Learning Span-Level Interactions for Aspect Sentiment Triplet Extraction

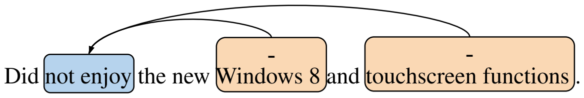
Lu Xu[*][1,2], **Yew Ken Chia**[*][1,2], **Lidong Bing**[2]
[1] Singapore University of Technology and Design
[2] DAMO Academy, Alibaba Group
{xu_lu,yewken_chia}@mymail.sutd.edu.sg
l.bing@alibaba-inc.com

Abstract

Aspect Sentiment Triplet Extraction (ASTE) is the most recent subtask of ABSA which outputs triplets of an aspect target, its associated sentiment, and the corresponding opinion term. Recent models perform the triplet extraction in an end-to-end manner but heavily rely on the interactions between each target word and opinion word. Thereby, they cannot perform well on targets and opinions which contain multiple words. Our proposed span-level approach explicitly considers the interaction between the whole spans of targets and opinions when predicting their sentiment relation. Thus, it can make predictions with the semantics of whole spans, ensuring better sentiment consistency. To ease the high computational cost caused by span enumeration, we propose a dual-channel span pruning strategy by incorporating supervision from the Aspect Term Extraction (ATE) and Opinion Term Extraction (OTE) tasks. This strategy not only improves computational efficiency but also distinguishes the opinion and target spans more properly. Our framework simultaneously achieves strong performance for the ASTE as well as ATE and OTE tasks. In particular, our analysis shows that our span-level approach achieves more significant improvements over the baselines on triplets with multi-word targets or opinions. [1]

1 Introduction

Aspect-Based Sentiment Analysis (ABSA) (Liu, 2012; Pontiki et al., 2014) is an aggregation of several fine-grained sentiment analysis tasks, and its various subtasks are designed with the aspect target as the fundamental item. For the example in Figure 1, the aspect targets are *"Windows 8"* and *"touchscreen functions"*. Aspect Sentiment Classification (ASC) (Dong et al., 2014; Zhang et al., 2016; Yang et al., 2017; Li et al., 2018a; Tang et al., 2019) is one of the most well-explored subtasks of ABSA and aims to predict the sentiment polarity of a given aspect target. However, it is not always practical to assume that the aspect target is provided. Aspect Term Extraction (ATE) (Yin et al., 2016; Li et al., 2018b; Ma et al., 2019) focuses on extracting aspect targets, while Opinion Term Extraction (OTE) (Yang and Cardie, 2012; Klinger and Cimiano, 2013; Yang and Cardie, 2013) aims to extract the opinion terms which largely determine the sentiment polarity of the sentence or the corresponding target term. Aspect Sentiment Triplet Extraction (ASTE) (Peng et al., 2019) is the most recently proposed subtask of ABSA, which forms a more complete picture of the sentiment information through the triplet of an aspect target term, the corresponding opinion term, and the expressed sentiment. For the example in Figure 1, there are two triplets: (*"Windows 8"*, *"not enjoy"*, Negative) and (*"touchscreen functions"*, *"not enjoy"*, Negative).



Triplets: (Windows 8, not enjoy, Negative);
(touchscreen functions, not enjoy, Negative).

Figure 1: An example of ASTE. The spans highlighted in orange are target terms, and the span in blue is opinion term. The "-" on top of target terms indicates negative sentiment.

The initial approach to ASTE (Peng et al., 2019) was a two-stage pipeline. The first stage extracts target terms and their sentiments via a joint labeling scheme [2], as well as the opinion terms with stan-

[*] Equal contribution. Lu Xu and Yew Ken Chia are under the Joint PhD Program between Alibaba and Singapore University of Technology and Design.

[1] We make our code publicly available at https://github.com/chiayewken/Span-ASTE.

[2] For example, the joint tag "B-POS" denotes the beginning

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 4755–4766
August 1–6, 2021. ©2021 Association for Computational Linguistics

dard BIOES [3] tags. The second stage then couples the extracted target and opinion terms to determine their paired sentiment relation. We know that in ABSA, the aspect sentiment is mostly determined by the opinion terms expressed on the aspect target (Qiu et al., 2011; Yang and Cardie, 2012). However, this pipeline approach breaks the interaction within the triplet structure. Moreover, pipeline approaches usually suffer from the error propagation problem.

Recent end-to-end approaches (Wu et al., 2020; Xu et al., 2020b; Zhang et al., 2020) can jointly extract the target and opinion terms and classify their sentiment relation. One drawback is that they heavily rely on word-to-word interactions to predict the sentiment relation for the target-opinion pair. Note that it is common for the aspect targets and opinions to contain multiple words, which accounts for roughly one-third of triplets in the benchmark datasets. However, the previous methods (Wu et al., 2020; Zhang et al., 2020) predict the sentiment polarity for each word-word pair independently, which cannot guarantee their sentiment consistency when forming a triplet. As a result, this prediction limitation on triplets that contain multi-word targets or opinions inevitably hurts the overall ASTE performance. For the example in Figure 1, by only considering the word-to-word interactions, it is easy to wrongly predict that *"enjoy"* expresses a positive sentiment on *"Windows"*. Xu et al. (2020b) proposed a position-aware tagging scheme to allow the model to couple each word in a target span with all possible opinion spans, i.e., aspect word to opinion span interactions (or vice versa, aspect span to opinion word interactions). However, it still cannot directly model the span-to-span interactions between the whole target spans and opinion spans.

In this paper, we propose a span-based model for ASTE (Span-ASTE), which for the first time directly captures the span-to-span interactions when predicting the sentiment relation of an aspect target and opinion pair. Of course, it can also consider the single-word aspects or opinions properly. Our model explicitly generates span representations for all possible target and opinion spans, and their paired sentiment relation is independently predicted for all possible target and opinion pairs. Span-based methods have shown encouraging performance on other tasks, such as coreference resolution (Lee et al., 2017), semantic role labeling (He et al., 2018a), and relation extraction (Luan et al., 2019; Wadden et al., 2019). However, they cannot be directly applied to the ASTE task due to different task-specific characteristics.

