A Novel Aspect-Guided Deep Transition Model
for Aspect Based Sentiment Analysis

Yunlong Liang[1*], **Fandong Meng**[2], **Jinchao Zhang**[2], **Jinan Xu**[1†]
Yufeng Chen[1] and **Jie Zhou**[2]
[1]Beijing Jiaotong University, China
[2]Pattern Recognition Center, WeChat AI, Tencent Inc, China
{yunlonliang,jaxu,chenyf}@bjtu.edu.cn
{fandongmeng,dayerzhang,withtomzhou}@tencent.com

Abstract

Aspect based sentiment analysis (ABSA) aims to identify the sentiment polarity towards the given aspect in a sentence, while previous models typically exploit an aspect-independent (weakly associative) encoder for sentence representation generation. In this paper, we propose a novel **A**spect-**G**uided **D**eep **T**ransition model, named AGDT, which utilizes the given aspect to guide the sentence encoding from scratch with the specially-designed deep transition architecture. Furthermore, an aspect-oriented objective is designed to enforce AGDT to reconstruct the given aspect with the generated sentence representation. In doing so, our AGDT can accurately generate aspect-specific sentence representation, and thus conduct more accurate sentiment predictions. Experimental results on multiple SemEval datasets demonstrate the effectiveness of our proposed approach, which significantly outperforms the best reported results with the same setting.[1]

1 Introduction

Aspect based sentiment analysis (ABSA) is a fine-grained task in sentiment analysis, which can provide important sentiment information for other natural language processing (NLP) tasks. There are two different subtasks in ABSA, namely, aspect-category sentiment analysis and aspect-term sentiment analysis (Pontiki et al., 2014; Xue and Li, 2018). Aspect-category sentiment analysis aims at predicting the sentiment polarity towards the given aspect, which is in predefined several categories and it may not appear in the sentence. For instance, in Table 1, the aspect-category sentiment analysis is going to predict the sentiment polarity towards the aspect "*food*", which

Sentence	The appetizers are ok, but the service is slow.	
Aspect-Category	food	service
Aspect-Term	The appetizers	service
Sentiment Polarity	Neutral	Negative

Table 1: The instance contains different sentiment polarities towards two aspects.

is not appeared in the sentence. By contrast, the goal of aspect-term sentiment analysis is to predict the sentiment polarity over the aspect term which is a subsequence of the sentence. For instance, the aspect-term sentiment analysis will predict the sentiment polarity towards the aspect term "*The appetizers*", which is a subsequence of the sentence. Additionally, the number of categories of the aspect term is more than one thousand in the training corpus.

As shown in Table 1, sentiment polarity may be different when different aspects are considered. Thus, the given aspect (term) is crucial to ABSA tasks (Jiang et al., 2011; Ma et al., 2017; Wang et al., 2018; Xing et al., 2019; Liang et al., 2019). Besides, Li et al. (2018a) show that not all words of a sentence are useful for the sentiment prediction towards a given aspect (term). For instance, when the given aspect is the "*service*", the words "*appetizers*" and "*ok*" are irrelevant for the sentiment prediction. Therefore, an aspect-independent (weakly associative) encoder may encode such background words (e.g., "*appetizers*" and "*ok*") into the final representation, which may lead to an incorrect prediction.

Numerous existing models (Tang et al., 2016b; Tay et al., 2017; Fan et al., 2018; Xue and Li, 2018) typically utilize an aspect-independent encoder to generate the sentence representation, and then apply the attention mechanism (Luong et al., 2015) or gating mechanism to conduct feature

---

[1]The code is publicly available at: https://github.com/XL2248/AGDT

selection and extraction, while feature selection and extraction may base on noised representations. In addition, some models (Tang et al., 2016a; Wang et al., 2016; Majumder et al., 2018) simply concatenate the aspect embedding with each word embedding of the sentence, and then leverage conventional Long Short-Term Memories (LSTMs) (Hochreiter and Schmidhuber, 1997) to generate the sentence representation. However, it is insufficient to exploit the given aspect and conduct potentially complex feature selection and extraction.

To address this issue, we investigate a novel architecture to enhance the capability of feature selection and extraction with the guidance of the given aspect from scratch. Based on the deep transition Gated Recurrent Unit (GRU) (Cho et al., 2014; Pascanu et al., 2014; Miceli Barone et al., 2017; Meng and Zhang, 2019), an aspect-guided GRU encoder is thus proposed, which utilizes the given aspect to guide the sentence encoding procedure at the very beginning stage. In particular, we specially design an aspect-gate for the deep transition GRU to control the information flow of each token input, with the aim of guiding feature selection and extraction from scratch, i.e. sentence representation generation. Furthermore, we design an aspect-oriented objective to enforce our model to reconstruct the given aspect, with the sentence representation generated by the aspect-guided encoder. We name this **A**spect-**G**uided **D**eep **T**ransition model as AGDT. With all the above contributions, our AGDT can accurately generate an aspect-specific representation for a sentence, and thus conduct more accurate sentiment predictions towards the given aspect.

We evaluate the AGDT on multiple datasets of two subtasks in ABSA. Experimental results demonstrate the effectiveness of our proposed approach. And the AGDT significantly surpasses existing models with the same setting and achieves state-of-the-art performance among the models without using additional features (e.g., BERT (Devlin et al., 2018)). Moreover, we also provide empirical and visualization analysis to reveal the advantages of our model. Our contributions can be summarized as follows:

- We propose an aspect-guided encoder, which utilizes the given aspect to guide the encoding of a sentence from scratch, in order to conduct the aspect-specific feature selection

and extraction at the very beginning stage.

- We propose an aspect-reconstruction approach to further guarantee that the aspect-specific information has been fully embedded into the sentence representation.

- Our AGDT substantially outperforms previous systems with the same setting, and achieves state-of-the-art results on benchmark datasets compared to those models without leveraging additional features (e.g., BERT).

