Eliminating Sentiment Bias for Aspect-Level Sentiment Classification with Unsupervised Opinion Extraction

Bo Wang\textsuperscript{1}, Tao Shen\textsuperscript{2}, Guodong Long\textsuperscript{2}, Tianyi Zhou\textsuperscript{3,4}, Yi Chang\textsuperscript{1,5,6,7}\textsuperscript{8}
\textsuperscript{1}School of Artificial Intelligence, Jilin University
\textsuperscript{2}Australian AI Institute, School of CS, FEIT, University of Technology Sydney
\textsuperscript{3}University of Washington, Seattle; \textsuperscript{4}University of Maryland, College Park
\textsuperscript{5}International Center of Future Science, Jilin University

bowang19@mails.jlu.edu.cn,\{tao.shen,guodong.long\}@uts.edu.au, tianyizh@uw.edu,yichang@jlu.edu.cn

Abstract

Aspect-level sentiment classification (ALSC) aims at identifying the sentiment polarity of a specified aspect in a sentence. ALSC is a practical setting in aspect-based sentiment analysis due to no opinion term labeling needed, but it fails to interpret why a sentiment polarity is derived for the aspect. To address this problem, recent works fine-tune pre-trained Transformer encoders for ALSC to extract an aspect-centric dependency tree that can locate the opinion words. However, the induced opinion words only provide an intuitive cue far below human-level interpretability. Besides, the pre-trained encoder tends to internalize an aspect’s intrinsic sentiment, causing sentiment bias and thus affecting model performance. In this paper, we propose a span-based anti-bias aspect representation learning framework. It first eliminates the sentiment bias in the aspect embedding by adversarial learning against aspects’ prior sentiment. Then, it aligns the distilled opinion candidates with the aspect by span-based dependency modeling to highlight the interpretable opinion terms. Our method achieves new state-of-the-art performance on five benchmarks, with the capability of unsupervised opinion extraction.

1 Introduction

Aspect-based sentiment analysis (ABSA) (Jiang et al., 2011) aims to determine sentiment polarity w.r.t. a specified aspect term in a piece of text. For example, in “The food is tasty, but the service is terrible”, the sentiment towards aspect term (AT) “food” is positive according to the opinion term (OT) “tasty”, while the sentiment towards “service” is negative according to “terrible”. The most comprehensive setting of ABSA is aspect sentiment triplet extraction (ASTE) (Peng et al., 2020) consisting of a series of subtasks, i.e., aspect extraction, aspect-level sentiment classification and opinion extraction. Thereby, given a piece of text, ASTE can produce a set of triples, i.e., (Aspect Term, Sentiment, Opinion Term), to describe sentiment with details of What, How and Why, so it enjoys full interpretability. Continuing the above example, ASTE can generate triples like (food, Positive, tasty). However, the human annotation on opinion terms is much more label-intensive than traditional sentiment analysis task.

Therefore, by following many recent works, we target the practical subtask of ASTE, called aspect-level sentiment classification (ALSC). It predicts a three-categorical sentiment (i.e., positive, neutral or negative) of a given aspect term in a sentence. Most recent works capture the modification relation between aspect and opinion terms in an implicit manner, which is usually achieved by integrating graph neural networks (GNNs) over dependency parsing tree into text representation learning (Zhang et al., 2019a; Sun et al., 2019b; Tang et al., 2020; Wang et al., 2020). Further, the performance can be significantly boosted when incorporating pre-trained Transformer encoder, e.g., BERT (Devlin et al., 2019) in a fine-tuning paradigm (Sun et al., 2019a; Tang et al., 2020; Wang et al., 2020; Chen et al., 2020). Although these works achieve excellent results even close to humans, they cannot derive the interpretability to explain why an aspect is associated with the polarity prediction.