Our contribution can be summarized as follows:

- We tailor a span-level approach to explicitly consider the span-to-span interactions for the ASTE task and conduct extensive analysis to demonstrate its effectiveness. Our approach significantly improves performance, especially on triplets which contain multi-word targets or opinions.

- We propose a dual-channel span pruning strategy by incorporating explicit supervision from the ATE and OTE tasks to ease the high computational cost caused by span enumeration and maximize the chances of pairing valid target and opinion candidates together.

- Our proposed Span-ASTE model outperforms the previous methods significantly not only for the ASTE task, but also for the ATE and OTE tasks on four benchmark datasets with both BiLSTM and BERT encoders.

2 Span-based ASTE

2.1 Task Formulation

Let $X = \{x_1, x_2, ..., x_n\}$ denote a sentence of n tokens, let $S = \{s_{1,1}, s_{1,2}, ..., s_{i,j}, ..., s_{n,n}\}$ be the set of all possible enumerated spans in X, with i and j indicating the start and end positions of a span in the sentence. We limit the span length as $0 \leq j - i \leq L$. The objective of the ASTE task is to extract all possible triplets in X. Each sentiment triplet is defined as $(target, opinion, sentiment)$ where $sentiment \in \{Positive, Negative, Neutral\}$.

2.2 Model Architecture

As shown in Figure 2, Span-ASTE consists of three modules: sentence encoding, mention module, and triplet module. For the given example, the sentence is first input to the sentence encoding module to obtain the token-level representation, from which we derive the span-level representation for each enumerated span, such as *"did not enjoy"*, *"Windows 8"*. We then adopt the ATE and OTE tasks to supervise our proposed dual-channel span pruning strategy which obtains the pruned target and

of a target span with positive sentiment polarity.

[3] A common tagging scheme for sequence labeling, denoting "begin, inside, outside, end and single" respectively.

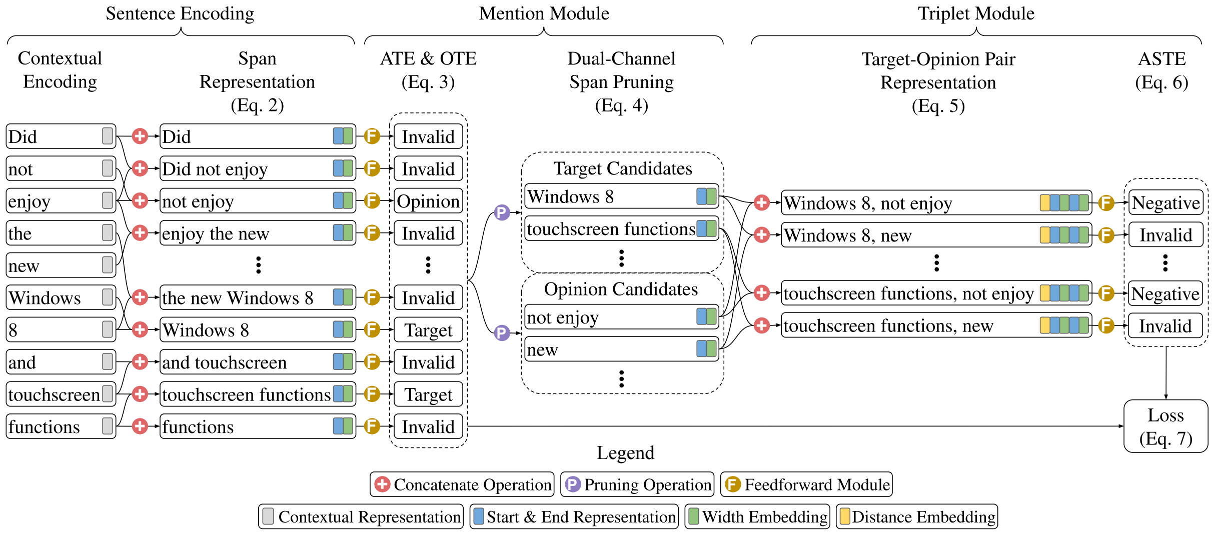


Figure 2: Span-ASTE model structure.

opinion candidates, such as *"Windows 8"* and *"not enjoy"* respectively. Finally, each target candidate and opinion candidate are coupled to determine the sentiment relation between them.

2.2.1 Sentence Encoding

We explore two encoding methods to obtain the contextualized representation for each word in a sentence: BiLSTM and BERT.

BiLSTM We first obtain the word representations $\{\mathbf{e_1}, \mathbf{e_2}, ..., \mathbf{e_i}, ..., \mathbf{e_n}\}$ from the 300-dimension pre-trained GloVe (Pennington et al., 2014) embeddings which are then contextualized by a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) layer. The i^{th} token is represented as:

$$\mathbf{h}_i = [\overrightarrow{\mathbf{h}_i};\ \overleftarrow{\mathbf{h}_i}] \qquad (1)$$

where $\overrightarrow{\mathbf{h}_i}$ and $\overleftarrow{\mathbf{h}_i}$ are the hidden states of the forward and backward LSTMs respectively.

BERT An alternative encoding method is to use a pre-trained language model such as BERT (Devlin et al., 2019) to obtain the contextualized word representations $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n]$. For words that are tokenized as multiple word pieces, we use mean pooling to aggregate their representations .

Span Representation We define each span representation $s_{i,j} \in S$ as:

$$\mathbf{s}_{i,j} = \begin{cases} [\mathbf{h}_i;\ \mathbf{h}_j;\ f_{width}(i,j)] & \text{if BiLSTM} \\ [\mathbf{x}_i;\ \mathbf{x}_j;\ f_{width}(i,j)] & \text{if BERT} \end{cases} \qquad (2)$$

where $f_{width}(i,j)$ produces a trainable feature embedding representing the span width (i.e., $j-i+1$). Besides the concatenation of the start token, end token, and width representations, the span representation $\mathbf{s}_{i,j}$ can also be formed by max-pooling or mean-pooling across all token representations of the span from position i to j. The experimental results can be found in the ablation study.

