. Moreover, we introduce a pair encoder to represent the aspect-opinion pair on the span level. Thus, our dual-encoder model could learn from the ABSA subtasks individually and benefit from each other in an end-to-end manner.

Experiments on benchmark datasets show that our model significantly outperforms previous approaches at the aspect level. We also conduct a series of experiments to analyze the gain of additional representation from the proposed dual-encoder structure.

The contributions of our work are as follows:

- We propose a jointly optimized dual-encoder model for ABSA to boost the performance of ABSA tasks.
- We apply an attention mechanism to allow information transfer between words to promote the model to know the word pairs before inference.
- We achieve state-of-the-art performance on benchmark datasets at the time of submission.

2 Our Approach

2.1 Problem Formulation

In this paper, we split the ABSA task into two periods: aspect/opinion term extraction and sentiment classification (SC), as shown in Figure 1. The aspect/opinion term extraction subtask extracts the aspect terms (AT) and opinion terms (OT) in the sentences without considering the sentiment polarities (SP). Furthermore, according to the sentiment polarity tagging style of the dataset, the SC subtask is divided into two categories: ASTE, tagging SP on AT and OT, and AESC, which tags SP only on AT.

In particular, we denote $\mathbf{AT}$, $\mathbf{OT}$ and $\mathbf{SP}$ as the set of predefined aspect terms, opinion terms and sentiment polarities, respectively, where $AT \in \mathbf{AT}$, $OT \in \mathbf{OT}$, and $SP \in \mathbf{SP} = \{\text{POS, NEU, NEG}\}$. Given a sentence s consisting of n tokens $\omega_1, \omega_2, ..., \omega_n$, we denote T as the sentence output of our model. Specifically, for the ASTE task, $T = \{(AT, OT, SP)\}$, and for the AESC task, $T = \{(AT, SP)\}$.

2.2 Model Overview

Our approach for the ABSA task is designed to subtly modeling high affinity between aspect/opinion pair and ground truth by effectively leveraging the pair representation. As shown in Figure 2, our dual-encoder comprises two modules: (1) a sequence encoder, a Transformer network initialized with the pre-trained language model to represent AT and OT with the corresponding context. (2) a pair encoder, encoding the aspect-opinion pair (for ASTE) or aspect-aspect pair (for AESC) for each sentiment polarity.

2.3 Token Representation

For a length-n input sentence $s = \omega_1, ..., \omega_n$, besides the word-level representation $\mathbf{x}_{\text{word}}$, we also feed the characters of the word into the LSTM to generate the character-level representation $\mathbf{x}_{\text{char}}$. Additionally, the pre-trained language model provides the contextualized representation $\mathbf{x}_{\text{plm}}$. Finally, we concatenate these three representations

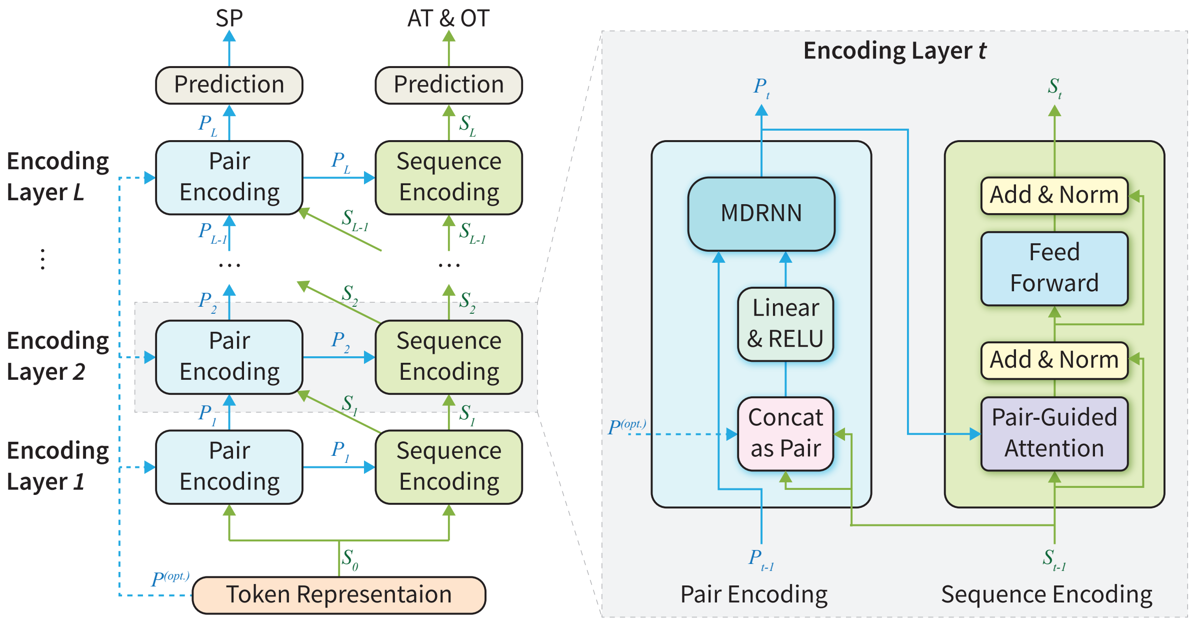


Figure 2: The framework of our model. Dashed lines are for optional components.