2 Model Description

As shown in Figure 1, the AGDT model mainly consists of three parts: *aspect-guided encoder*, *aspect-reconstruction* and *aspect concatenated embedding*. The aspect-guided encoder is specially designed to guide the encoding of a sentence from scratch for conducting the aspect-specific feature selection and extraction at the very beginning stage. The aspect-reconstruction aims to guarantee that the aspect-specific information has been fully embedded in the sentence representation for more accurate predictions. The aspect concatenated embedding part is used to concatenate the aspect embedding and the generated sentence representation so as to make the final prediction.

2.1 Aspect-Guided Encoder

The aspect-guided encoder is the core module of AGDT, which consists of two key components: Aspect-guided GRU and Transition GRU (Cho et al., 2014).

A-GRU: Aspect-guided GRU (A-GRU) is a specially-designed unit for the ABSA tasks, which is an extension of the L-GRU proposed by Meng and Zhang (2019). In particular, we design an aspect-gate to select aspect-specific representations through controlling the transformation scale of token embeddings at each time step.

At time step t, the hidden state $\mathbf{h}_t$ is computed as follows:

$$\mathbf{h}_t = (1 - \mathbf{z}_t) \odot \mathbf{h}_{t-1} + \mathbf{z}_t \odot \widetilde{\mathbf{h}}_t \qquad (1)$$

where $\odot$ represents element-wise product; $\mathbf{z}_t$ is the update gate (Cho et al., 2014); and $\widetilde{\mathbf{h}}_t$ is the candidate activation, which is computed as:

$$\begin{aligned} \widetilde{\mathbf{h}}_t = \tanh(\mathbf{g}_t \odot (\mathbf{W}_{xh}\mathbf{x}_t) + \mathbf{r}_t \odot (\mathbf{W}_{hh}\mathbf{h}_{t-1})) \\ + \mathbf{l}_t \odot \mathbf{H}_1(\mathbf{x}_t) + \mathbf{g}_t \odot \mathbf{H}_2(\mathbf{x}_t) \qquad (2) \end{aligned}$$
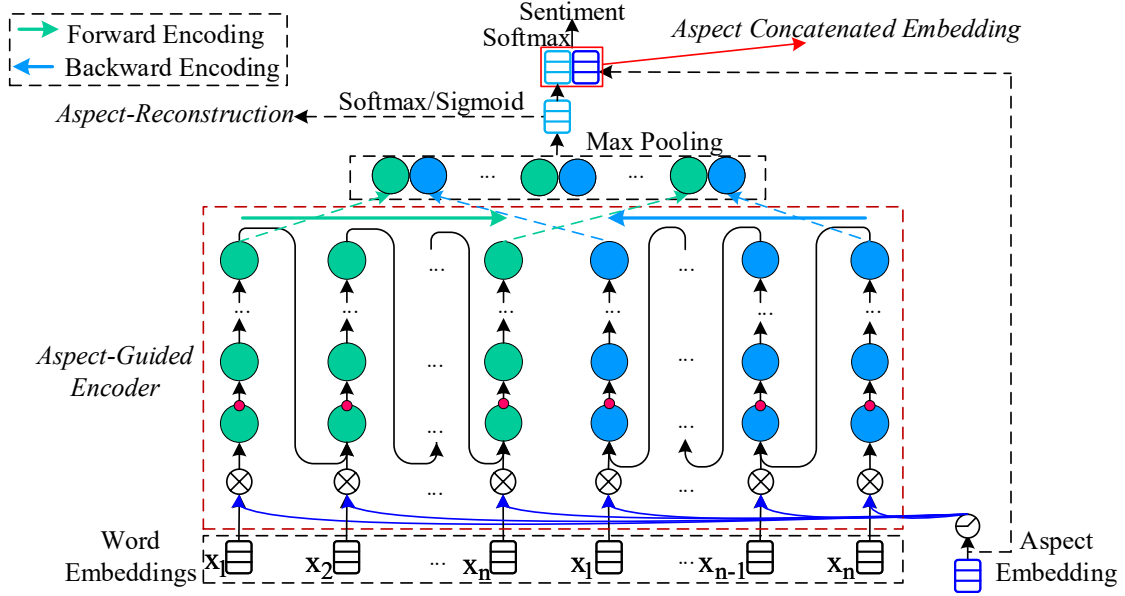
Figure 1: The overview of AGDT. The bottom right dark node (above the aspect embedding) is the aspect gate and other dark nodes ($\otimes$) means element-wise multiply for the input token and the aspect gate. The *aspect-guided encoder* consists of a L-GRU (the circle frames fused with a small circle on above) at the bottom followed by several T-GRUs (the circle frames) from bottom to up.

where $\mathbf{g}_t$ denotes the aspect-gate; $\mathbf{x}_t$ represents the input word embedding at time step t; $\mathbf{r}_t$ is the reset gate (Cho et al., 2014); $\mathbf{H}_1(\mathbf{x}_t)$ and $\mathbf{H}_2(\mathbf{x}_t)$ are the linear transformation of the input $\mathbf{x}_t$, and $\mathbf{l}_t$ is the linear transformation gate for $\mathbf{x}_t$ (Meng and Zhang, 2019). $\mathbf{r}_t$, $\mathbf{z}_t$, $\mathbf{l}_t$, $\mathbf{g}_t$, $\mathbf{H}_1(\mathbf{x}_t)$ and $\mathbf{H}_2(\mathbf{x}_t)$ are computed as:

$$\mathbf{r}_t = \sigma(\mathbf{W}_{xr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1}) \quad (3)$$

$$\mathbf{z}_t = \sigma(\mathbf{W}_{xz}\mathbf{x}_t + \mathbf{W}_{hz}\mathbf{h}_{t-1}) \quad (4)$$

$$\mathbf{l}_t = \sigma(\mathbf{W}_{xl}\mathbf{x}_t + \mathbf{W}_{hl}\mathbf{h}_{t-1}) \quad (5)$$

$$\mathbf{g}_t = \text{relu}(\mathbf{W}_a\mathbf{a} + \mathbf{W}_{hg}\mathbf{h}_{t-1}) \quad (6)$$

$$\mathbf{H}_1(\mathbf{x_t}) = \mathbf{W}_1\mathbf{x}_t \quad (7)$$

$$\mathbf{H}_2(\mathbf{x_t}) = \mathbf{W}_2\mathbf{x}_t \quad (8)$$

where "$\mathbf{a}$" denotes the embedding of the given aspect, which is the same at each time step. The update gate $\mathbf{z}_t$ and reset gate $\mathbf{r}_t$ are the same as them in the conventional GRU.