2.2.2 Mention Module

ATE & OTE Tasks We employ the ABSA subtasks of ATE and OTE to guide our dual-channel span pruning strategy through the scores of the predicted opinion and target span. Note that the target terms and opinion terms are not yet paired together at this stage. The mention module takes the representation of each enumerated span $\mathbf{s}_{i,j}$ as input and predicts the mention types $m \in \{Target, Opinion, Invalid\}$.

$$P(m|\mathbf{s}_{i,j}) = \text{softmax}(\text{FFNN}_m(\mathbf{s}_{i,j})) \qquad (3)$$

where FFNN denotes a feed-forward neural network with non-linear activation.

Pruned Target and Opinion For a sentence X of length n, the number of enumerated spans is $O(n^2)$, while the number of possible pairs between all opinion and target candidate spans is $O(n^4)$ at the later stage (i.e., the triplet module). As such, it is not computationally practical to consider all possible pairwise interactions when using a span-based approach. Previous works (Luan et al., 2019; Wadden et al., 2019) employ a pruning strategy to reduce the number of spans, but they only prune the spans to a single pool which is a mix of different mention types. This strategy does not fully consider

Dataset	Rest 14						Lap 14						Rest 15						Rest 16					
	#S	# +	# 0	# -	#SW	#MW	#S	# +	# 0	# -	#SW	#MW	#S	# +	# 0	# -	#SW	#MW	#S	# +	# 0	# -	#SW	#MW
Train	1266	1692	166	480	1586	752	906	817	126	517	824	636	605	783	25	205	678	335	857	1015	50	329	918	476
Dev	310	404	54	119	388	189	219	169	36	141	190	156	148	185	11	53	165	84	210	252	11	76	216	123
Test	492	773	66	155	657	337	328	364	63	116	291	252	322	317	25	143	297	188	326	407	29	78	344	170

Table 1: Statistics of datasets. #S denotes the number of sentences. # +, # 0, and # - denote the numbers of positive, neutral, and negative sentiment triplets respectively. #SW denotes the number of triplets where both target and opinion terms are single-word spans. #MW denotes the number of triplets where at least one of the target or opinion terms are multi-word spans.

the structure of an aspect sentiment triplet as it does not recognize the fundamental difference between a target and an opinion term. Hence, we propose to use a dual-channel pruning strategy which results in two separate pruned pools of aspects and opinions. This minimizes computational costs while maximizing the chance of pairing valid opinion and target spans together. The opinion and target candidates are selected based on the scores of the mention types for each span based on Equation 3:

$$\begin{aligned}\Phi_{target}(\mathbf{s}_{i,j}) &= P(m = target|\mathbf{s}_{i,j}) \\ \Phi_{opinion}(\mathbf{s}_{i,j}) &= P(m = opinion|\mathbf{s}_{i,j})\end{aligned} \quad (4)$$

We use the mention scores Φ_{target} and $\Phi_{opinion}$ to select the top candidates from the enumerated spans and obtain the target candidate pool $S^t = \{..., \mathbf{s}^t_{a,b}, ...\}$ and the opinion candidate pool $S^o = \{..., \mathbf{s}^o_{c,d}, ...\}$ respectively. To consider a proportionate number of candidates for each sentence, the number of selected spans for both pruned target and opinion candidates is nz, where n is the sentence length and z is a threshold hyper-parameter. Note that although the pruning operation prevents the gradient flow back to the FFNN in the mention module, it is already receiving supervision from the ATE and OTE tasks. Hence, our model can be trained end-to-end without any issue or instability.

2.2.3 Triplet Module

Target Opinion Pair Representation We obtain the target-opinion pair representation by coupling each target candidate representation $\mathbf{s}^t_{a,b} \in S^t$ with each opinion candidate representation $\mathbf{s}^o_{c,d} \in S^o$:

$$\mathbf{g}_{\mathbf{s}^t_{a,b},\mathbf{s}^o_{c,d}} = [\mathbf{s}^t_{a,b};\ \mathbf{s}^o_{c,d};\ f_{distance}(a, b, c, d)] \quad (5)$$

where $f_{distance}(a, b, c, d)$ produces a trainable feature embedding based on the distance (i.e., $min(|b - c|, |a - d|)$) between the target and opinion spans, following (Lee et al., 2017; He et al., 2018a; Xu et al., 2020b).

Sentiment Relation Classifier Then, we input the span pair representation $\mathbf{g}_{\mathbf{s}^t_{a,b},\mathbf{s}^o_{c,d}}$ to a feed-forward neural network to determine the probability of sentiment relation $r \in R = \{Positive, Negative, Neutral, Invalid\}$ between the target $\mathbf{s}^t_{a,b}$ and the opinion $\mathbf{s}^o_{c,d}$:

$$P(r|\mathbf{s}^t_{a,b}, \mathbf{s}^o_{c,d}) = \text{softmax}(\text{FFNN}_r(\mathbf{g}_{\mathbf{s}^t_{a,b},\mathbf{s}^o_{c,d}})) \quad (6)$$

$Invalid$ here indicates that the target and opinion pair has no valid sentiment relationship.

2.3 Training

The training objective is defined as the sum of the negative log-likelihood from both the mention module and triplet module.

$$\begin{aligned}\mathcal{L} = &- \sum_{\mathbf{s}_{i,j} \in S} \log P(m^*_{i,j}|\mathbf{s}_{i,j}) \\ &- \sum_{\mathbf{s}^t_{a,b} \in S^t, \mathbf{s}^o_{c,d} \in S^o} \log P(r^*|\mathbf{s}^t_{a,b}, \mathbf{s}^o_{c,d})\end{aligned} \quad (7)$$

where $m^*_{i,j}$ is the gold mention type of the span $\mathbf{s}_{i,j}$, and r^* is the gold sentiment relation of the target and opinion span pair $(\mathbf{s}^t_{a,b}, \mathbf{s}^o_{c,d})$. S indicates the enumerated span pool; S^t and S^o are the pruned target and opinion span candidates.