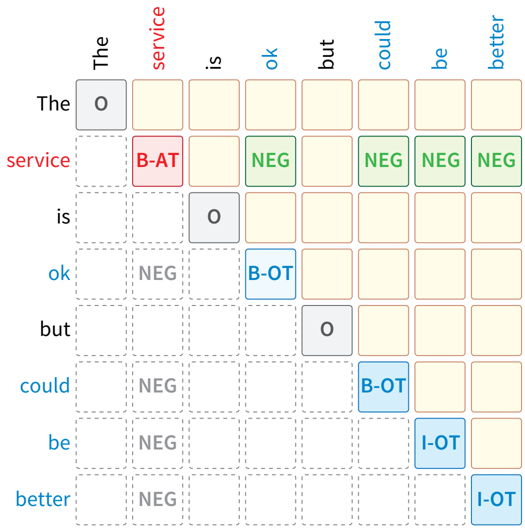
of each word to feed into the dual-encoder:

$$\mathbf{x}_i = [\mathbf{x}_{\text{char}}; \mathbf{x}_{\text{word}}; \mathbf{x}_{\text{plm}}]. \qquad (1)$$

In our proposed dual-encoder architecture, we still treat the ASTE/AESC task as a unified sequence tagging task in previous work: for a given sentence s, where AT and OT on the main diagonal are annotated with B/I/O (Begin, Inside, Outside), each entry $E_{i,j}$ of the upper triangular matrix denotes the pair (ω_i, ω_j) from the input sentence. Our work is partially motivated by Wu et al. (2020) but significantly different.

First, we improve the word-level pair representation to span-level pair representation with more accurate boundary information fed into our model. The tagging scheme of our model is illustrated in Figure 3, in which the main diagonal are filled with AT and OT accompanying entries to the right of the main diagonal with span pairs. Compared to (Wu et al., 2020), our method may heavily reduce the redundancy aroused by AT and OT tags at the right of the main diagonal.

Second, we consider the context information on both two-dimension spaces and the historical information with the utilization of the recurrent neural network (RNN). However, Wu et al. (2020) merely adopted a single encoder which based on DE-CNN (Xu et al., 2018)/BiLSTM/BERT (Devlin et al., 2019) to establish token representation, and they formulated the final word-pair representation by a simplified method of attention-guided word-pair concatenation.



	The	service	is	ok	but	could	be	better
The	O							
service		B-AT		NEG		NEG	NEG	NEG
is			O					
ok		NEG		B-OT				
but					O			
could		NEG				B-OT		
be		NEG					I-OT	
better		NEG						I-OT

Figure 3: A tagging example for our model.

Thus, our dual-encoder could jointly encode AT, OT (with the corresponding context on both dimensions), and AT-OT pairs with representation information sharing.

2.4 Sequence Encoder

Following the previous work of Vaswani et al. (2017), we construct the sequence encoder as a Transformer network.

Here we apply a stack of m self-attention layers, shown in Figure 2. Each layer consists of two sublayers: namely multi-head attention sublayer, feed-forward sublayer, at the top of each sublayer followed with both residual connection and layer normalization.

2.4.1 Multi-head Attention Sublayer

In this section, the token representation $\mathbf{x}_i$ is fed into a multi-head attention sublayer.

At first of our sequence encoder, the token representation $\mathbf{x}_i$ will be mapped into vector space as query $\mathbf{Q}_i$, key $\mathbf{K}_i$, value $\mathbf{V}_i$:

$$\begin{aligned}\mathbf{Q}_i &= \mathbf{x}_i\mathbf{W}_Q\\ \mathbf{K}_i &= \mathbf{x}_i\mathbf{W}_K\\ \mathbf{V}_i &= \mathbf{x}_i\mathbf{W}_V\end{aligned} \tag{2}$$

then the value vectors of all positions will be aggregated according to the normalized attention weight to get the single-head representation:

$$\text{SingleHead}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i) = \text{softmax}(\frac{\mathbf{Q}_i\mathbf{K}_i^T}{\sqrt{d/m}})V \tag{3}$$

where m is the number of heads, d is the dimension of $\mathbf{x}_i$, and in our sequence encoder, $\mathbf{Q} = \mathbf{W} = \mathbf{V} = \mathbf{x}_i$.

Then with multi-heads attention, our model builds up representations of the input sequence:

$$\begin{aligned}\mathbf{r}_i &= \text{MultiHead}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i)\\ &= \text{Concat}(\text{SingleHead}_{1,..,m}(\mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i))\mathbf{W}^O\end{aligned} \tag{4}$$

where $\mathbf{W}^O \in \mathbb{R}^d$. We adopt the residual connection and layer normalization (Ba et al., 2016) on $\mathbf{r}_i$ and $\mathbf{x}_i$:

$$\mathbf{a}_i = \text{LayerNorm}(\mathbf{r}_i + \mathbf{x}_i) \tag{5}$$

2.4.2 Feed-Forward Sublayer

The outputs of the multi-head attention are fed into a feed-forward network:

$$\mathbf{e}_i = \text{FFNN}(\mathbf{r}_i) = (\mathbf{a}_i\mathbf{W}_1 + b_1)\mathbf{W}_2 + b_2 \tag{6}$$

where $\mathbf{W}_1, \mathbf{W}_2, \in \mathbb{R}^{d\times d/m}$ and $b_1, b_2 \in \mathbb{R}^d$. At last, the sequence representation will be performed by layer normalization with residual connection:

$$\mathbf{S}_i = \text{LayerNorm}(\mathbf{e}_i + \mathbf{a}_i) \tag{7}$$

2.5 Pair Encoder

As shown in Eq. (3), our task-specific pair representation is an $n \times n$ matrix of vectors, where the vector at row i and column j represents i-th and j-th word pair of the input sentence. For the l-th layer of our network, we first add a Multi-Layer Perception (MLP) layer with ReLU (Nair and Hinton, 2010) to contextualize the concatenation of representations from the sequence encoder:

$$\mathbf{S}'_{l,i,j} = \text{ReLU}(\mathbf{MLP}([\mathbf{S}_{l-1,i}; \mathbf{S}_{l-1,j}])) \tag{8}$$

Then we utilize the multi-dimensional recurrent neural network (MDRNN) (Graves et al., 2007) and gated recurrent unit (GRU) (Cho et al., 2014) to contextualize $\mathbf{S}'_{l,i,j}$. The contextualized pair representation $\mathbf{P}_i$ is computed iteratively from the hidden states of each cell:

$$\mathbf{P}_{l,i,j} = \text{GRU}(\mathbf{S}'_{l,i,j}, \mathbf{P}_{l-1,i,j}, \mathbf{P}_{l,i-1,j}, \mathbf{P}_{l,i,j-1}) \tag{9}$$

The pair encoder does not consider only the word pair at neighboring rows and columns but also those of the previous layer.