In Eq. (2) $\sim$ (8), the aspect-gate $\mathbf{g}_t$ controls both nonlinear and linear transformations of the input $\mathbf{x}_t$ under the guidance of the given aspect at each time step. Besides, we also exploit a linear transformation gate $\mathbf{l}_t$ to control the linear transformation of the input, according to the current input $\mathbf{x}_t$ and previous hidden state $\mathbf{h}_{t-1}$, which has been proved powerful in the deep transition architecture (Meng and Zhang, 2019).

As a consequence, A-GRU can control both

non-linear transformation and linear transformation for input $\mathbf{x}_t$ at each time step, with the guidance of the given aspect, i.e., A-GRU can guide the encoding of aspect-specific features and block the aspect-irrelevant information at the very beginning stage.

T-GRU: Transition GRU (T-GRU) (Pascanu et al., 2014) is a crucial component of deep transition block, which is a special case of GRU with only "state" as an input, namely its input embedding is zero embedding. As in Figure 1, a deep transition block consists of an A-GRU followed by several T-GRUs at each time step. For the current time step t, the output of one A-GRU/T-GRU is fed into the next T-GRU as the input. The output of the last T-GRU at time step t is fed into A-GRU at the time step $t+1$. For a T-GRU, each hidden state at both time step t and transition depth i is computed as:

$$\mathbf{h}_t^i = (1 - \mathbf{z}_t^i) \odot \mathbf{h}_t^{i-1} + \mathbf{z}_t^i \odot \widetilde{\mathbf{h}}_t^i \quad (9)$$

$$\widetilde{\mathbf{h}}_t^i = \tanh(\mathbf{r}_t^i \odot (\mathbf{W}_h^i\mathbf{h}_t^{i-1})) \quad (10)$$

where the update gate $\mathbf{z}_t^i$ and the reset gate $\mathbf{r}_t^i$ are computed as:

$$\mathbf{z}_t^i = \sigma(\mathbf{W}_z^i\mathbf{h}_t^{i-1}) \quad (11)$$

$$\mathbf{r}_t^i = \sigma(\mathbf{W}_r^i\mathbf{h}_t^{i-1}) \quad (12)$$

The AGDT encoder is based on deep transition cells, where each cell is composed of one A-GRU

at the bottom, followed by several T-GRUs. Such AGDT model can encode the sentence representation with the guidance of aspect information by utilizing the specially designed architecture.

2.2 Aspect-Reconstruction

We propose an aspect-reconstruction approach to guarantee the aspect-specific information has been fully embedded in the sentence representation. Particularly, we devise two objectives for two sub-tasks in ABSA respectively. In terms of aspect-category sentiment analysis datasets, there are only several predefined aspect categories. While in aspect-term sentiment analysis datasets, the number of categories of term is more than one thousand. In a real-life scenario, the number of term is infinite, while the words that make up terms are limited. Thus we design different loss-functions for these two scenarios.

For the aspect-category sentiment analysis task, we aim to reconstruct the aspect according to the aspect-specific representation. It is a multi-class problem. We take the softmax cross-entropy as the loss function:

$$\mathcal{L}_c = min(-\sum_{i=0}^{C1} y_i^c \log(p_i^c)) \qquad (13)$$

where C1 is the number of predefined aspects in the training example; y_i^c is the ground-truth and p_i^c is the estimated probability of a aspect.

For the aspect-term sentiment analysis task, we intend to reconstruct the aspect term (may consist of multiple words) according to the aspect-specific representation. It is a multi-label problem and thus the sigmoid cross-entropy is applied:

$$\mathcal{L}_t = min\{-\sum_{i=0}^{C2} [y_i^t \log(p_i^t) \\ + (1 - y_i^t) \log(1 - p_i^t)]\} \qquad (14)$$

where C2 denotes the number of words that constitute all terms in the training example, y_i^t is the ground-truth and p_i^t represents the predicted value of a word.

Our aspect-oriented objective consists of $\mathcal{L}_c$ and $\mathcal{L}_t$, which guarantee that the aspect-specific information has been fully embedded into the sentence representation.

2.3 Training Objective

The final loss function is as follows:

$$\boldsymbol{J} = min(-\sum_{i=0}^{C} y_i \log(p_i) + \boldsymbol{\lambda}\mathcal{L}) \qquad (15)$$

where the underlined part denotes the conventional loss function; C is the number of sentiment labels; y_i is the ground-truth and p_i represents the estimated probability of the sentiment label; $\mathcal{L}$ is the aspect-oriented objective, where Eq. 13 is for the aspect-category sentiment analysis task and Eq. 14 is for the aspect-term sentiment analysis task. And $\boldsymbol{\lambda}$ is the weight of $\mathcal{L}$.

As shown in Figure 1, we employ the aspect reconstruction approach to reconstruct the aspect (term), where "softmax" is for the aspect-category sentiment analysis task and "sigmoid" is for the aspect-term sentiment analysis task. Additionally, we concatenate the aspect embedding on the aspect-guided sentence representation to predict the sentiment polarity. Under that loss function (Eq. 15), the AGDT can produce aspect-specific sentence representations.