Luckily, pre-trained Transformer encoder can also be used to explain linguistic knowledge underlying the given text via dependency probing (Clark et al., 2019; Wu et al., 2020). This has been exploited by Dai et al. (2021) to reveal that, after fine-tuning the pre-trained encoder on ALSC, an aspect-centric dependency tree can be induced to highlight the modifier of an aspect. Intuitively, the highlighted modifier is viewed as an opinion word of the corresponding aspect, which thus, to some extent, brings the interpretability back.

Nevertheless, compared with the span-formatted opinion terms in ASTE, the opinion word illustrated by the induced dependency tree can only pro-
vide an intuitive, noisy, word-level sentiment cue but is far from the human-level interpretability as in opinion extraction. What’s worse, as verified by Huang et al. (2020), the Transformer pre-trained on large-scale raw corpora tends to internalize terms’ intrinsic attributes, so it causes the problem of sentiment bias when generating text given a particular term prompt. In ALSC scenario, we found that sentiment bias also exists and affects a model to determine the sentiment of an aspect term regardless of its contextual information (e.g., opinion terms). The bias is especially obvious for the aspects that can imply strong sentiment themselves. For example, for “There’s candlelight and music”, a model based on pre-trained Transformer is likely to misclassify the sentiment towards “music” as positive, whereas the oracle label is neutral.

In this work, based on the pre-trained Transformer encoder to ensure state-of-the-art ALSC performance, we aim to eliminate sentiment bias in the ALSC scenario while equipping the model with human-level aspect-opinion interpretability.

To this end, we propose a Span-based Anti-bias aspect Representation Learning (SARL) framework for ALSC with unsupervised opinion extraction. First, instead of widely feeding a concatenation of sentence and aspect into a pre-trained Transformer, we adopt a span-level paradigm (Hu et al., 2019; Zhao et al., 2020) which focuses on deriving span representation of an aspect term. Then, we propose an anti-bias aspect encoding module to eliminate the sentiment bias existing in aspect representations, which is achieved by an adversarial learning against the aspect’s prior sentiment indicated in SentiWordNet (Esuli and Sebastiani, 2006). Next, built upon sentiment-agnostic aspect representation from the above encoder, we propose an aspect-opinion dependency alignment module to capture explicit modifications from opinion term candidates to the targeted aspect, and integrate the modifications into aspect representation via gating. The integrated representation is lastly passed into a neural classifier for sentiment prediction.

For pairwise aspect-opinion alignment via a span-based model, we work share a high-level inspiration with SpanMlt (Zhao et al., 2020) but differs in that, SpanMlt targets fully-supervised aspect and opinion terms extraction whereas ours leverages the alignment to empower aspect-opinion interpretability even without opinion supervisions.

Our main contributions are as follows:

- We propose a span-based aspect encoding module to alleviate the sentiment bias problem from which pre-trained Transformers suffer.
- By a dependency aligner, our model can derive human-level aspect-opinion interpretability w/o opinion extraction supervisions.
- We achieve new state-of-the-art results on 5 ALSC datasets with extensive analyses and present metrics to measure the interpretability.

2 Methodology

This section begins with a task definition of ALSC. Then, we present our Span-based Anti-bias aspect Representation Learning (SARL) framework (as in Figure 1) consisting of an adversarial anti-bias aspect encoder (§2.1) and a distilled aspect-opinion dependency aligner (§2.2). Lastly, we detail the training and inference of the proposed model (§2.3), including unsupervised opinion extraction.

Task Definition. Given a sentence with \( n \) words, \( x = [x_1, \ldots, x_n] \), ALSC aims to predict a three-categorical sentiment from \{Positive, Neutral, Negative\} of an aspect term, \( a \), where \( a \) is a span of \( x \), from \( s^a \) and to \( e^a \), i.e., \( a = [s^a, e^a] \).