2.6 Training

Given a sentence s with pre-defined tags AT, OT, and $SP \in \{\text{POS, NEU, NEG}\}$, we denote the AT or OT tag of token ω_i as a_i and the SP tag between the tokens ω_i and ω_j as t_{ij}. To predict the label of the posterior of the aspect/opinion terms $\hat{y}$, we apply a softmax layer on the sequence embedding of aspect/opinion terms $\mathbf{S}_l$. Similarly, to obtain the distribution of sentiment polarity type label $\hat{v}$, we apply softmax on the pair representation of $\mathbf{P}_l$:

$$P(\hat{y}|a_i, s) = \text{softmax}(\mathbf{W}_{term}\mathbf{S}_l) \tag{10}$$

$$P(\hat{v}|t_{ij}, s) = \text{softmax}(\mathbf{W}_{pola}\mathbf{P}_l) \tag{11}$$

where $\mathbf{W}_{term}$ and $\mathbf{W}_{pola}$ are learnable parameters.

At the training, we adopt the Cross-Entropy as our loss function. For the gold aspect and opinion term $a_i \in \mathbf{AT} \bigcap \mathbf{OT}$ and gold polarity $t_{ij} \in \mathbf{SP}$, the training losses are respectively:

$$\mathcal{L}_{term} = -\sum_{a_i \in \mathbf{AT}\cap\mathbf{OT}} \log(P(\hat{y} = y|a_i, s)) \tag{12}$$

$$\mathcal{L}_{pola} = -\sum_{t_{ij} \in \mathbf{SP}, i\neq j} \log(P(\hat{v} = v|t_{ij}, s)) \tag{13}$$

where the y and v are the gold annotations of corresponding terms.

To jointly train the model, we utilize the summation of these two loss functions as our training objective:

$$\mathcal{L} = \mathcal{L}_{term} + \mathcal{L}_{pola} \tag{14}$$

3 Experiments

3.1 Data

To make a fair comparison with previous methods, we adopt two versions of datasets for the ASTE task: (1) *ASTE-Data-V1*, originally provided by Peng et al. (2020) from the SemEval 2014 Task 4 (Pontiki et al., 2014), SemEval 2015 Task 12 (Pontiki et al., 2015) and SemEval 2016 Task 5 (Pontiki et al., 2016), and (2) *ASTE-Data-V2*, the refined version annotated by Xu et al. (2020), with additional annotation of implicitly overlapping triplets. Furthermore, the name of each dataset is composed of two parts. The former part denotes the year when the corresponding SemEval data was proposed, and the latter part is the domain name of the reviews on restaurant service or laptop sales. Data statistics of them is shown in Table 9.

Then, for the AESC task, we adopt the dataset annotated by Wang et al. (2017), which is composed of three datasets, and the statistics is shown in Table 10. The implementation details of our dual-encoder model are unfolded in Appendix A.2 for the sake of putting main concentration on our argument. Our code will be available at `https://github.com/Betahj/PairABSA`.

3.2 Results on the ASTE Task

Our model will compare to the following baselines on the ASTE task, and more details about these baseline models are listed in Appendix A.3.

1) **RINANTE+** (Peng et al., 2020).

2) **CMLA+** (Peng et al., 2020).

3) **Li-unified-R** (Peng et al., 2020).

4) **Peng et al.** (Peng et al., 2020).

5) **OTE-MTL** (Zhang et al., 2020a).

6) **JET** (Xu et al., 2020).

7) **GTS** (Wu et al., 2020).

8) **Huang et al.** (Huang et al., 2021).

The main results of all the models on the ASTE task are shown in Table 1. Compared with the best baseline model (Huang et al., 2021), our BERT-based dual-encoder model achieves an improvement by 1.39, 0.53, 0.68, and 2.92 absolute F_1 score on benchmark datasets. This result signifies that our dual-encoder model is capable of capturing the difference between AT/OT extraction subtask and SC subtask with the help of the additional pair encoder. Besides, our ALBERT-based model significantly outperforms all the other competitive methods on most metrics of 4 datasets *14Rest*, *14Lap*, *15Rest* and *16Rest* except for precision score of *15Rest*. Most notably, our ALBERT-based model achieves an improvement of 6.66, 4.72, 9.08, and 4.49 absolute F_1 score over all the baseline models on four benchmark datasets, respectively. This result demonstrates the superiority of our dual-encoder model. However, we notice that our precision score of *15Rest* is comparable to (Xu et al., 2020), which might be due to our model is more biased towards positive predictions but that the F1 score still suggests it is an overall improvement.

The similar phenomenon that our BERT-based dual-encoder model shows larger improvements in F1 scores on 14Rest (1.39) and 16Rest (2.92) than on 14Lap (0.53) and 15Rest (0.68) verifies the explanation of Xu et al. (2020) on large distribution differences of *14Rest* and *15Rest*. Nevertheless, we also observe a different phenomenon that our ALBERT-based dual-encoder model achieves significant F_1 score improvements on *14Rest* (6.66) and *15Rest* (9.08), better than *14Lap* (4.72) and *16Rest* (4.49), makes a challenge to the explanation developed by Xu et al. (2020). From our perspective, it might be due to the different fitting degree between the distribution of *ASTE-Data-V2* datasets and corresponding pre-trained language models. Additionally, we evaluate our model on the *ASTE-Data-V1* datasets and then experimental results further demonstrate the effectiveness of our dual-encoder model. These results are shown in Table 8 of the Appendix.