3 Experiments

3.1 Datasets and Metrics

Data Preparation. We conduct experiments on two datasets of the aspect-category based task and two datasets of the aspect-term based task. For these four datasets, we name the full dataset as "DS". In each "DS", there are some sentences like the example in Table 1, containing different sentiment labels, each of which associates with an aspect (term). For instance, Table 1 shows the customer's different attitude towards two aspects: "*food*" ("*The appetizers*") and "*service*". In order to measure whether a model can detect different sentiment polarities in one sentence towards different aspects, we extract a hard dataset from each "DS", named "HDS", in which each sentence only has different sentiment labels associated with different aspects. When processing the original sentence s that has multiple aspects $a_1, a_2, ..., a_n$ and corresponding sentiment labels $l_1, l_2, ..., l_n$ (n is the number of aspects or terms in a sentence), the sentence will be expanded into (s, a_1, l_1), (s, a_2, l_2), ..., (s, a_n, l_n) in each dataset (Ruder et al., 2016b,a; Xue and Li, 2018), i.e, there will be n duplicated sentences associated with different aspects and labels.

		Positive		Negative		Neutral		Conflict		Total	
		DS	HDS	DS	HDS	DS	HDS	DS	HDS	DS	HDS
Restaurant-14	Train	2,179	139	839	136	500	50	195	40	3,713	365
	Test	657	32	222	26	94	12	52	19	1,025	89
Restaurant-Large	Train	2,710	182	1,198	178	757	107	-	-	4,665	467
	Test	1,505	92	680	81	241	61	-	-	2,426	234

Table 2: Statistics of datasets for the aspect-category sentiment analysis task.

		Positive		Negative		Neutral		Conflict		Total		NC
		DS	HDS	DS	HDS	DS	HDS	DS	HDS	DS	HDS	DS
Restaurant	Train	2,164	379	805	323	633	293	91	43	3,693	1,038	3,602
	Test	728	92	196	62	196	83	14	8	1,134	245	1,120
Laptop	Train	987	159	866	147	460	173	45	17	2,358	496	2,313
	Test	341	31	128	25	169	49	16	3	654	108	638

Table 3: Statistics of datasets for the aspect-term sentiment analysis task. The '**NC**' indicates No "*Conflict*" label, which is just removed the "*conflict*" label and is prepared for the three-class experiment.

Aspect-Category Sentiment Analysis. For comparison, we follow Xue and Li (2018) and use the restaurant reviews dataset of SemEval 2014 ("restaurant-14") Task 4 (Pontiki et al., 2014) to evaluate our AGDT model. The dataset contains five predefined aspects and four sentiment labels. A large dataset ("restaurant-large") involves restaurant reviews of three years, i.e., 2014 $\sim$ 2016 (Pontiki et al., 2014). There are eight predefined aspects and three labels in that dataset. When creating the "restaurant-large" dataset, we follow the same procedure as in Xue and Li (2018). Statistics of datasets are shown in Table 2.

Aspect-Term Sentiment Analysis. We use the restaurant and laptop review datasets of SemEval 2014 Task 4 (Pontiki et al., 2014) to evaluate our model. Both datasets contain four sentiment labels. Meanwhile, we also conduct a three-class experiment, in order to compare with some work (Wang et al., 2016; Ma et al., 2017; Li et al., 2018a) which removed "conflict" labels. Statistics of both datasets are shown in Table 3.

Metrics. The evaluation metrics are accuracy. All instances are shown in Table 2 and Table 3. Each experiment is repeated five times. The mean and the standard deviation are reported.

3.2 Implementation Details

We use the pre-trained 300d Glove[2] embeddings (Pennington et al., 2014) to initialize word em-

---

[2]Pre-trained Glove embeddings can be obtained from http://nlp.stanford.edu/projects/glove/

beddings, which is fixed in all models. For out-of-vocabulary words, we randomly sample their embeddings by the uniform distribution $U(-0.25, 0.25)$. Following Tang et al. (2016b); Chen et al. (2017); Liu and Zhang (2017), we take the averaged word embedding as the aspect representation for multi-word aspect terms. The transition depth of deep transition model is 4 (see Section 3.4). The hidden size is set to 300. We set the dropout rate (Srivastava et al., 2014) to 0.5 for input token embeddings and 0.3 for hidden states. All models are optimized using Adam optimizer (Kingma and Ba, 2014) with gradient clipping equals to 5 (Pascanu et al., 2012). The initial learning rate is set to 0.01 and the batch size is set to 4096 at the token level. The weight of the reconstruction loss λ in Eq. 15 is fine-tuned (see Section 3.4) and respectively set to 0.4, 0.4, 0.2 and 0.5 for four datasets. The neural model is implemented in Tensorflow (Abadi et al., 2016) and all computations are done on a NVIDIA Tesla M40 GPU.

3.3 Baselines

To comprehensively evaluate our AGDT, we compare the AGDT with several competitive models.

ATAE-LSTM. It is an attention-based LSTM model. It appends the given aspect embedding with each word embedding, and then the concatenated embedding is taken as the input of LSTM. The output of LSTM is appended aspect embedding again. Furthermore, attention is applied to extract features for final predictions.

Models	Restaurant-14		Restaurant-Large	
	DS	HDS	DS	HDS
ATAE-LSTM(Wang et al., 2016)*	78.29±0.68	45.62±0.90	83.91±0.49	66.32±2.28
CNN(Kim, 2014)*	79.47±0.32	44.94±0.01	84.28±0.15	50.43±0.38
GCAE(Xue and Li, 2018)*	79.35±0.34	50.55±1.83	85.92±0.27	70.75±1.19
AGDT	**81.78**±0.31	**62.02**±1.31	**87.55**±0.17	**75.73**±0.50

Table 4: The accuracy of the aspect-category sentiment analysis task. '*' refers to citing from GCAE (Xue and Li, 2018).