3 Experiment

3.1 Datasets

Our proposed Span-ASTE model is evaluated on four ASTE datasets released by Xu et al. (2020b), which include three datasets in the restaurant domain and one dataset in the laptop domain. The first version of the ASTE datasets are released by Peng et al. (2019). However, it is found that not all triplets are explicitly annotated (Xu et al., 2020b; Wu et al., 2020). Xu et al. (2020b) refined the datasets with the missing triplets and removed triplets with conflicting sentiments. Note that these

	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
BiLSTM	CMLA+ (Wang et al., 2017)†	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+ (Dai and Song, 2019)†	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 (Li et al., 2019)†	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. (2019)†	43.24	63.66	51.46	37.38	50.38	42.87	48.07	57.51	52.32	46.96	64.24	54.21
	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 (Wu et al., 2020)*	66.13	57.91	61.73	53.35	40.99	46.31	60.10	46.89	52.66	63.28	58.56	60.79
	JET$^o_{M=6}$ (Xu et al., 2020b)†	61.50	55.13	58.14	53.03	33.89	41.35	64.37	44.33	52.50	70.94	57.00	63.21
	Span-ASTE (Ours)	72.52	62.43	**67.08**	59.85	45.67	**51.80**	64.29	52.12	**57.56**	67.25	61.75	**64.37**
BERT	GTS (Wu et al., 2020)*	67.76	67.29	67.50	57.82	51.32	54.36	62.59	57.94	60.15	66.08	69.91	67.93
	JET$^o_{M=6}$ (Xu et al., 2020b) †	70.56	55.94	62.40	55.39	47.33	51.04	64.45	51.96	57.53	70.42	58.37	63.83
	Span-ASTE (Ours)	72.89	70.89	**71.85**	63.44	55.84	**59.38**	62.18	64.45	**63.27**	69.45	71.17	**70.26**

Table 2: Results on the test set of the ASTE task. †: The results are retrieved from Xu et al. (2020b). *: For a fair comparison, we reproduce the results using their released implementation code and configuration on the same ASTE datasets released by Xu et al. (2020b).

four benchmark datasets are derived from the SemEval Challenge (Pontiki et al., 2014, 2015, 2016), and the opinion terms are retrieved from (Fan et al., 2019). Table 1 shows the detailed statistics.

3.2 Experiment Settings

When using the BiLSTM encoder, the pre-trained GloVe word embeddings are trainable. The hidden size of the BiLSTM encoder is 300 and the dropout rate is 0.5. In the second setting, we fine-tune the pre-trained BERT (Devlin et al., 2019) to encode each sentence. Specifically, we use the uncased version of BERT$_{\text{base}}$. The model is trained for 10 epochs with a linear warmup for 10% of the training steps followed by a linear decay of the learning rate to 0. We employ AdamW as the optimizer with the maximum learning rate of 5e-5 for transformer weights and weight decay of 1e-2. For other parameter groups, we use a learning rate of 1e-3 with no weight decay. The maximum span length L is set as 8. The span pruning threshold z is set as 0.5. We select the best model weights based on the F_1 scores on the development set and the reported results are the average of 5 runs with different random seeds. [4]

3.3 Baselines

The baselines can be summarized as two groups: pipeline methods and end-to-end methods.

Pipeline For the pipeline approaches listed below, they are modified by Peng et al. (2019) to extract the aspect terms together with their associated sentiments via a joint labeling scheme, and opinion terms with BIOES tags at the first stage. At the second stage, the extracted targets and opinions are then paired to determine if they can form a valid triplet. Note that these approaches employ different methods to obtain the features for the first stage. CMLA+ (Wang et al., 2017) employs an attention mechanism to consider the interaction between aspect terms and opinion terms. RINANTE+ (Dai and Song, 2019) adopts a BiLSTM-CRF model with mined rules to capture the dependency relations. Li-unified-R (Li et al., 2019) uses a unified tagging scheme to jointly extract the aspect term and associated sentiment. Peng et al. (2019) includes dependency relation information when considering the interaction between the aspect and opinion terms.

End-to-end The end-to-end methods aim to jointly extract full triplets in a single stage. Previous work by Zhang et al. (2020) and Wu et al. (2020) independently predict the sentiment relation for all possible word-word pairs, hence they require decoding heuristics to determine the overall sentiment polarity of a triplet. JET (Xu et al., 2020b) models the ASTE task as a structured prediction problem with a position-aware tagging scheme to capture the interaction of the three elements in a triplet.

3.4 Experiment Results

Table 2 compares Span-ASTE with previous models in terms of Precision ($P.$), Recall ($R.$), and F_1 scores on four datasets. Under the F_1 metric, our model consistently outperforms the previous works for both BiLSTM and BERT sentence encoders. In most cases, our model significantly out-

[4] See Appendix for more experimental settings, and also the dev results on the four datasets.

performs other end-to-end methods in both precision and recall. We also observe that the two strong pipeline methods (Li et al., 2019; Peng et al., 2019) achieved competitive recall results, but their overall performance is much worse due to the low precision. Specifically, using the BiLSTM encoder with GloVe embedding, our model outperforms the best pipeline model (Peng et al., 2019) by 15.62, 8.93, 5.24, and 10.16 F_1 points on the four datasets. This result indicates that our end-to-end approach can effectively encode the interaction between target and opinion spans, and also alleviates the error propagation. In general, the other end-to-end methods are also more competitive than the pipeline methods. However, due to the limitations of relying on word-level interactions, their performances are less encouraging in a few cases, such as the results on Lap 14 and Rest 15. With the BERT encoder, all three end-to-end models achieve much stronger performance than their LSTM-based versions, which is consistent with previous findings (Devlin et al., 2019). Our approach outperforms the previous best results GTS (Wu et al., 2020) by 4.35, 5.02, 3.12, and 2.33 F_1 points on the four datasets.