3.3 Results on the AESC Task

For the AESC task, our model will compare to the following baselines:

1) **SPAN-BERT** (Hu et al., 2019).

2) **IMN-BERT** (Hu et al., 2019).

3) **RACL-BERT** (Chen and Qian, 2020).

4) **Mao et al.** (Mao et al., 2021).

To investigate whether the performance of our model on the AESC task maintains the same efficiency as the ASTE task, we conduct a series of experiments on AESC datasets. Results of all the models on the AESC task are shown in Table 2.

Models	*14Rest*			*14Lap*			*15Rest*			*16Rest*		
	P.	*R.*	F_1	*P.*	*R.*	F_1	*P.*	*R.*	F_1	*P.*	*R.*	F_1
CMLA+	39.18	47.13	42.79	30.09	36.92	33.16	34.56	39.84	37.01	41.34	42.10	41.72
RINANTE+	31.42	39.38	34.95	21.71	18.66	20.07	29.88	30.06	29.97	25.68	22.30	23.87
Li-unified-R	41.04	67.35	51.00	40.56	44.28	42.34	44.72	51.39	47.82	37.33	54.51	44.31
(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
OTE-MTL	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-BiLSTM	71.41	53.00	60.84	58.02	40.11	47.43	64.57	44.33	52.57	70.17	55.95	62.26
JET^t	66.76	49.09	56.58	52.00	35.91	42.48	59.77	42.27	49.52	63.59	50.97	56.59
JET^o	61.50	55.13	58.14	53.03	33.89	41.35	64.37	44.33	52.50	70.94	57.00	63.21
GTS$_{+BERT}$	71.76	59.09	64.81	57.12	53.42	55.21	54.71	55.05	54.88	65.89	66.27	66.08
JET$^t_{+BERT}$	63.44	54.12	58.41	53.53	43.28	47.86	**68.20**	42.89	52.66	65.28	51.95	57.85
JET$^o_{+BERT}$	70.56	55.94	62.40	55.39	47.33	51.04	64.45	51.96	57.53	70.42	58.37	63.83
(Huang et al., 2021)$_{+BERT}$	63.59	73.44	68.16	57.84	59.33	58.58	54.53	63.30	58.59	63.57	71.98	67.52
Ours$_{+BERT}$	67.95	71.23	69.55	62.12	56.38	59.11	58.55	60.00	59.27	70.65	70.23	70.44
Ours$_{+ALBERT}$	**75.20**	**74.45**	**74.82**	**66.67**	**60.26**	**63.30**	66.74	**69.69**	**67.67**	**71.40**	**74.32**	**72.01**

Table 1: Results on *ASTE-Data-V2* test datasets. Baseline results are directly retrieved from (Huang et al., 2021). The extensive experiment of *ASTE-Data-V1* test datasets are supplemented in the Appendix.

Models	*14Rest*			*14Lap*			*15Rest*		
	AE	*OE*	*AESC*	*AE*	*OE*	*AESC*	*AE*	*OE*	*AESC*
SPAN-BERT	86.71	-	73.68	82.34	-	61.25	74.63	-	62.29
IMN-BERT	84.06	85.10	70.72	77.55	81.00	61.73	69.90	73.29	60.22
RACL-BERT	86.38	87.18	75.42	81.79	79.72	63.40	73.99	76.00	66.05
(Mao et al., 2021)	86.60	-	75.95	82.51	-	65.94	75.08	-	65.08
Baseline$_{+BERT}$	86.64	85.59	70.20	80.03	80.52	57.81	72.24	75.72	62.91
Ours$_{+BERT}$	86.94	85.80	70.49	80.26	80.61	57.98	72.68	75.94	63.19
Ours$_{+ALBERT}$	86.52	85.82	74.19	81.80	80.47	61.51	**75.42**	**78.86**	64.82

Table 2: Results for AESC on the test datasets annotated by Wang et al. (2017). Baseline results are directly retrieved from (Mao et al., 2021). The best result of each evaluation metric is bolded.

Compared with the best baseline model of Mao et al. (2021), the performance of our model is not comparable except for the absolute F_1 score on AE and OE of *15Rest* dataset. Then, to excavate the contribution of our dual-encoder structure on the AESC task, we evaluate our model on the baseline without the pair encoder. From Table 2 we can see that the performance of our dual-encoder model is comparable on the AESC task than single-encoder structure. The AESC task is only a simplified version of the ASTE task without taking AE/OE paring and sentiment polarity classification into consideration reversely, which is the training objective of our joint model with the help of task-specific structure design. Consequently, our model is incapable of functioning well in the AESC task.

4 Ablation Studies

4.1 Different Pre-trained Language Models

We conduct the experiment on the *14Lap* of *ASTE-Data-V2* datasets to excavate the performance of three frequently utilized pre-trained language models (PLMs): XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019) and ALBERT (Lan et al., 2020).

Table 3 shows that ALBERT helps achieve the best result among these four PLMs. However, even with BERT as the baseline model (Xu et al., 2020; Huang et al., 2021), our model also performs better. We also notice that, different from most models, our model is sensitive to different PLMs. Specifically, the absolute F_1 score between BERT and RoBERTa, ALBERT is 3.90 and 7.05, respectively. It demonstrates that our model performance could effectively be boosted by our choice of PLM, and thus we choose ALBERT as our base encoder.