Models	Restaurant		Laptop	
	DS	HDS	DS	HDS
TD-LSTM(Tang et al., 2016a)*	73.44±1.17	56.48±2.46	62.23±0.92	46.11±1.89
ATAE-LSTM(Wang et al., 2016)*	73.74±3.01	50.98±2.27	64.38±4.52	40.39±1.30
IAN(Ma et al., 2017)*	76.34±0.27	55.16±1.97	68.49±0.57	44.51±0.48
RAM(Chen et al., 2017)*	76.97±0.64	55.85±1.60	68.48±0.85	45.37±2.03
GCAE(Xue and Li, 2018)*	77.28±0.32	56.73±0.56	69.14±0.32	47.06±2.45
AGDT	**78.85**±0.45	**60.33**±1.01	**71.50**±0.85	**51.30**±1.26

Table 5: The accuracy of the aspect-term sentiment analysis task. '*' refers to citing from GCAE (Xue and Li, 2018).

CNN. This model focuses on extracting n-gram features to generate sentence representation for the sentiment classification.

TD-LSTM. This model uses two LSTMs to capture the left and right context of the term to generate target-dependent representations for the sentiment prediction.

IAN. This model employs two LSTMs and interactive attention mechanism to learn representations of the sentence and the aspect, and concatenates them for the sentiment prediction.

RAM. This model applies multiple attentions and memory networks to produce the sentence representation.

GCAE. It uses CNNs to extract features and then employs two Gated Tanh-Relu units to selectively output the sentiment information flow towards the aspect for predicting sentiment labels.

3.4 Main Results and Analysis

Aspect-Category Sentiment Analysis Task

We present the overall performance of our model and baseline models in Table 4. Results show that our AGDT outperforms all baseline models on both "restaurant-14" and "restaurant-large" datasets. ATAE-LSTM employs an aspect-weakly associative encoder to generate the aspect-specific sentence representation by simply concatenating the aspect, which is insufficient to exploit the given aspect. Although GCAE incorporates the

gating mechanism to control the sentiment information flow according to the given aspect, the information flow is generated by an aspect-independent encoder. Compared with GCAE, our AGDT improves the performance by 2.4% and 1.6% in the "DS" part of the two dataset, respectively. These results demonstrate that our AGDT can sufficiently exploit the given aspect to generate the aspect-guided sentence representation, and thus conduct accurate sentiment prediction. Our model benefits from the following aspects. First, our AGDT utilizes an aspect-guided encoder, which leverages the given aspect to guide the sentence encoding from scratch and generates the aspect-guided representation. Second, the AGDT guarantees that the aspect-specific information has been fully embedded in the sentence representation via reconstructing the given aspect. Third, the given aspect embedding is concatenated on the aspect-guided sentence representation for final predictions.

The "HDS", which is designed to measure whether a model can detect different sentiment polarities in a sentence, consists of replicated sentences with different sentiments towards multiple aspects. Our AGDT surpasses GCAE by a very large margin (**+11.4%** and **+4.9%** respectively) on both datasets. This indicates that the given aspect information is very pivotal to the accurate sentiment prediction, especially when the sentence

has different sentiment labels, which is consistent with existing work (Jiang et al., 2011; Ma et al., 2017; Wang et al., 2018). Those results demonstrate the effectiveness of our model and suggest that our AGDT has better ability to distinguish the different sentiments of multiple aspects compared to GCAE.

Aspect-Term Sentiment Analysis Task

As shown in Table 5, our AGDT consistently outperforms all compared methods on both domains. In this task, TD-LSTM and ATAE-LSTM use a aspect-weakly associative encoder. IAN, RAM and GCAE employ an aspect-independent encoder. In the "DS" part, our AGDT model surpasses all baseline models, which shows that the inclusion of A-GRU (aspect-guided encoder), aspect-reconstruction and aspect concatenated embedding has an overall positive impact on the classification process.

In the "HDS" part, the AGDT model obtains +3.6% higher accuracy than GCAE on the restaurant domain and +4.2% higher accuracy on the laptop domain, which shows that our AGDT has stronger ability for the multi-sentiment problem against GCAE. These results further demonstrate that our model works well across tasks and datasets.

Ablation Study

We conduct ablation experiments to investigate the impacts of each part in AGDT, where the GRU is stacked with 4 layers. Here "AC" represents aspect concatenated embedding , "AG" stands for A-GRU (Eq. (1) $\sim$ (8)) and "AR" denotes the aspect-reconstruction (Eq. (13) $\sim$ (15)).

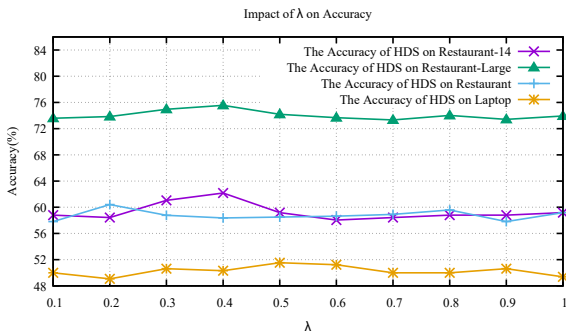From Table 6 and Table 7, we can conclude:

Figure 2: The impact of λ w.r.t. accuracy on "HDS".

1). Deep Transition (DT) achieves superior performances than GRU, which is consistent with

	AC	AG	AR	Rest-14		Rest-Large		
				DS	HDS	DS	HDS	
GRU	√	×	×	80.90	53.93	86.75	68.46	①
DT	√	×	×	81.74	56.63	87.54	72.39	②
	√	√	×	81.88	60.42	**87.72**	74.81	③
	×	√	×	**81.95**	59.33	87.68	74.44	④
	×	√	√	81.83	61.35	87.34	75.56	⑤
	√	√	√	81.78	**62.02**	87.55	**75.73**	⑥

Table 6: Ablation study of the AGDT on the aspect-category sentiment analysis task. Here "AC", "AG" and "AR" represent aspect concatenated embedding, A-GRU and aspect-reconstruction, respectively, '√' and '×' denotes whether to apply the operation. 'Rest-14': Restaurant-14, 'Rest-Large': Restaurant-Large.