2.1 Adversarial Anti-bias Aspect Encoder

As suggested by Devlin et al. (2019) and verified by many works, a common practice to tackle pair inputs is feeding the pre-trained Transformer with a concatenation of them. Hence, many ALSC works (Jiang et al., 2020; Hou et al., 2021) feed a concatenation of the sentence \( x \) and an aspect \( a \) into the encoder. Nonetheless, considering that every aspect is a word span of the sentence, it is natural to employ a span-level paradigm for aspect representation, which has been proven as effective as the concatenation models (Zhou et al., 2019; Zhao et al., 2020). Also, a span-level model is more efficient as it predicts the sentiment for all aspect terms in a single feed-forward process.

Span-level Aspect Representation. We thereby adopt the span-level paradigm to generate aspect representation. Formally, a Transformer is fed with a sentence without additional aspect terms,

\[
H = \text{Transformer-Enc}(x; \theta^{ptm}) \tag{1}
\]

where, \( H = [h_1, \ldots, h_n] \in \mathbb{R}^{d \times n} \) denotes contextualized representations corresponding to all the words. Then, given span \([s^a, e^a]\) of each aspect term \( a \), we derive its representation by

\[
e^a = [h^a_s; \text{Attn-Pool}(H_{s^a:e^a}; \theta^{aa})], \tag{2}
\]
where \([;\) is vector concatenation, \(\text{Attn-Pool}(\cdot)\) is attention pooling to generate sequence-level embedding with multi-layer perceptron (MLP)-derived weights (Lin et al., 2017; Liu et al., 2016).

And \(c^a \in \mathbb{R}^{3d}\) is the resulting span-level aspect representation. In the remainder, we omit the aspect indicator, \(a\), if no confusion caused. But, as briefed in §1, sentiment bias will occur since the Transformer encoders pre-trained on large-scale text corpora incline to internalize the sentiment polarity of aspect terms. Hence, directly applying a classifier to the aspect representation \(c\) for its sentiment prediction will be affected.

**Adversarial Anti-Bias Module.** A promising way is leveraging adversarial learning to eliminate the sentiment bias of a particular aspect. Open questions still remain about how to define the sentiment bias as discriminator’s objective and how to fool the discriminator for our anti-bias purpose. As an answer to the first answer, we resort to external sentiment knowledge, SentiWordNet (Esuli and Sebastiani, 2006), for its prior three-categorical sentiment for common lexicons and phrases. Given an aspect \(a\) from a sentence, we can easily obtain its prior sentiment polarity \(y^{(pr)}\) by querying SentiWordNet. The prior sentiment polarity is the intrinsic attribute of an aspect, which can thus be viewed as its sentiment bias. Hence, we train a sentiment bias discriminator towards \(y^{(pr)}\) to exploit bias information underlying the aspect representation. Formally, we present an MLP-based neural classifier upon \(c\) as the discriminator, i.e.,

\[
p^{(pr)} = \text{softmax}(\text{MLP}(c^a; \theta^{dis})) \in \mathbb{R}^3. \quad (3)
\]

Next, training loss of this discriminator is

\[
\mathcal{L}_{\theta^{dis}} = -\sum_D \sum_A \log p^{(pr)}[\hat{y} = y^{(pr)}], \quad (4)
\]

where \(D\) denotes ALSC dataset with sentence-level samples, where \(A\) denotes all aspects in a sentence, and \(p^{(pr)}[\hat{y} = y^{(pr)}]\) denotes fetching the probability value corresponding to prior sentiment \(y^{(pr)}\).

On the other side, to eliminate sentiment bias, the span-level aspect encoder aims to fool the discriminator towards generating neutral-sentiment representation. The adversarial loss is written as

\[
\mathcal{L}_{\theta^{adv}} = -\sum_D \sum_A \log p^{(pr)}[\hat{y} = \text{Neutral}] \quad (5)
\]

With the adversarial learning between \(\mathcal{L}^{(dis)}\) and \(\mathcal{L}^{(adv)}\), the aspect representation \(c\) can escape from sentiment bias and become sentiment-agnostic.