PLM	*P.*	*R.*	F_1
BERT	59.63	53.23	56.25
XLNet	63.24	51.20	56.59
RoBERTa	61.79	58.60	60.15
ALBERT	66.67	60.26	63.30

Table 3: Comparison of our model with different pre-trained language models on *14Lap* test set of *ASTE-Data-V2*.

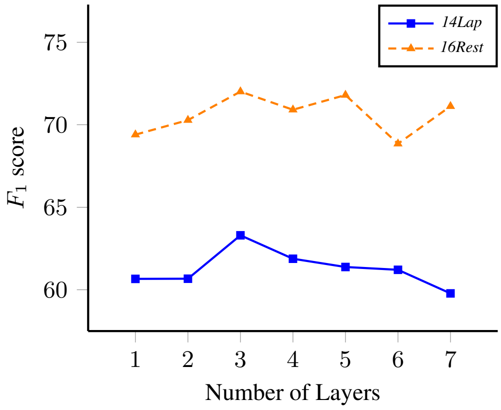


Figure 4: The impact of number of encoder layers on model performance.

4.2 Dual-encoder Structure

Therefore, the joint modeling method must take not only the fitting degree between individual modules and subtasks but also the difference of each module into consideration.

Settings	*P.*	*R.*	F_1
Default Setting	66.67	60.26	63.30
w/o Pair Encoder	58.16	59.15	58.65
w/o Interaction	64.55	58.88	61.58

Table 4: Ablation of our dual-encoder structure on *14Lap* test set of *ASTE-Data-V2*.

4.3 Number of Encoder Layers

The results with different numbers of encoder layers are in Figure 4. Generally, the performance of triplet extraction synchronously increases with the number of encoder layers of both dataset distributions. Nevertheless, when the number of encoder layers exceeds 3, the performance shows a continuous decreasing trend, except that on *16Rest* when the number of encoder layers is increased to 7, the performance increases by nearly 2.5 absolute F_1 score. Despite this inconsistent phenomenon, to mainly consider computational/time complexities, we adopt 3 as the number of encoders.

4.4 The Impact of The Number of GRU

Settings	*P.*	*R.*	F_1
Uni-directional	63.51	59.52	61.45
Bi-directional	64.96	58.60	61.61
Quad-directional	66.67	60.26	63.30

Table 5: Ablation of different settings of multi-dimensional recurrent neural networks on *14Lap* test set of *ASTE-Data-V2*.

Table 5 shows the results with different settings of multi-dimensional recurrent neural networks. The *Uni-directional* denotes the hidden state from forward GRU results in one quadrant of same dimension space, the *Bi-directional* denotes the hidden state from forward and backward GRU results in two quadrants of same dimension space, and *Quad-directional* denotes the hidden state from forward and backward GRU results in four quadrants of same dimension space. We observe that the *Quad-directional* setting significantly outperforms the other two settings. It is also noteworthy that the performance gap between *Bi-directional* and *Uni-directional* dimensions is much lower than the gap between *Quad-directional* and *Bi-directional* dimensions, which might be the reason why most previous work using bidirectional modelings cannot perform well. Thus, we choose *Quad-directional* as the dimensional setting of our multi-dimensional RNNs.

4.5 The Effect of Character-level Representation

To investigate the contribution of character-level representation to our input sequence, we remove the character-level representation generated by LSTM. Experimental result shows that the performance decreases by 0.44 absolute F_1 score.

5 Case Study

To investigate why our model far exceeds the baseline models, we conduct a case study of three typical cases from *14Lap* test dataset of *ASTE-Data-V1*, as shown in Table 6.

From **Example-1**, we observe that our model is able to handle the one-to-one case. However, our dual-encoder structure is more biased towards coordinative relation between *colors* and *speedy*. More cases we investigated further demonstrating that our model performs slightly worse on on-to-one than one-to-many and many-to-many relation types. From **Example-2**, we see that our model can tackle the one-opinion to many-target problem. However, most previous works are even unable to tackle one-opinion to two-target. From **Example-3**, we observe that our model is capable of well handling the one-target to many-opinion problem, which is neglected by most of the existing work but important for triplet extraction. Because many sentences compose conflicting sentiments on target, the model will fail to recognize the opposite