	AC	AG	AR	Restaurant		Laptop		
				DS	HDS	DS	HDS	
GRU	√	×	×	78.31	55.92	70.21	46.48	①
DT	√	×	×	78.36	56.24	71.07	47.59	②
	√	√	×	78.77	60.14	71.42	50.83	③
	×	√	×	78.55	60.08	71.38	50.74	④
	×	√	√	78.59	60.16	71.47	51.11	⑤
	√	√	√	**78.85**	**60.33**	**71.50**	**51.30**	⑥

Table 7: Ablation study of the AGDT on the aspect-term sentiment analysis task.

previous work (Miceli Barone et al., 2017; Meng and Zhang, 2019) (② vs. ①).

2). Utilizing "AG" to guide encoding aspect-related features from scratch has a significant impact for highly competitive results and particularly in the "HDS" part, which demonstrates that it has the stronger ability to identify different sentiment polarities towards different aspects. (③ vs. ②).

3). Aspect concatenated embedding can promote the accuracy to a degree (④ vs. ③).

4). The aspect-reconstruction approach ("AR") substantially improves the performance, especially in the "HDS" part (⑤ vs. ④).

5). the results in ⑥ show that all modules have an overall positive impact on the sentiment classification.

Impact of Model Depth

We have demonstrated the effectiveness of the AGDT. Here, we investigate the impact of model depth of AGDT, varying the depth from 1 to 6. Table 8 shows the change of accuracy on the test

Depth	1	2	3	4	5	6
$\mathbb{D}_1$ DS	81.12	81.45	81.52	**81.78**	81.07	80.68
$\mathbb{D}_1$ HDS	55.73	57.08	60.67	**62.02**	59.10	58.65
$\mathbb{D}_2$ DS	87.20	87.47	87.53	**87.55**	87.11	87.21
$\mathbb{D}_2$ HDS	73.93	74.27	**76.07**	75.73	75.56	74.27
$\mathbb{D}_3$ DS	78.18	77.94	78.69	**78.85**	78.40	77.88
$\mathbb{D}_3$ HDS	59.35	58.94	59.43	**60.33**	59.27	57.80
$\mathbb{D}_4$ DS	71.13	71.10	**71.62**	71.50	71.16	70.86
$\mathbb{D}_4$ HDS	49.44	50.00	50.56	**51.30**	49.81	49.63

Table 8: The accuracy of model depth on the four datasets. '$\mathbb{D}_1$': Restaurant-14, '$\mathbb{D}_2$': Restaurant-Large, '$\mathbb{D}_3$': Restaurant, '$\mathbb{D}_4$': Laptop.

	Rest-14	Rest-Large	Rest.	Laptop
DS	99.55	99.80	76.21	70.92

Table 9: The accuracy of aspect reconstruction on the full test set. 'Rest-14': Restaurant-14, 'Rest-Large': Restaurant-Large, 'Rest.': Restaurant.

sets as depth increases. We find that the best results can be obtained when the depth is equal to 4 at most case, and further depth do not provide considerable performance improvement.

Effectiveness of Aspect-reconstruction Approach

Here, we investigate how well the AGDT can reconstruct the aspect information. For the aspect-term reconstruction, we count the construction is correct when all words of the term are reconstructed. Table 9 shows all results on four test datasets, which shows the effectiveness of aspect-reconstruction approach again.

Impact of Loss Weight λ

We randomly sample a temporary development set from the "HDS" part of the training set to choose the lambda for each dataset. And we investigate the impact of λ for aspect-oriented objectives. Specifically, λ is increased from 0.1 to 1.0. Figure 2 illustrates all results on four "HDS" datasets, which show that reconstructing the given aspect can enhance aspect-specific sentiment features and thus obtain better performances.

Comparison on Three-Class for the Aspect-Term Sentiment Analysis Task

We also conduct a three-class experiment to compare our AGDT with previous models, i.e., IARM, TNet, VAE, PBAN, AOA and MGAN, in Table 10.

Models	Rest.	Laptop
IARM(Majumder et al., 2018)*	80.00	73.80
TNet(Li et al., 2018a)*	80.79	**76.54**
VAE(Xu and Tan, 2018)*	81.10	75.34
PBAN(Gu et al., 2018)*	81.16	74.12
AOA(Huang et al., 2018)*	81.20	74.50
MGAN(Fan et al., 2018)*	81.25	75.39
DAuM(Zhu and Qian, 2018)*	82.32	74.45
AGDT	**82.95**	75.86

Table 10: The three-class accuracy of the aspect-term sentiment analysis task on SemEval 2014. '*' refers to citing from the original paper. 'Rest.': Restaurant.
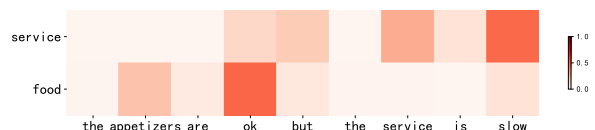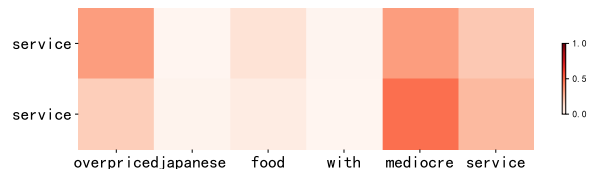

Figure 3: The output of A-GRU.

Figure 4: The above is the output of A-GRU. The bottom is the output after reconstructing the given aspect.

These previous models are based on an aspect-independent (weakly associative) encoder to generate sentence representations. Results on all domains suggest that our AGDT substantially outperforms most competitive models, except for the TNet on the laptop dataset. The reason may be TNet incorporates additional features (e.g., position features, local ngrams and word-level features) compared to ours (only word-level features).