**2.2 Aspect-Opinion Dependency Aligner**

As opinion term is also a span of words, it is intuitive to use a span-based alignment model (Lee et al., 2017) for explicit modification from opinion...
terms to the aspect. This model is first proposed for co-reference resolution to align an entity mention with its antecedent, but usually requires full supervision. For example, it has been adapted to extract pair-wise aspect and opinion terms (Zhao et al., 2020) in ABSA, and the supervisions include both opinion terms and aspect-opinion associations but none of them is available in ALSC.

Fortunately, built upon the sentiment-agnostic aspect representation derived from the above encoder, the span-based alignment model is likely to automatically learn the dependency between opinion and aspect spans, which is guided by the aspect’s sentiment label. To this end, we present an aspect-opinion dependency aligner w/o supervisions.

**Opinion Term Candidates.** Similar with Lee et al. (2017) and Zhao et al. (2020), we first enumerate all the possible spans as candidates of opinion terms. Given the sentence $x = [x_1, \ldots, x_n]$, an opinion term candidate $o$ is a span of words from $s_o$ to $e_o$, i.e., $o = [x_{s_o}, e_o]$. A set of all the opinion term candidates can be written as

$$\mathcal{O} = \{x_{s_o,e_o} | \sum_{1 \leq s_o \leq e_o \leq n} \wedge (e_o - s_o \leq l) \wedge (e_o < s_o \lor s_o > e_o)\},$$

(6)

where the last condition means no overlap between an arbitrary oracle aspect term and its opinion candidates, and $m := |\mathcal{O}|$. Then, we can easily derive the representation of each candidate $o \in \mathcal{O}$ given the contextualized embedding $H$ by the encoder in Eq.(1). We adopt the same scheme to generate span-level opinion embedding as in Eq.(2), i.e.,

$$u^o = [h_o; h_o; \text{Attn-Pool}(H_{s_o+e_o}; \theta(o))],$$

(7)

where the resulting vector $u^o \in \mathbb{R}^d$ represents the candidate $o$. Thus, we can get a set of candidate representations, $U = \{u^o\}_{o \in \mathcal{O}} \in \mathbb{R}^{d \times m}$.

Following previous models (Lee et al., 2017; Zhao et al., 2020), two kinds of scores are generated for the dependency between opinions and aspects: (1) a sentiment mention score determines if a span candidate is an opinion mention; and (2) an aspect-opinion alignment score estimates the modification relation in an aspect-opinion pair.

**Distilled Sentiment Mention Scoring.** Most previous span-based alignment models employ an one-dim-out neural module (e.g., MLP with Sigmoid) to determine the confidence of a mention but require full supervision for accurate predictions. Considering such supervision is unavailable in ALSC, we consequently weaken the supervision from “whether a mention is a gold opinion term” to “whether a mention expresses a sentiment polarity”. The weak supervision can be readily obtained from a well-trained phrase-level sentiment classification model via distilling model (Hinton et al., 2015). Hence, we first employ an MLP-based classifier built upon an opinion term candidate $u^o$ to derive a three-categorical sentiment distribution,

$$p^{(ms)}_o = \text{softmax}(\text{MLP}(u^o; \theta^{(ms)})) \in \mathbb{R}^3;$$

(8)

and a phrase-level sentiment classification model is

$$p^{(pl)}_o = \text{Senti-Model}(o; \theta^{(pl)}),$$

(9)

which is also based on a pre-trained Transformer and trained on a popular phrase-level sentiment analysis dataset, Stanford Sentiment Treebank (Socher et al., 2013). Then, we can define a soft loss of sentiment distillation based on Kullback–Leibler (KL) divergence, i.e.,

$$\mathcal{L}^{(kl)} = \text{KL}(p^{(pl)}_o || p^{(ms)}_o).$$

(10)