Example-1	Also stunning colors and speedy.
gold	Also [stunning]$_{\mathrm{OT}|\mathrm{POS}_{t_1}}$ [colors]$_{\mathrm{AT}|\mathrm{POS}_{t_1}}$ and speedy.
predict	Also [stunning]$_{\mathrm{OT}|\mathrm{POS}_{t_1}|\mathrm{POS}_{t_2}}$ [colors]$_{\mathrm{AT}|\mathrm{POS}_{t_1}}$ and [speedy]$_{\mathrm{AT}|\mathrm{POS}_{t_2}}$.
Example-2	Excellent performance, usability, presentation and time response.
gold	[Excellent]$_{\mathrm{OT}|\mathrm{POS}_{t_1}|\mathrm{POS}_{t_2}|\mathrm{POS}_{t_3}|\mathrm{POS}_{t_4}}$ [performance]$_{\mathrm{AT}|\mathrm{POS}_{h_1}}$, [usability]$_{\mathrm{AT}|\mathrm{POS}_{h_2}}$, [presentation]$_{\mathrm{AT}|\mathrm{POS}_{h_3}}$ and [time response]$_{\mathrm{AT}|\mathrm{POS}_{h_4}}$.
predict	[Excellent]$_{\mathrm{OT}|\mathrm{POS}_{t_1}|\mathrm{POS}_{t_2}|\mathrm{POS}_{t_3}|\mathrm{POS}_{t_4}}$ [performance]$_{\mathrm{AT}|\mathrm{POS}_{h_1}}$, [usability]$_{\mathrm{AT}|\mathrm{POS}_{h_2}}$, [presentation]$_{\mathrm{AT}|\mathrm{POS}_{h_3}}$ and [time response]$_{\mathrm{AT}|\mathrm{POS}_{h_4}}$.
Example-3	OSX Lion is a great performer..extremely fast and reliable.
gold	[OSX Lion]$_{\mathrm{AT}|\mathrm{POS}_{h_1}|\mathrm{POS}_{h_2}|\mathrm{POS}_{h_3}}$ is a [great]$_{\mathrm{OT}|\mathrm{POS}_{t_1}}$ performer..extremely [fast]$_{\mathrm{OT}|\mathrm{POS}_{t_2}}$ and [reliable]$_{\mathrm{OT}|\mathrm{POS}_{t_3}}$.
predict	[OSX Lion]$_{\mathrm{AT}|\mathrm{POS}_{h_1}|\mathrm{POS}_{h_2}|\mathrm{POS}_{h_3}}$ is a [great]$_{\mathrm{OT}|\mathrm{POS}_{t_1}}$ performer..extremely [fast]$_{\mathrm{OT}|\mathrm{POS}_{t_2}}$ and [reliable]$_{\mathrm{OT}|\mathrm{POS}_{t_3}}$.
Example-4	I am please with the products ease of use; out of the box ready; appearance and functionality.
gold	I am [please]$_{\mathrm{OT}|\mathrm{POS}_{t_1}|\mathrm{POS}_{t_2}|\mathrm{POS}_{t_3}}$ with the products [ease]$_{\mathrm{OT}|\mathrm{POS}_{t_4}}$ of [use]$_{\mathrm{AT}|\mathrm{POS}_{h_1}|\mathrm{POS}_{h_4}}$; out of the box ready; [appearance]$_{\mathrm{AT}|\mathrm{POS}_{h_2}}$ and [functionality]$_{\mathrm{AT}|\mathrm{POS}_{h_3}}$.
predict	I am [please]$_{\mathrm{OT}|\mathrm{POS}_{t_1}|\mathrm{POS}_{t_2}|\mathrm{POS}_{t_3}}$ with the products [ease]$_{\mathrm{OT}|\mathrm{POS}_{t_4}}$ of [use]$_{\mathrm{AT}|\mathrm{POS}_{h_1}|\mathrm{POS}_{h_4}}$; out of the box ready; [appearance]$_{\mathrm{AT}|\mathrm{POS}_{h_2}}$ and [functionality]$_{\mathrm{AT}|\mathrm{POS}_{h_3}}$.

Table 6: Case study of our proposed model, where AT/OT denote aspect term/opinion term, POS denotes sensitive polarity of positive, the subscript of sensitive polarity $\mathrm{h}_1/\mathrm{t}_1$ denotes the head/tail term of the 1st pair in terms of corresponding sentiment, etc.

polarity of the same AT when the incorrect AT extraction happens. Finally, we also observe that our model accurately inferences the boundary of *OSX Lion* span, which demonstrates the usefulness of our transformation that utilizes span to replace the word. From **Example-4**, we notice that our model could efficiently handle the complex situation of many-opinion to many-target with long-range dependency, which was particularly paid attention to but not solved well by Zhang et al. (2020a). It is due to incorporating the self-attention mechanism and GRU in two dimensions, and our model is sensitive to the difference between the proposed dual-encoder architecture. Collectively, these aforementioned cases demonstrate the robustness of our dual-encoder model.

6 Related Work

Recently, NLP has been developed rapidly (He et al., 2018; Li et al., 2018; Cai et al., 2018; Li et al., 2019b; Jiang et al., 2020; Zhang et al., 2021), and the process is further by deep neural networks (Parnow et al., 2021; Li et al., 2021a) and pre-trained language models (Li et al., 2021b; Zhang et al., 2020b). Aspect-based sentiment analysis was proposed by Pontiki et al. (2014) and also received lots of attention in recent years.

6.1 ASTE Task

The ASTE task aims to make triplet extraction of aspect terms, opinion terms, and sentiment polarity, which was introduced by Peng et al. (2020). In their work, they leveraged the sequence labeling method to extract aspect terms and target sentiment and utilized graph neural networks to detect candidate opinion terms. Zhang et al. (2020a) proposed a multi-task framework that decomposes the original ASTE task into two subtasks, sequence tagging of AT/OT, and word pair dependency parsing. For joint learning, Xu et al. (2020) proposed a sequence tagging framework based on LSTM-CRF. Wu et al. (2020) constructed an encoder-decoder model to handle this task with grid representation of aspect-opinion pairs. Then with the incorporation of a more specific semantic information guide for the proposed model, the ASTE is transformed as MRC task (Chen et al., 2021; Mao et al., 2021). Recently, Huang et al. (2021) proposed a sequence tagging-based model to perform representation learning on the ASTE task.

6.2 AESC Task

The AESC task is to perform aspect terms extraction and sentiment classification simultaneously. Hu et al. (2019) and Zhou et al. (2019) used a span-level sequence tagging method to tackle huge search space and sentiment inconsistency problems. Although the huge search space issue has been solved by Hu et al. (2019), there still exists a low-performance problem. Addressing this issue, Lin and Yang (2020) utilized a BERT encoder to contextualize shared information of target extraction and target classification subtasks. Meanwhile, they used two BiLSTM networks to encode the private information of each subtask, which greatly boosted the model performance.

7 Conclusion

In this paper, we observe the significant differences between the AT/OT extraction subtask and the SC subtask of ABSA for the joint model. Specifically, the results on 8 benchmark datasets with significant improvement over state-of-the-art baselines verify the effectiveness of our proposed model. Furthermore, to distinguish such differences and keep the shared part between different modules simultaneously, we construct a dual-encoder framework with representation learning and self-attention mechanism. In addition to the encoder-sharing approach, our dual-encoder framework can capture the difference between the subtasks by interconnecting encoders at each layer to share the critical information.