4 Analysis and Discussion

Case Study and Visualization. To give an intuitive understanding of how the proposed A-GRU works from scratch with different aspects, we take a review sentence as an example. As the example "*the appetizers are ok, but the service is slow.*" shown in Table 1, it has different sentiment labels towards different aspects. The color depth denotes the semantic relatedness level between the given aspect and each word. More depth means stronger relation to the given aspect.

Figure 3 shows that the A-GRU can effectively guide encoding the aspect-related features with

the given aspect and identify corresponding sentiment. In another case, "*overpriced Japanese food with mediocre service.*", there are two extremely strong sentiment words. As the above of Figure 4 shows, our A-GRU generates almost the same weight to the word "*overpriced*" and "*mediocre*". The bottom of Figure 4 shows that reconstructing the given aspect can effectively enhance aspect-specific sentiment features and produce correct sentiment predictions.

Error Analysis. We further investigate the errors from AGDT, which can be roughly divided into 3 types. **1)** The decision boundary among the sentiment polarity is unclear, even the annotators can not sure what sentiment orientation over the given aspect in the sentence. **2)** The "conflict/neutral" instances are extremely easily misclassified as "positive" or "negative", due to the imbalanced label distribution in training corpus[3]. **3)** The polarity of complex instances is hard to predict, such as the sentence that express subtle emotions, which are hardly effectively captured, or containing negation words (e.g., *never*, *less* and *not*), which easily affect the sentiment polarity.

5 Related Work

Sentiment Analysis. There are kinds of sentiment analysis tasks, such as document-level (Thongtan and Phienthrakul, 2019), sentence-level[4] (Zhang and Zhang, 2019; Zhang et al., 2019), aspect-level (Pontiki et al., 2014; Wang et al., 2019a) and multimodal (Chen et al., 2018; Akhtar et al., 2019) sentiment analysis. For the aspect-level sentiment analysis, previous work typically apply attention mechanism (Luong et al., 2015) combining with memory network (Weston et al., 2014) or gating units to solve this task (Tang et al., 2016b; He et al., 2018a; Huang and Carley, 2018; Xue and Li, 2018; Duan et al., 2018; Tang et al., 2019; Yang et al., 2019; Bao et al., 2019), where an aspect-independent encoder is used to generate the sentence representation. In addition, some work leverage the aspect-weakly associative encoder to generate aspect-specific sentence representation (Tang et al., 2016a; Wang et al., 2016; Majumder et al., 2018). All of these methods make insufficient use of the given aspect information. There are also some work which

jointly extract the aspect term (and opinion term) and predict its sentiment polarity (Schmitt et al., 2018; Li et al., 2018b; Ma et al., 2018; Angelidis and Lapata, 2018; He et al., 2019; Luo et al., 2019; Hu et al., 2019; Dai and Song, 2019; Wang et al., 2019b). In this paper, we focus on the latter problem and leave aspect extraction (Shu et al., 2017) to future work. And some work (Sun et al., 2019; Xu et al., 2019; He et al., 2018b; Xu and Tan, 2018; Chen and Qian, 2019; He et al., 2019) employ the well-known BERT (Devlin et al., 2018) or document-level corpora to enhance ABSA tasks, which will be considered in our future work to further improve the performance.

Deep Transition. Deep transition has been proved its superiority in language modeling (Pascanu et al., 2014) and machine translation (Miceli Barone et al., 2017; Meng and Zhang, 2019). We follow the deep transition architecture in Meng and Zhang (2019) and extend it by incorporating a novel A-GRU for ABSA tasks.

6 Conclusions

In this paper, we propose a novel aspect-guided encoder (AGDT) for ABSA tasks, based on a deep transition architecture. Our AGDT can guide the sentence encoding from scratch for the aspect-specific feature selection and extraction. Furthermore, we design an aspect-reconstruction approach to enforce AGDT to reconstruct the given aspect with the generated sentence representation. Empirical studies on four datasets suggest that the AGDT outperforms existing state-of-the-art models substantially on both aspect-category sentiment analysis task and aspect-term sentiment analysis task of ABSA without additional features.

Acknowledgments

---

[3]More details can be seen in the dataset or see here: http://alt.qcri.org/semeval2014/

[4]https://nlp.stanford.edu/sentiment/

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. Tensorflow: A system for large-scale machine learning. In *OSDI*, pages 265–283.

Md Shad Akhtar, Dushyant Chauhan, Deepanway Ghosal, Soujanya Poria, Asif Ekbal, and Pushpak Bhattacharyya. 2019. Multi-task learning for multimodal emotion recognition and sentiment analysis. In *ACL*, pages 370–379.

Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. *CoRR*, abs/1808.08858.

Lingxian Bao, Patrik Lambert, and Toni Badia. 2019. Attention and lexicon regularized LSTM for aspect-based sentiment analysis. In *ACL*, pages 253–259.

F. Chen, R. Ji, J. Su, D. Cao, and Y. Gao. 2018. Predicting microblog sentiments via weakly supervised multimodal deep learning. *IEEE Transactions on Multimedia*, 20(4):997–1007.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *EMNLP*, pages 452–461.

Zhuang Chen and Tieyun Qian. 2019. Transfer capsule network for aspect level sentiment classification. In *ACL*, pages 547–556.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, pages 1724–1734.

Hongliang Dai and Yangqiu Song. 2019. Neural aspect and opinion term extraction with mined rules as weak supervision. In *ACL*, pages 5268–5277.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Junwen Duan, Xiao Ding, and Ting Liu. 2018. Learning sentence representations over tree structures for target-dependent classification. In *ACL*, pages 551–560.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *EMNLP*, pages 3433–3442.