Last, we use the non-neutral probability as the confidence of a mention expressing sentiment polarity, i.e., the sentiment mention score,

$$r^o := 1 - p^{(ms)}_o[y = \text{Neutral}], \forall o \in \mathcal{O},$$

(11)

$$r = \{r^o\}_{o \in \mathcal{O}} \in \mathbb{R}^m.$$ 

(12)

**Aspect-Opinion Dependency Modeling.** Besides the mention score, we use an MLP to estimate the alignment between the aspect and opinion candidates. First, we obtain a relationship representation by an interactive concatenation (Reimers and Gurevych, 2019) of their span-level embeddings,

$$q^o = [c; u^o; c \odot u^o; z^o], \forall o \in \mathcal{O},$$

(13)

where “$\odot$” is Hadamard product and $z$ is a learnable relative-position embedding indicating their distance over the syntactic dependency parsing tree of the sentence. Again, please note we omitted the superscript $a$ in the equation for simplification. Next, $q^o$ is passed into an MLP-based scorer to calculate pairwise alignment score, i.e.,

$$\hat{f}^o = \text{MLP}(q^o; \theta^{(as)}) \in \mathbb{R}, \forall o \in \mathcal{O},$$

(14)

$$\hat{f} = \{\hat{f}^o\}_{o \in \mathcal{O}} \in \mathbb{R}^m.$$ 

(15)

Then, we apply softmax to $\hat{f}$ for normalized alignment scores, which are subsequently weighted by the corresponding sentiment mention scores $r$ in Eq.(12) to derive the dependency scores between the aspect and all opinion candidates, i.e.,

$$f = \text{softmax}(\hat{f}) \odot r.$$ 

(16)
## Model Training and Inference

### Aspect-level Sentiment Classification.

On top of $v$, we define a neural classifier for the final three-categorical sentiment prediction as

$$p^{(sc)} = \text{softmax}(\text{MLP}(v; \theta^{(sc)})) \in \mathbb{R}^3. \quad (22)$$

Table 1: Summary statistics of five benchmark datasets.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Positive</th>
<th>Neutral</th>
<th>Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Train</td>
<td>Test</td>
<td>Train</td>
</tr>
<tr>
<td>Laptop14</td>
<td>994</td>
<td>341</td>
<td>464</td>
</tr>
<tr>
<td>Rest14</td>
<td>2164</td>
<td>728</td>
<td>637</td>
</tr>
<tr>
<td>Rest15</td>
<td>912</td>
<td>326</td>
<td>36</td>
</tr>
<tr>
<td>Rest16</td>
<td>1240</td>
<td>469</td>
<td>69</td>
</tr>
<tr>
<td>Twitter</td>
<td>1561</td>
<td>173</td>
<td>3127</td>
</tr>
</tbody>
</table>

Table 2: Summary statistics of unsupervised opinion extraction test set. $\#a^{neu/\alpha}$ is the number of neutral-sentiment aspects without corresponding opinion terms.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#AT</th>
<th>#OT</th>
<th>#0</th>
<th>#a^{neu/\alpha}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop14</td>
<td>444</td>
<td>527</td>
<td>112</td>
<td>128</td>
</tr>
<tr>
<td>Rest14</td>
<td>832</td>
<td>974</td>
<td>137</td>
<td>128</td>
</tr>
</tbody>
</table>

Table 3: Settings of learning rate. “others” includes $\theta^{(ane)}$, $\theta^{(bae)}$, $\theta^{(mne)}$, $\theta^{(oe)}$ and $\theta^{(se)}$.

<table>
<thead>
<tr>
<th>LR ($\times 10^{-3}$)</th>
<th>size</th>
<th>Laptop14</th>
<th>Rest14</th>
<th>Rest15</th>
<th>Rest16</th>
<th>Twitter</th>
</tr>
</thead>
<tbody>
<tr>
<td>lr-\theta^{(pm)}</td>
<td>base</td>
<td>1.3</td>
<td>2.2</td>
<td>2.2</td>
<td>2.2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>large</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>lr-others</td>
<td>base</td>
<td>20</td>
<td>1</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>large</td>
<td>20</td>
<td>1</td>
<td>1</td>
<td>20</td>
<td>1</td>
</tr>
</tbody>
</table>

And the training loss of ALSC task is written as

$$L^{(sc)} = - \sum_D \sum_{A} \log p^{(sc)}_{y^{(se)}}, \quad (23)$$

where $y^{(sc)}$ denotes the oracle label of an aspect.