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Jiaxun Cai, Shexia He, Zuchao Li, and Hai Zhao. 2018. A full end-to-end semantic role labeler, syntactic-agnostic over syntactic-aware? In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2753–2765, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Shaowei Chen, Yu Wang, Jie Liu, and Yuelin Wang. 2021. Bidirectional machine reading comprehension for aspect sentiment triplet extraction. *arXiv preprint arXiv:2103.07665*.

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.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Ruth Cobos, Francisco Jurado, and Alberto Blázquez-Herranz. 2019. A content analysis system that supports sentiment analysis for subjectivity and polarity detection in online courses. *Rev. Iberoam. de Tecnol. del Aprendiz.*, 14(4):177–187.

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.

Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. 2007. Multi-dimensional recurrent neural networks. In *Artificial Neural Networks – ICANN 2007*, pages 549–558, Berlin, Heidelberg. Springer Berlin Heidelberg.

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.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2061–2071, Melbourne, Australia. Association for Computational Linguistics.

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.

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*.

Md Rakibul Islam and Minhaz F Zibran. 2017. Leveraging automated sentiment analysis in software engineering. In *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, pages 203–214. IEEE.

Shu Jiang, Hai Zhao, Zuchao Li, and Bao-Liang Lu. 2020. Document-level neural machine translation with document embeddings. *CoRR*, abs/2009.08775.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hao Li and Wei Lu. 2019. Learning explicit and implicit structures for targeted sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5478–5488, Hong Kong, China. Association for Computational Linguistics.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5849–5859, Online. Association for Computational Linguistics.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019a. Entity-relation extraction as multi-turn question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1340–1350, Florence, Italy. Association for Computational Linguistics.

Zuchao Li, Jiaxun Cai, Shexia He, and Hai Zhao. 2018. Seq2seq dependency parsing. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3203–3214, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019b. Dependency or span, end-to-end uniform semantic role labeling. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6730–6737. AAAI Press.

Zuchao Li, Zhuosheng Zhang, Hai Zhao, Rui Wang, Kehai Chen, Masao Utiyama, and Eiichiro Sumita. 2021a. Text compression-aided transformer encoding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zuchao Li, Hai Zhao, Shexia He, and Jiaxun Cai. 2021b. Syntax Role for Neural Semantic Role Labeling. *Computational Linguistics*, pages 1–46.

Peiqin Lin and Meng Yang. 2020. A shared-private representation model with coarse-to-fine extraction for target sentiment analysis. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4280–4289, Online. Association for Computational Linguistics.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

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. *CoRR*, abs/1907.11692.

Dehong Ma, Sujian Li, and Houfeng Wang. 2018. Joint learning for targeted sentiment analysis. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4737–4742, Brussels, Belgium. Association for Computational Linguistics.

Yue Mao, Yi Shen, Chao Yu, and Longjun Cai. 2021. A joint training dual-mrc framework for aspect based sentiment analysis. *arXiv preprint arXiv:2101.00816*.

Vinod Nair and Geoffrey E. Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel*, pages 807–814. Omnipress.

Nicole Novielli, Daniela Girardi, and Filippo Lanubile. 2018. A benchmark study on sentiment analysis for software engineering research. In *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*, pages 364–375. IEEE.

Kevin Parnow, Zuchao Li, and Hai Zhao. 2021. Grammatical error correction as GAN-like sequence labeling. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3284–3290, Online. Association for Computational Linguistics.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2020. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8600–8607. AAAI Press.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

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, 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.

G. Preethi, P. Venkata Krishna, Mohammad S. Obaidat, Vankadara Saritha, and Sumanth Yenduri. 2017. Application of deep learning to sentiment analysis for recommender system on cloud. In *International Conference on Computer, Information and Telecommunication Systems, CITS 2017, Dalian, China, July 21-23, 2017*, pages 93–97. IEEE.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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 Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3316–3322. AAAI Press.

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, 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.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Chen Zhang, Qiuchi Li, Dawei Song, and Benyou Wang. 2020a. 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.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020b. Semantics-aware BERT for language understanding. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9628–9635. AAAI Press.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2021. Retrospective reader for machine reading comprehension. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 14506–14514. AAAI Press.

Yan Zhou, Longtao Huang, Tao Guo, Jizhong Han, and Songlin Hu. 2019. A span-based joint model for opinion target extraction and target sentiment classification. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5485–5491. ijcai.org.

A Additional Results

A.1 Evaluation Metric

We adopt F_1 score as our evaluation metric as other baseline models. In precise, we measure the F1 score calculated between the final exact match of AT/OT span, AT/OT types and corresponding polarity predictions and gold triplets.

A.2 Implementation Details

For the token representation, we utilize 100-dimensional GloVe (Pennington et al., 2014) as initialization and restrict the update of word embedding. The hidden size is 200. The decay rate is 0.05, and the decay steps are 1000. Besides, to further boost the performance of our proposed model, we utilize the `ALBERT-xxlarge-v1` (Lan et al., 2020) as our pre-trained language model. We also use Adam with a learning rate of 0.001 and update parameters with a batch size of 24. Training is limited to the preset max steps. All models are implemented on the TITAN RTX. More implementation details are listed in Table 7.

Setting	Value
Char/ Char/Word/Glove	100
Word/Glove	100
Hidden Embedding Dim	200
Token Embedding Dim	100
Char Embedding Dim	30
Gradient Clipping	5.0
Batch Size	24
Optimizer	Adam
Learning Rate	$1e^{-3}$
Dropout Rate	0.5
Decay Rate	0.05
Number of Layer	3
Attention Heads	8

Table 7: Hyperparameter settings for our models

A.3 Baselines

Our model will compare to the following baselines on the ASTE task.