Shuqin Gu, Lipeng Zhang, Yuexian Hou, and Yin Song. 2018. A position-aware bidirectional attention network for aspect-level sentiment analysis. In *COLING*, pages 774–784.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018a. Effective attention modeling for aspect-level sentiment classification. In *COLING*, pages 1121–1131.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018b. Exploiting document knowledge for aspect-level sentiment classification. In *ACL*, pages 579–585.

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 *ACL*, pages 504–515.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019. Open-domain targeted sentiment analysis via span-based extraction and classification. In *ACL*, pages 537–546.

Binxuan Huang and Kathleen Carley. 2018. Parameterized convolutional neural networks for aspect level sentiment classification. In *EMNLP*, pages 1091–1096.

Binxuan Huang, Yanglan Ou, and Kathleen M. Carley. 2018. Aspect level sentiment classification with attention-over-attention neural networks. *CoRR*, abs/1804.06536.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. 2011. Target-dependent twitter sentiment classification. In *ACL*, pages 151–160.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Xin Li, Lidong Bing, Wai Lam, and Bei Shi. 2018a. Transformation networks for target-oriented sentiment classification. In *ACL*, pages 946–956.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2018b. A unified model for opinion target extraction and target sentiment prediction. *CoRR*, abs/1811.05082.

Bin Liang, Jiachen Du, Ruifeng Xu, Binyang Li, and Hejiao Huang. 2019. Context-aware embedding for targeted aspect-based sentiment analysis. In *ACL*, pages 4678–4683.

Jiangming Liu and Yue Zhang. 2017. Attention modeling for targeted sentiment. In *ACL*, pages 572–577.

Huaishao Luo, Tianrui Li, Bing Liu, and Junbo Zhang. 2019. DOER: Dual cross-shared RNN for aspect term-polarity co-extraction. In *ACL*, pages 591–601.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*, pages 1412–1421.

Dehong Ma, Sujian Li, and Houfeng Wang. 2018. Joint learning for targeted sentiment analysis. In *EMNLP*, pages 4737–4742.

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. In *IJCAI*, pages 4068–4074.

Navonil Majumder, Soujanya Poria, Alexander Gelbukh, Md Shad Akhtar, Erik Cambria, and Asif Ekbal. 2018. Iarm: Inter-aspect relation modeling with memory networks in aspect-based sentiment analysis. In *EMNLP*, pages 3402–3411.

Fandong Meng and Jinchao Zhang. 2019. DTMT: A novel deep transition architecture for neural machine translation. *CoRR*, abs/1812.07807.

Antonio Valerio Miceli Barone, Jindřich Helcl, Rico Sennrich, Barry Haddow, and Alexandra Birch. 2017. Deep architectures for neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 99–107.

Razvan Pascanu, aglar Glehre, Kyunghyun Cho, and Yoshua Bengio. 2014. How to construct deep recurrent neural networks. *CoRR*, abs/1312.6026.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2012. Understanding the exploding gradient problem. *CoRR*, abs/1211.5063.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *SemEval*, pages 27–35.

Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016a. A hierarchical model of reviews for aspect-based sentiment analysis. In *EMNLP*, pages 999–1005.

Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016b. Insight-1 at semeval-2016 task 5: Deep learning for multilingual aspect-based sentiment analysis. In *SemEval*, pages 330–336.

Martin Schmitt, Simon Steinheber, Konrad Schreiber, and Benjamin Roth. 2018. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. In *EMNLP*, pages 1109–1114.

Lei Shu, Hu Xu, and Bing Liu. 2017. Lifelong learning CRF for supervised aspect extraction. *CoRR*, abs/1705.00251.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *CoRR*, abs/1903.09588.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *COLING*, pages 3298–3307.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. In *EMNLP*, pages 214–224.

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 *ACL*, pages 557–566.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2017. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. *CoRR*, abs/1712.05403.

Tan Thongtan and Tanasanee Phienthrakul. 2019. Sentiment classification using document embeddings trained with cosine similarity. In *ACL*, pages 407–414.

Jingjing Wang, Changlong Sun, Shoushan Li, Xiaozhong Liu, Luo Si, Min Zhang, and Guodong Zhou. 2019a. Aspect sentiment classification towards question-answering with reinforced bidirectional attention network. In *ACL*, pages 3548–3557.

Shuai Wang, Sahisnu Mazumder, Bing Liu, Mianwei Zhou, and Yi Chang. 2018. Target-sensitive memory networks for aspect sentiment classification. In *ACL*, pages 957–967.

Yequan Wang, Minlie Huang, xiaoyan zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *EMNLP*, pages 606–615.

Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019b. Aspect-level sentiment analysis using as-capsules. In *WWW*, pages 2033–2044.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *CoRR*, abs/1410.3916.

Bowen Xing, Lejian Liao, Dandan Song, Jingang Wang, Fuzheng Zhang, Zhongyuan Wang, and Heyan Huang. 2019. Earlier attention? aspect-aware LSTM for aspect sentiment analysis. *CoRR*, abs/1905.07719.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. *CoRR*, abs/1904.02232.

Weidi Xu and Ying Tan. 2018. Semi-supervised target-level sentiment analysis via variational autoencoder. *CoRR*, abs/1810.10437.

Wei Xue and Tao Li. 2018. Aspect based sentiment analysis with gated convolutional networks. In *ACL*, pages 2514–2523.

Chao Yang, Hefeng Zhang, Bin Jiang, and Keqin Li. 2019. Aspect-based sentiment analysis with alternating coattention networks. *Information Processing and Management*, 56:463–478.

Liwen Zhang, Kewei Tu, and Yue Zhang. 2019. Latent variable sentiment grammar. In *ACL*, pages 4642–4651.

Yuan Zhang and Yue Zhang. 2019. Tree communication models for sentiment analysis. In *ACL*, pages 3518–3527.

Peisong Zhu and Tieyun Qian. 2018. Enhanced aspect level sentiment classification with auxiliary memory. In *COLING*, pages 1077–1087.