3.5 Additional Experiments

As mentioned in Section 2.2.2, we employ the ABSA subtasks of ATE and OTE to guide our span pruning strategy. To examine if Span-ASTE can effectively extract target spans and opinion spans, we also evaluate our model on the ATE and OTE tasks on the four datasets. Table 3 shows the comparisons of our approach and the previous method GTS (Wu et al., 2020). [5] Without additional retraining or tuning, our model can directly address the ATE and OTE tasks, with significant performance improvement than GTS in terms of F_1 scores on both tasks. Even though GTS shows a better recall score on the Rest 16 dataset, the low precision score results in worse F_1 performance. The better overall performance indicates that our span-level method not only benefits the sentiment triplet extraction, but also improves the extraction of target and opinion terms by considering the semantics of each whole span rather than relying on decoding heuristics of tagging-based methods.

Dataset	Model	ATE			OTE		
		P.	*R.*	F_1	*P.*	*R.*	F_1
Rest 14	GTS	78.12	85.64	81.69	81.12	88.24	84.53
	Ours	83.56	87.59	**85.50**	82.93	89.67	**86.16**
Lap 14	GTS	76.63	82.68	79.53	76.11	78.44	77.25
	Ours	81.48	86.39	**83.86**	83.00	82.28	**82.63**
Rest 15	GTS	75.13	81.57	78.21	74.96	82.52	78.49
	Ours	78.97	84.68	**81.72**	77.36	84.86	**80.93**
Rest 16	GTS	75.06	89.42	81.61	78.99	88.71	83.57
	Ours	79.78	88.50	**83.89**	82.59	90.91	**86.54**

Table 3: Test results on the ATE and OTE tasks with BERT encoder. For reference, we include the results of the RACL framework (Chen and Qian, 2020) in the Appendix. RACL is the current state-of-the-art method for both tasks. However, their framework does not consider the pairing relation between each target and opinion, therefore it is difficult to have a completely fair comparison.

4 Analysis

4.1 Comparison of Single-word and Multi-word Spans

We compare the performance of Span-ASTE with the previous model GTS (Wu et al., 2020) for the following two settings in Table 4: Single-Word: Both target and opinion terms in a triplet are single-word spans, Multi-Word: At least one of the target or opinion terms in a triplet is a multi-word span. For the single-word setting, our method shows consistent improvement in terms of both precision and recall score on the four datasets, which results in the improvement of F_1 score. When we compare the evaluations for multi-word triplets, our model achieves more significant improvements for F_1 scores. Compared to precision, our recall shows greater improvement over the GTS approach. GTS heavily relies on word-pair interactions to extract triplets, while our methods explicitly consider the span-to-span interactions. Our span enumeration also naturally benefits the recall of multi-word spans. For both GTS and our model, multi-word triplets pose challenges and their F_1 results drop by more than 10 points, even more than 20 points for Rest 14. As shown in Table 1, comparing with the single-word triplets, multi-word triplets are common and account for one-third or even half of the datasets. Therefore, a promising direction for future work is to further improve the model's performance on such difficult triplets.

To identify further areas for improvement, we analyze the results for the ASTE task based on whether each sentiment triplet contains a multi-

[5] See Appendix for the target and opinion data statistics. Note that the JET model (Xu et al., 2020b) is not able to directly solve the ATE and OTE tasks unless the evaluation is conducted based on the triplet predictions. We include such comparisons in the Appendix.

	Mode	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
BERT	Single-Word	GTS	74.93	79.15	76.98	65.47	62.54	63.97	66.55	65.66	66.10	69.66	76.74	73.03
		Ours	79.12	79.60	79.36	68.09	65.98	67.02	70.23	70.71	70.47	71.66	77.91	74.65
		Δ	+4.19	+0.46	+2.38	+2.62	+3.44	+3.04	+3.68	+5.05	+4.37	+2.00	+1.16	+1.62
	Multi-Word	GTS	56.85	49.26	52.78	52.26	41.27	46.12	50.28	47.34	48.77	56.63	55.29	55.95
		Ours	61.64	55.79	58.57	54.63	44.44	49.02	50.70	57.45	53.87	62.43	63.53	62.97
		Δ	+4.79	+6.53	+5.78	+2.37	+3.17	+2.90	+0.42	+10.11	+5.10	+5.80	+8.24	+7.02

Table 4: Analysis with different evaluation modes on the ASTE task.

Dataset	Model	Multi-word Target			Multi-word Opinion		
		P.	*R.*	F_1	*P.*	*R.*	F_1
Rest 14	GTS	56.54	49.81	52.96	50.67	41.76	45.78
	Ours	65.96	57.62	**61.51**	49.43	47.25	**48.31**
Lap 14	GTS	55.11	44.09	48.99	37.50	26.09	**30.77**
	Ours	56.99	48.18	**52.22**	34.62	26.09	29.75
Rest 15	GTS	51.09	51.09	51.09	43.40	35.94	39.32
	Ours	55.33	60.58	**57.84**	37.18	45.31	**40.85**
Rest 16	GTS	62.69	65.12	63.88	28.26	24.07	26.00
	Ours	66.43	72.09	**69.14**	36.73	33.33	**34.95**

Table 5: Further comparison of test results for our model and GTS based on triplets of multi-word targets and opinions for the ASTE task.

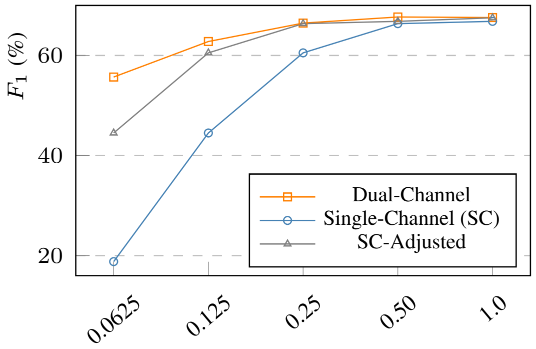


Figure 3: Dev results with respect to pruning threshold z which intuitively refers to the number of candidate spans to keep per word in the sentence.

word target or multi-word opinion term. From Table 5, the results show that the performance is lower when the triplet contains a multi-word opinion term. This trend can be attributed to the imbalanced data distribution of triplets which contain multi-word target or opinion terms.