1) **RINANTE+** (Peng et al., 2020). The model RINANTE is modified from that by Ma et al. (2018). RINANTE+ is an LSTM-CRF model which first uses dependency relations of words to extract opinion and aspects with the sentiment. Then, all the candidate aspect-opinion pairs with position embedding are fed into the Bi-LSTM encoder to make a final classification.

2) **CMLA+** (Peng et al., 2020). The model is adjusted from the one by Wang et al. (2017), which is an attention-based model, following the same two-stage processing with dependency relations as RINANTE+.

3) **Li-unified-R** (Peng et al., 2020). Li-unified-R utilizes a modulated multi-layer LSTM encoder by Li and Lu (2019), and adopts the same aspect-opinion pair classification as RINANTE+.

4) **Peng et al.** (Peng et al., 2020). This model adopts GCN to capture dependency information, and at the second stage, uses the same strategy of RINANTE+ to fulfill triplet extraction.

5) **OTE-MTL** (Zhang et al., 2020a). A multi-task learning approach that incorporates word dependency parsing boosts the performance of triplet extraction.

6) **JET** (Xu et al., 2020). This model jointly extracts all the subtasks through a unified sequence labeling method. JET^{t} and JET^{o} denote two different tagging forms.

7) **GTS** (Wu et al., 2020). A sequence tagging model leverages the property element upper triangular matrix to model the extraction of aspect and opinion terms.

8) **Huang et al.** (Huang et al., 2021). The latest sequence labeling model which utilizes the restricted attention field mechanism and represents word-word perceivable pairs for the final classification.

For the AESC task, our model will compare to the following baselines:

1) **SPAN-BERT** (Hu et al., 2019). It is a BERT-based model which utilizes span representation to perform the AESC task.

2) **IMN-BERT** (Hu et al., 2019). It is a multi task learning model modified by He et al. (2019) and utilizes BERT as encoder to perform aspect term extraction and sentiment classification.

3) **RACL-BERT** (Chen and Qian, 2020). It is a multi-layer multi-task learning model with mutual information propagation to boost the performance of the AESC task.

4) **Mao et al.** (Mao et al., 2021). It is a dual-MRC architecture model to detect the AT/OT and corresponding sentiment polarity by means of a two-round query answering approach.

A.4 Results on ASTE-Data-V1 for ASTE

Results on the *ASTE-Data-V1* datasets also show the effectiveness of our model. But there is an interesting phenomenon that on the *16Rest* test set, the result of ALBERT-based model is lower than

Models	*14Rest*			*14Lap*			*15Rest*			*16Rest*		
	P.	*R.*	F$_1$	*P.*	*R.*	F$_1$	*P.*	*R.*	F$_1$	*P.*	*R.*	F$_1$
CMLA+	40.11	46.63	43.12	31.40	34.60	32.90	34.40	37.60	35.90	43.60	39.80	41.60
RINANTE+	31.07	37.63	34.03	23.10	17.70	20.00	29.40	26.90	28.00	27.10	20.50	23.30
Li-unified-R	41.44	68.79	51.68	42.25	42.78	42.47	43.34	50.73	46.69	38.19	53.47	44.51
(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
JET^{t}	70.39	51.68	59.72	57.98	36.33	44.67	61.99	43.74	51.29	68.99	51.18	58.77
JET^{o}	62.26	56.84	59.43	52.01	39.59	44.96	63.25	46.15	53.37	66.58	57.85	61.91
JET$^{t}_{+BERT}$	70.20	53.02	60.41	51.48	42.65	46.65	62.14	47.25	53.68	71.12	57.20	63.41
JET$^{o}_{+BERT}$	67.97	60.32	63.92	58.47	43.67	50.00	58.35	51.43	54.67	64.77	61.29	62.98
Ours$_{+BERT}$	73.96	67.87	70.78	65.21	60.82	62.94	64.86	63.30	64.07	**73.71**	**76.56**	**75.11**
Ours$_{+ALBERT}$	**77.32**	**75.52**	**76.41**	**68.65**	**61.22**	**64.72**	**68.36**	**66.81**	**67.18**	73.18	73.33	73.25

Table 8: Results on *ASTE-Data-V1* test datasets. Baseline results are directly retrieved from (Xu et al., 2020).

Dataset	*14Rest*		*14Lap*		*15Rest*		*16Rest*	
	Sentences	Target	Sentences	Target	Sentences	Target	Sentences	Target
ASTE-Data-V1-Train	1,300	2,145	920	1,265	593	923	842	1,289
ASTE-Data-V1-Valid	323	524	228	337	148	238	210	316
ASTE-Data-V1-Test	496	862	339	490	318	455	320	465
ASTE-Data-V2-Train	1,266	2,338	906	1,460	605	1,013	857	1,394
ASTE-Data-V2-Valid	310	577	219	346	148	249	210	339
ASTE-Data-V2-Test	492	994	328	543	322	485	326	514

Table 9: Statistics of the datasets used for the ASTE task.

Datasets	Sentence	Aspect	Opinion
Restaurant14-Train	3,044	3,699	3,484
Restaurant14-Test	800	1,134	1,008
Laptop14-Train	3,048	2,373	2,504
Laptop14-Test	800	654	674
Restaurant15-Train	1,315	1,199	1,210
Restaurant15-Test	685	542	510

Table 10: Statistics of the datasets used for the AESC task.

that of BERT-based model. It may be due to the inconsistent domain between the test set and the pre-trained language model.

A.5 Data Statistics

Table 9 and Table 10 show the statistics of the datasets we used.