### Training and Inference.

Besides the discriminator loss $L^{(dis)}$ in Eq.(4), we train the learnable parameters in our proposed SARL model towards a linear combination of the other three losses, i.e.,

$$L^{(alsc)} = L^{(sc)} + \beta L^{(dis)} + \gamma L^{(kl)}. \quad (24)$$

We also set a hyper-parameter $\alpha$ to control the proportion of discriminator learning (i.e., $L^{(dis)}$) against the ALSC model learning (i.e., $L^{(alsc)}$). The inference procedure can be simply written as

$$y^* = \arg\max_{y} p^{(sc)}. \quad (25)$$

### Unsupervised Opinion Extraction.

A well-trained SARL model is equipped with the capability to extract opinion term(s) for an aspect, based on its intermediate variable, i.e.,

$$o^* = \arg\max_{o, \text{dummy}} p^{(sc)}.$$ \quad (26)

where $f \in \mathbb{R}^{m+1}$ from Eq.(16) denotes aspect-opinion dependency scores, including the last dim for dummy opinion, i.e., an aspect w/o any opinion in the sentence. Thereby, $o^*$ is the extracted opinion term for a specified aspect term in the sentence, and it is worth mentioning again that the opinion extraction is learned in an unsupervised manner.

## 3 Experiment

### Datasets.

For ALSC task, we evaluate our model on five datasets\(^1\), whose statistics are listed in Table 1, including (i) Laptop14 (SemEval-2014T4) and (ii) Rest14 (SemEval-2014T4).

\(^1\)The source code and datasets are available at [https://github.com/wangbo9719/SARL_ABSA](https://github.com/wangbo9719/SARL_ABSA)
(Pontiki et al., 2014) with laptop reviews, (ii) Rest14 (SemEval-2014T4), Rest15 (SemEval-2015T12) and Rest16 (SemEval-2016T5) (Pontiki et al., 2014, 2015, 2016) with restaurant reviews, and (iii) Twitter (Mitchell et al., 2013) with tweets. Following most competitors including the models compared in Table 4, we do not split training set.

To evaluate unsupervised opinion extraction, we employ the test set from Xu et al. (2020) where annotations of opinion terms\(^2\) are from Fan et al. (2019). The statistics are listed in Table 2.

**Training Setups.** We use a mini-batch Stochastic Gradient Descent (SGD) to minimize the loss functions, with Adam optimizer, 10% warm-up, and a linear decay of the learning rate. To initialize the Transformer, we alternate between pre-trained RoBERTa\(_{\text{base}}\) and large. We set batch size = 16 and max sequence length = 64 based on experience, and conduct grid searches for the other hyperparameters. Then, we set the max width of candidate span \(l = 15\), the number of training epochs = 7 for Twitter and 10 for other datasets, the training proportion \(\alpha = 1/3\) for Laptop14 and Twitter, 0.2 for Rest14 and Rest15, 0.1 for Rest16, the loss weight in Eq.(24) \(\beta = 0.05\) for Rest15 and Twitter and 0.1 for the remains, \(\gamma = 1\), and \(\delta = 1/m\) in Eq.(17). The learning rates are listed in Table 3.