4.2 Pruning Efficiency

To demonstrate the efficiency of the proposed dual-channel pruning strategy, we also compare it to a simpler strategy, denoted as Single-Channel (SC) which does not distinguish between opinion and target candidates. Figure 3 shows the comparisons. Note the mention module under this strategy does not explicitly solve the ATE and OTE tasks as it only predicts mention label $m \in \{Valid, Invalid\}$, where $Valid$ means the span is either a target or an opinion span and $Invalid$ means the span does not belong to the two groups. Given sentence length n and pruning threshold z, the number of candidates is limited to nz, and hence the computational cost scales with the number of pairwise interactions, n^2z^2. The dual-channel strategy considers each target-opinion pair where the pruned target and opinion candidate pools both have nz spans. Note that it is possible for the two pools to share some candidates. In comparison, the single-channel strategy considers each target-opinion pair where the target and opinion candidates are drawn from the same single pool of nz spans. In order to consider at least as many target and opinion candidates as the dual-channel strategy, the single-channel strategy has to scale the threshold z by two, which leads to 4 times more pairs and computational cost. We denote this setting in Figure 3 as SC-Adjusted. When controlling for computational efficiency, there is a significant performance difference between Dual-Channel and Single-Channel in F_1 score, especially for lower values of z. Although the performance gap narrows with increasing z, it is not practical for high values. According to our experimental results, we select the dual-channel pruning strategy with $z = 0.5$ for the reported model.

4.3 Qualitative Analysis

To illustrate the differences between the models, we present sample sentences from the ASTE test set with the gold labels as well as predictions from GTS (Wu et al., 2020) and Span-ASTE in Figure 4. For the first example, GTS correctly extracts the target term *"Windows 8"* paired with the opinion term *"not enjoy"*, but the sentiment is incorrectly predicted as positive. When forming the triplet, their decoding heuristic considers the sentiment inde-

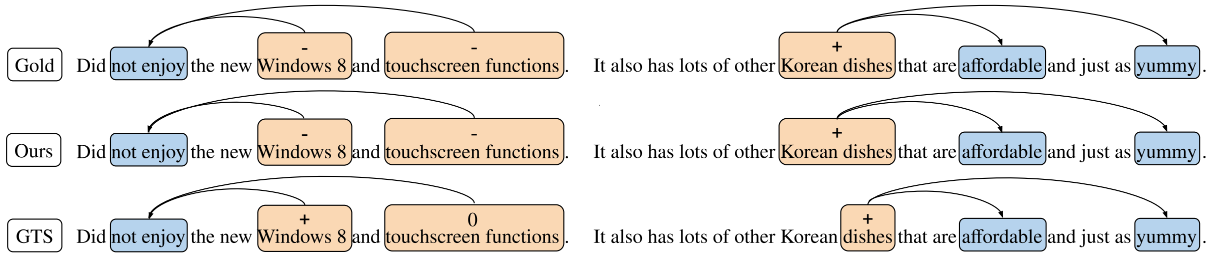

Gold
Did not enjoy the new Windows 8 and touchscreen functions.
-
-
It also has lots of other Korean dishes that are affordable and just as yummy.
+
Ours
Did not enjoy the new Windows 8 and touchscreen functions.
-
-
It also has lots of other Korean dishes that are affordable and just as yummy.
+
GTS
Did not enjoy the new Windows 8 and touchscreen functions.
+
0
It also has lots of other Korean dishes that are affordable and just as yummy.
+

Figure 4: Qualitative analysis. The target and opinion terms are highlighted in orange and blue respectively. Each arc indicates the pairing relation between target and opinion terms. The sentiment polarity of each triplet is indicated above the target terms.

pendently for each word-word pair: $\{$(*"Windows"*, *"not"*, Neutral), (*"8"*, *"not"*, Neutral), (*"Windows"*, *"enjoy"*, Positive), (*"8"*, *"enjoy"*, Positive)$\}$. Their heuristic votes the overall sentiment polarity as the most frequent label among the pairs. In the case of a tie (2 neutral and 2 positive), the heuristic has a predefined bias to assign the sentiment polarity to positive. Similarly, the word-level method fails to capture the negative sentiment expressed by *"not enjoy"* on the other target term *"touchscreen functions"*. In the second example, it incompletely extracts the target term *"Korean dishes"*, resulting in the wrong triplet. For both examples, our method is able to accurately extract the target-opinion pairs and determine the overall sentiment even when each term has multiple words.

4.4 Ablation Study

We conduct an ablation study to examine the performance of different modules and span representation methods, and the results are shown in Table 6. The average F_1 denotes the average dev results of Span-ASTE on the four benchmark datasets over 5 runs. Similar to the observation for coreference resolution (Lee et al., 2017), we find that the ASTE performance is reduced when removing the span width and distance embedding. This indicates that the positional information is still useful for the ASTE task as targets and opinions which are far apart or too long are less likely to form a valid span pair. As mentioned in Section 2.2.1, we explore two other methods (i.e., max pooling and mean pooling) to form span representations instead of concatenating the span boundary token representations. The negative results suggest that using pooling to aggregate the span representation is disadvantageous due to the loss of information that is useful for distinguishing valid and invalid spans.

Model	Average F_1	ΔF_1
Full model	**67.69**	
W/O width & distance embedding	66.45	-1.24
max pooling	66.09	-1.60
mean pooling	66.19	-1.53

Table 6: Ablation study on the development sets.

5 Related Work

Sentiment Analysis is a major Natural Language Understanding (NLU) task (Wang et al., 2019) and has been extensively studied as a classification problem at the sentence level (Raffel et al., 2020; Lan et al., 2020; Yang et al., 2020). Aspect-Based Sentiment Analysis (ABSA) (Pontiki et al., 2014) addresses various sentiment analysis tasks at a fine-grained level. As mentioned in the Section 1, the subtasks mainly include ASC (Dong et al., 2014; Zhang et al., 2016; Chen et al., 2017; He et al., 2018b; Li et al., 2018a; Peng et al., 2018; Wang and Lu, 2018; He et al., 2019; Li and Lu, 2019; Xu et al., 2020a), ATE (Qiu et al., 2011; Yin et al., 2016; Li et al., 2018b; Ma et al., 2019), OTE (Hu and Liu, 2004; Yang and Cardie, 2012; Klinger and Cimiano, 2013; Yang and Cardie, 2013). There is also another subtask named Target-oriented Opinion Words Extraction (TOWE) (Fan et al., 2019), which aim to extract the corresponding opinion words for a given target term. Another line of research focuses on addressing different subtasks together. Aspect and Opinion Term Co-Extraction (AOTE) aiming to extract the aspect and opinion terms together (Wang et al., 2017; Ma et al., 2019; Dai and Song, 2019) and is often treated as a sequence labeling problem. Note that AOTE does not consider the paired sentiment relationship between each target and opinion term. End-to-End ABSA (Li and Lu, 2017; Ma et al., 2018; Li et al., 2019; He et al., 2019) jointly extracts each aspect term and its associated sentiment in an end-to-end manner. A few other methods are recently proposed to jointly solve three or more subtasks of ABSA. Chen and Qian (2020) proposed a relation aware collaborative learning framework to unify the three fundamental subtasks and achieved strong performance on each subtask and combined task. While Wan et al. (2020) focused more on aspect category related subtasks, such as Aspect Category Extraction and Aspect Category and Target Joint Extraction. ASTE (Peng et al., 2019; Wu et al., 2020; Xu et al., 2020b; Zhang et al., 2020) is the most recent development of ABSA and its aim is to extract and form the aspect term, its associated sentiment, and the corresponding opinion term into a triplet.

6 Conclusions

In this work, we propose a span-level approach - Span-ASTE to learn the interactions between target spans and opinion spans for the ASTE task. It can address the limitation of the existing approaches that only consider word-to-word interactions. We also propose to include the ATE and OTE tasks as supervision for our dual-channel pruning strategy to reduce the number of enumerated target and opinion candidates to increase the computational efficiency and maximize the chances of pairing valid target and opinion candidates together. Our method significantly outperforms the previous methods for ASTE as well as ATE and OTE tasks and our analysis demonstrates the effectiveness of our approach. While we achieve strong performance on the ASTE task, the performance can be mostly attributed to the improvement on the multi-word triplets. As discussed in Section 4.1, there is still a significant performance gap between single-word and multi-word triplets, and this can be a potential area for future work.

References

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proc. of EMNLP*.

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proc. of ACL*.

Hongliang Dai and Yangqiu Song. 2019. Neural aspect and opinion term extraction with mined rules as weak supervision. In *Proc. of ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*.

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proc. of ACL*.

Zhifang Fan, Zhen Wu, Xinyu Dai, Shujian Huang, and Jiajun Chen. 2019. Target-oriented opinion words extraction with target-fused neural sequence labeling. In *Porc. of NAACL*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. 2018a. Jointly predicting predicates and arguments in neural semantic role labeling. In *Proc. of ACL*.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018b. Effective attention modeling for aspect-level sentiment classification. In *Proc. of COLING*.

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 *Proc. of ACL*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9:1735–80.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proc. of ACM SIGKDD*.

R. Klinger and P. Cimiano. 2013. Joint and pipeline probabilistic models for fine-grained sentiment analysis: Extracting aspects, subjective phrases and their relations. In *2013 IEEE 13th International Conference on Data Mining Workshops*.

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 *Proc. of ICLR*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proc. of EMNLP*.

Hao Li and Wei Lu. 2017. Learning latent sentiment scopes for entity-level sentiment analysis. In *Proc. of AAAI*.

Hao Li and Wei Lu. 2019. Learning explicit and implicit structures for targeted sentiment analysis. In *Proc. of EMNLP*.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018a. Transformation networks for target-oriented sentiment classification. In *Proc. of ACL*.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019. A unified model for opinion target extraction and target sentiment prediction. In *Proc. of AAAI*.

Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. 2018b. Aspect term extraction with history attention and selective transformation. In *Proc. of IJCAI*.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. In *Proc. of NAACL*.

Dehong Ma, Sujian Li, and Houfeng Wang. 2018. Joint learning for targeted sentiment analysis. In *Proc. of EMNLP*.

Dehong Ma, Sujian Li, Fangzhao Wu, Xing Xie, and Houfeng Wang. 2019. Exploring sequence-to-sequence learning in aspect term extraction. In *Proc. of ACL*.

Haiyun Peng, Yukun Ma, Yang Li, and Erik Cambria. 2018. Learning multi-grained aspect target sequence for chinese sentiment analysis. *Knowledge-Based Systems*, 148:167–176.

Haiyun Peng, Lu Xu, Lidong Bing, Fei Huang, Wei Lu, and Luo Si. 2019. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *Proc. of AAAI*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proc. of EMNLP*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammed AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proc. of SemEval*.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proc. of SemEval*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proc. of SemEval*.

Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. 2011. Opinion word expansion and target extraction through double propagation. *Computational Linguistics*, 37(1):9–27.

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

Jialong Tang, Ziyao Lu, Jinsong Su, Yubin Ge, Linfeng Song, Le Sun, and Jiebo Luo. 2019. Progressive self-supervised attention learning for aspect-level sentiment analysis. In *Proc. of ACL*.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proc. of EMNLP*.

Hai Wan, Yufei Yang, Jianfeng Du, Yanan Liu, Kunxun Qi, and Jeff Z. Pan. 2020. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. In *Proc. of AAAI*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. of ICLR*.

Bailin Wang and Wei Lu. 2018. Learning latent opinions for aspect-level sentiment classification. In *Proc. of AAAI*.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proc. of AAAI*.

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 EMNLP*.

Lu Xu, Lidong Bing, Wei Lu, and Fei Huang. 2020a. Aspect sentiment classification with aspect-specific opinion spans. In *Proc. of EMNLP*.

Lu Xu, Hao Li, W. Lu, and Lidong Bing. 2020b. Position-aware tagging for aspect sentiment triplet extraction. In *Proc. of EMNLP*.

Bishan Yang and Claire Cardie. 2012. Extracting opinion expressions with semi-Markov conditional random fields. In *Proc. of EMNLP*.

Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *Proc. of ACL*.

Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen. 2017. Attention based lstm for target dependent sentiment classification. In *Proc. of AAAI*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2020. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proc. of NeurIPS*.

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 *Proc. of IJCAI*.

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

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated neural networks for targeted sentiment analysis. In *Proc. of AAAI*.

A Additional Experimental Settings

We run our model experiments on a Nvidia Tesla V100 GPU, with CUDA version 10.2 and PyTorch version 1.6.0. The average run time for BERT-based model is 157 sec/epoch, 115 sec/epoch, 87 sec/epoch, and 111 sec/epoch for Rest 14, Lap 14, Rest 15, and Rest 16 respectively. The total number of parameters is 2.24M when GloVe is used, and is 110M when BERT base is used. The feed-forward neural networks in the mention module and triplet module have 2 hidden layers and hidden size of 150. We use ReLU activation and dropout of 0.4 after each hidden layer. We use Xavier Normal weight initialization for the feed-forward parameters. The span width and distance embeddings have 20 and 128 dimensions respectively. Their input values are bucketed (Gardner et al., 2017) before being fed to an embedding matrix lookup: [0, 1, 2, 3, 4, 5-7, 8-15, 16-31, 32-63, 64+]. During training, the model parameters are updated after each sentence which results in a batch size of 1. For each input text sequence, we restrict it to a maximum of 512 tokens.

B Additional Data Statistics

Table 9 shows the number of target terms and opinion terms on the four datasets.

C Dev Results

Table 10 shows the results of our model on the development datasets.

D Additional Comparisons

As mentioned by footnote 5 in Section 3.5, we cannot make a direct comparison with the JET model (Xu et al., 2020b), as it is not able to directly solve the ATE and OTE tasks unless the evaluation is conducted based on the triplet results. Table 7 shows such comparisons. Our proposed method generally outperforms the previous two end-to-end approaches on the four datasets.

Dataset	Model	ATE			OTE		
		P.	*R.*	F_1	*P.*	*R.*	F_1
Rest 14	$\text{JET}^{o}_{M=6}$	83.21	66.04	73.64	83.76	77.28	80.39
	GTS	83.25	81.49	82.36	86.55	86.65	**86.60**
	Ours	86.20	80.31	**83.15**	87.20	84.54	85.85
Lap 14	$\text{JET}^{o}_{M=6}$	83.33	68.03	74.91	77.16	75.53	76.34
	GTS	82.17	73.65	77.68	81.63	74.05	77.66
	Ours	87.69	75.38	**81.07**	85.61	76.58	**80.84**
Rest 15	$\text{JET}^{o}_{M=6}$	83.04	65.74	73.38	81.33	68.98	74.65
	GTS	80.95	74.77	77.74	80.96	76.57	78.70
	Ours	81.60	78.01	**79.76**	80.09	81.13	**80.61**
Rest 16	$\text{JET}^{o}_{M=6}$	83.33	68.58	75.24	89.44	80.21	84.57
	GTS	82.69	85.62	84.13	83.37	86.53	84.92
	Ours	84.20	86.06	**85.12**	84.62	88.00	**86.28**

Table 7: Test results on the ATE and OTE tasks with sub-optimal evaluation method. Our method and GTS (Wu et al., 2020) allow for ATE and OTE tasks to be predicted independently from the ASTE task. However, $\text{JET}^{o}_{M=6}$ (Xu et al., 2020b) does not. Hence, we make another comparison here by extracting all opinion and target spans from the ASTE predictions.

Dataset	Model	ATE			OTE		
		P.	*R.*	F_1	*P.*	*R.*	F_1
Rest 14	GTS	78.50	87.38	82.70	82.07	88.99	85.39
	RACL	79.90	87.74	83.63	80.26	87.99	83.94
	Ours	83.56	87.59	**85.50**	82.93	89.67	**86.16**
Lap 14	GTS	78.63	81.86	80.21	76.27	79.32	77.77
	RACL	78.11	81.99	79.99	75.12	79.92	77.43
	Ours	81.48	86.39	**83.86**	83.00	82.28	**82.63**
Rest 15	GTS	74.95	82.41	78.50	74.75	81.56	78.01
	RACL	75.22	81.94	78.43	76.41	82.56	79.35
	Ours	78.97	84.68	**81.72**	77.36	84.86	**80.93**
Rest 16	GTS	75.05	89.16	81.50	78.36	88.42	83.09
	RACL	74.12	89.20	80.95	79.25	89.77	84.17
	Ours	79.78	88.50	**83.89**	82.59	90.91	**86.54**

Table 8: Additional comparison of test results on the ATE and OTE tasks. Note that RACL (Chen and Qian, 2020) does not consider supervision from target-opinion pairs, but it includes the sentiment polarities on the target terms.

As mentioned in Table 3, it is challenging to make a fair comparison between the previous ABSA framework RACL (Chen and Qian, 2020), which also address the ATE and OTE tasks while solving other ABSA subtasks, and our approach as well as the GTS (Wu et al., 2020). This is because the mentioned approaches have different task settings. The RACL considers the sentiment polarity on the target terms when solving the ATE and OTE tasks, but GTS and our method both consider the pairing relation between target and opinion terms. However, for reference, Table 8 shows the compar-

Dataset	**Rest 14**		**Lap 14**		**Rest 15**		**Rest 16**	
	# Target	# Opinion	# Target	# Opinion	# Target	# Opinion	# Target	# Opinion
Train	2051	2086	1281	1268	862	941	1198	1307
Dev	500	503	296	304	213	236	296	319
Test	848	854	463	474	432	461	452	475

Table 9: Additional statistics. # Target denotes the number of target terms. # Opinion denotes the numbers of opinion terms.

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
Ours (BiLSTM)	66.76	53.90	59.61	60.78	49.37	54.45	69.13	60.08	64.26	71.59	61.95	66.41
Ours (BERT)	68.05	65.65	66.80	63.35	58.90	61.02	70.16	71.41	70.75	72.52	71.92	72.19

Table 10: Results on the development datasets.

isons of the three methods on the ATE and OTE tasks on the datasets released by Xu et al. (2020b).