### 3.1 Overall Performance

ALSC results of competitive approaches and our SARL on the five benchmarks are shown in Table 4. Following prior works, we adopt accuracy (Accu) and macro-F1 (Ma-F1) to evaluate the performance, and the results of SARL are the best values from ten runs. It is observed that our proposed SARL achieves state-of-the-art performance on all these datasets. Compared to static embedding-based methods, the methods with pre-trained Transformer gain better results. In the same embedding genre RoBERTa\(_{\text{base}}\), SARL outperforms others by an average of 1% on accuracy and 1.3% on macro-F1. Furthermore, the RoBERTa\(_{\text{large}}\)-based SARL derives more significant progress.

### 3.2 Sentiment Bias Elimination

**Bias Statistics.** As in Figure 2 (left), the sentiment bias is common since aspects with bias polarity account for nearly 50% in most datasets.

**The Success of Adversarial Learning.** To explore the effectiveness of our adversarial learning, we first train a model without adversarial, “w/o adv” for short (please refer to below ablation study for more). Then, based on this trained model, we add a sentiment bias detector that has the same architecture as the discriminator to detect the bias existing in aspects representations. Here, we use the ratio of neutral predictions to measure the accuracy of detector since the smaller ratio means more bias detected.

\[\frac{|Q|}{|S|}, \text{ where } Q = \{a | a \in S \land y^{(pr)} = y^{(sc)} \}, \quad |S| = \{|a | a \in S \land y^{(pr)} \neq y^{(sc)} \}, \text{ and the number on the top of bin is } |Q|.\]

\[\text{Accu Ma-F1 Accu Ma-F1 Accu Ma-F1 Accu Ma-F1 Accu Ma-F1}
Laptop14 Rest14 Rest15 Rest16 Twitter
Baseline SARL
8 9
24 14
18
20
6
4
12
8

Table 4: ALSC results on the five datasets. \(\dagger\) Numbers are from Dai et al. (2021), and others are from the original papers, i.e., ASGCN (Zhang et al., 2019a), CDT (Sun et al., 2019b), BiGCN (Zhang and Qian, 2020), DGEDT (Tang et al., 2020), RGAT (Wang et al., 2020), kumaGCN (Chen et al., 2020), PWCN-FT-RoBERTa (Zhang et al., 2019b).
Table 5: Unsupervised opinion extraction results (%) on Laptop14/Rest14. The resulting numbers of SpanMlt (Zhao et al., 2020) and Peng et al. (2020) are from their original papers which the metrics under @1 are the standard metrics. The results of induced tree (Dai et al., 2021) are calculated by regarding the aspects’ sub-nodes in the tree as its extracted opinions.

Table 6: The Hits@N results about dummy opinions.

Table 7: Ablation results of ALSC. The full model denotes our proposed SARL. “w/o adv” removes anti-bias module in §2.1. “w/o AT-OT aligner” removes the proposed module in §2.2. “w/o all” degrades our SARL model to Transformer-based span-level ALSC (i.e., classification based on Eq.(2)).
<table>
<thead>
<tr>
<th>Index</th>
<th>Example</th>
<th>“w/o all”</th>
<th>SARL</th>
<th>Top-3 candidates</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>After dinner I heard [music], neu playing and discovered that there is a lounge downstairs.</td>
<td>pos</td>
<td>neu</td>
<td>dummy; playing; playing and discovered</td>
</tr>
<tr>
<td>2</td>
<td>Desserts include [flan], neu and sopaipillas.</td>
<td>pos</td>
<td>neu</td>
<td>dummy; and; include</td>
</tr>
<tr>
<td>3</td>
<td>How is this [place] still open?</td>
<td>pos</td>
<td>neg</td>
<td>open; still open; dummy</td>
</tr>
<tr>
<td>4</td>
<td>This place has beautiful [sushi], and it’s delicious CHEAP.</td>
<td>pos</td>
<td>pos</td>
<td>delicious; delicious CHEAP; beautiful</td>
</tr>
<tr>
<td>5</td>
<td>The wait [staff], was loud and inconsiderate.</td>
<td>neg</td>
<td>neg</td>
<td>loud; was loud; loud and inconsiderate</td>
</tr>
<tr>
<td>6</td>
<td>The sauce is excellent (very fresh) with [dabs of real mozzarella].</td>
<td>pos</td>
<td>pos</td>
<td>excellent; excellent (very fresh); fresh</td>
</tr>
<tr>
<td>7</td>
<td>15% gratuity automatically added to the [bill].</td>
<td>neu</td>
<td>neu</td>
<td>dummy; automatically; 15</td>
</tr>
<tr>
<td>8</td>
<td>The [quality of the meat], neu was on par with your local grocery store.</td>
<td>pos</td>
<td>neu</td>
<td>dummy; par; was on par</td>
</tr>
</tbody>
</table>

Table 8: Case study for sentiment bias (row 1-3), unsupervised opinion extraction (row 4-5) and error analysis (6-8).

### 3.5 Case Study

**Does SARL eliminate sentiment bias?**  As the top-3 rows in Table 8, those aspects with intrinsic positive bias are mis-classified by our span-level baseline but correctly classified by SARL, which verifies the effectiveness of SARL.

**How does SARL obtain interpretability from unsupervised opinion extraction?**  As listed in Table 8, SARL can exactly extract the opinion terms of the targeted aspect so provide promising interpretability for sentiment prediction under the unsupervised opinion setting. In addition, for the neutral aspect without explicit opinion terms in a sentence, the dummy opinion always ranks first like the 2nd row in Table 8, which explains the reason for neutral prediction.

**Error Analysis.**  To analyze the limitation of ALSC models including ours, we investigate all the examples mis-classified by SARL on Rest14, and summarize two main problems: (1) The major (up to 66%) problem is neutral-related mis-classifying because a neutral aspect term is affected by the polarity words associated with other aspects (e.g., 6rd row). The other problem (23%) refers to that it is infeasible to determine aspects’ sentiment without commonsense knowledge (e.g., 7th row) or additional information (e.g., 8th row). In summary, how to accurately classify the aspect without explicit opinion is still an open problem.

### 4 Related Work

**Aspect-Level Sentiment Classification.**  ALSC relies heavily on modification relations between aspect term and opinion term in a sentence, so recent progresses mainly fall into modeling the relations by applying graph neural networks to dependency parsing tree (Sun et al., 2019b; Zhang et al., 2019a; Wang et al., 2020). Despite their effectiveness, they lack interpretability to the sentiment prediction. For further boosted, many methods (Sun et al., 2019a; Hu et al., 2019; Chen et al., 2020; Mao et al., 2021; Dai et al., 2021) introduce the pre-trained Transformer, which also brings a little interpretability due to the highlight modification between aspect and opinion terms derived from potential syntax knowledge (Wu et al., 2020; Dai et al., 2021). However, the derived interpretability is far from human-level. Furthermore, the Transformer tends to internalize terms’ intrinsic sentiment bias, which is harmful to ALSC.

**Aspect-Opinion Alignment.**  Fan et al. (2019) first defined the aspect-oriented opinion extraction task in which the aspect terms are given in advance. Later, Zhao et al. (2020) proposed the aspect-opinion co-extraction task. More recently, Peng et al. (2020) proposed the aspect sentiment triplet extraction and further explorations have conducted by Xu et al. (2020) and Mao et al. (2021). These models extract the opinion terms for aspects and thus provide explicit interpretability. However, they require manually labeled opinions data for training, which is much more expensive than three-categorical labeling in ALSC.

### 5 Conclusion

In this work, we propose SARL framework for ALSC. Specifically, we first present an adversarial anti-bias aspect encoder to eliminate sentiment bias in aspects and then propose an aspect-opinion dependency aligner to unsupervisedly extract opinions. The experiments on 5 benchmarks can greatly support our motivations and empirical results show state-of-the-art performance with interpretability.
Acknowledgement

This work is supported by the National Natural Science Foundation of China (No.61976102, No.U19A2065).

References


