An Iterative Knowledge Transfer Network with Routing 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) mainly involves three subtasks: aspect term extraction, opinion term extraction and aspect-level sentiment classification, which are typically handled separately or (partially) jointly. However, the semantic interrelationships among all the three subtasks are not well exploited in previous approaches, which restricts their performance. Additionally, the linguistic knowledge from document-level labeled sentiment corpora is usually used in a coarse way for the ABSA. To address these issues, we propose a novel **I**terative **K**nowledge **T**ransfer **N**etwork (IKTN) for the end-to-end ABSA. For one thing, to fully exploit the semantic correlations among the three aspect-level subtasks for mutual promotion, the IKTN transfers the task-specific knowledge from any two of the three subtasks to another one by leveraging a specially-designed routing algorithm, that is, any two of the three subtasks will help the third one. Besides, the IKTN discriminately transfers the document-level linguistic knowledge, i.e., domain-specific and sentiment-related knowledge, to the aspect-level subtasks to benefit the corresponding ones. Experimental results on three benchmark datasets demonstrate the effectiveness of our approach, which significantly outperforms existing state-of-the-art methods.

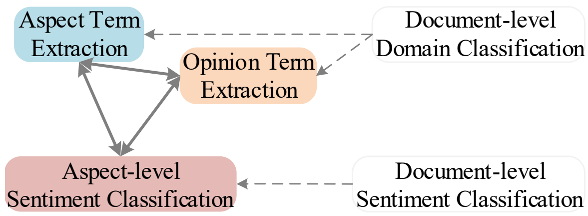

Figure 1: Overview of the relationships among three aspect-level subtasks (left part) and two document-level subtasks (right part). The three aspect-level subtasks are highly semantic correlated, and thus can incrementally facilitate one another through knowledge transfer. The knowledge from two document-level subtasks are discriminately transferred to enhance the corresponding aspect-level subtasks, namely, the domain-specific knowledge from document-level domain classification is transferred to aspect term extraction and opinion term extraction, and the sentiment-related knowledge from document-level sentiment classification is transferred to aspect-level sentiment classification.

1 Introduction

Aspect-based sentiment analysis (ABSA) includes three subtasks: **a**spect **t**erm **e**xtraction (ATE), **o**pinion **t**erm **e**xtraction[1] (OTE) and **a**spect-level **s**entiment **c**lassification (ASC). The first two subtasks aim to identify the aspect term and the opinion term appeared in one sentence, respectively. The goal of the ASC subtask is to detect the sentimental orientation towards the extracted aspect terms. Recently, the ABSA task has drawn an increasing attention in the community. Most of existing studies focus on joint models (Wang et al., 2016a, 2017; Dai and Song, 2019; Luo et al., 2019b; He et al., 2019) or incorporating document-level sentiment corpora (He et al., 2018, 2019; Dai and Song, 2019; Chen and Qian, 2019) to better complete these subtasks. Typically, these joint models only couple two subtasks, without modeling semantic interrelationships among all the three subtasks. Besides, the document-level corpora are usually coarsely applied, which is insufficient to exert its advantages.

The nature of the ABSA task determines that the relationships among its three sub-tasks are inseparable, just like the left part of Figure 1, showing that they can incrementally promote one another.

* Work was done when Yunlong Liang was an intern at Pattern Recognition Center, WeChat AI, Tencent Inc, China.
† Jinan Xu is the corresponding author.

[1]The ABSA task is more complete with the opinion term extraction subtask (He et al., 2019; Peng et al., 2019).

For example, for the sentence "*The **battery** is very longer.*" of the *Laptop* domain, the opinion term "*longer*" indicates that the sentiment polarity of the aspect term "***battery***" is *positive*, suggesting the strong semantic correlation among them. Given the aspect term "***battery***" and its sentiment polarity *positive*, the word "*longer*" rather than other words (e.g. "*very*") in the sentence will be easily extracted as an opinion term, that is, any two subtasks can help the third one. Some researchers also observe this and thus jointly perform the aspect term and opinion term co-extraction (Wang et al., 2016a, 2017; Dai and Song, 2019) or aspect term-polarity co-extraction (Luo et al., 2019b; He et al., 2019)[2], achieving promising performance in this direction. Unfortunately, these methods only couple two subtasks and neglect the potential and mutual effects among all the three subtasks.

In addition, since the easily accessible document-level sentiment corpora contain rich domain-specific and sentiment-related linguistic knowledge, thus, they can benefit the aspect-level tasks, which involve two document-level subtasks: domain classification and sentiment classification (He et al., 2019). Some studies focus on constructing auxiliary data (Dai and Song, 2019) or applying document-level sentiment classification to boost the ASC subtask by sharing the encoder (He et al., 2018; Chen and Qian, 2019), while the domain-specific properties[3] are not explicitly modeled. Although He et al. (2019) merge the knowledge from two document-level subtasks to enhance the ATE and ASC subtasks, such indiscriminate use of the document-level knowledge is coarse and thus limits its potential. Specifically, since the aspect term and the opinion term own domain-specific properties (Peng et al., 2018) while sentiment polarities are typically domain-invariant (i.e. *positive*, *negative* and *neutral*), therefore, the domain classification can help to extract the aspect term and the opinion term while it may be helpless when judging the sentiment polarity. Similarly, the document-level sentiment classification is more beneficial to the ASC subtask rather than the ATE and OTE subtasks, just as shown in the right part of Figure 1.

To address the issues above, we propose an **I**terative **K**nowledge **T**ransfer **N**etwork (IKTN) to fully exploit the semantic relationships at both the token level and the document level for the ABSA task. Particularly, we design a novel routing algorithm, which can mutually transfer task-specific knowledge among the three aspect-level subtasks, as illustrated in the left part of Figure 1. Furthermore, the IKTN employs a more fine-grained way to discriminately transfer document-level knowledge to the aspect-level subtasks, as shown in the right part of Figure 1, where the knowledge from domain classification subtask only serves for the ATE and OTE subtasks while the knowledge from document-level sentiment classification subtask only helps the ASC subtask. All the knowledge transfer processes are iteratively conducted for fully exploiting the knowledge in all the tasks to enhance the ABSA task.

We conduct experiments on three benchmark datasets. Results demonstrate the effectiveness of our approach, and we achieve consistent state-of-the-art performances on these benchmark datasets. Our contributions can be summarized as follows:

- We propose an iterative knowledge transfer network for the ABSA task, which can transfer the task-specific knowledge from any two of the three aspect-level subtasks to the third one for mutual promotion via a specially-designed routing algorithm.

- We propose a more fine-grained way to discriminately transfer the document-level knowledge to enhance the aspect-level tasks.

- Our approach significantly outperforms the existing methods and achieves new state-of-the-art results on three benchmark datasets.

2 Background

In this section, we mainly formulate the aspect-level tasks and document-level tasks, where the document-level tasks are taken as auxiliary tasks for improving the aspect-level tasks.

Aspect-level Tasks. The ABSA task is formulated as three sequence labeling subtasks. For

[2] He et al. (2019) take the aspect term extraction and opinion term extraction as a single sequence labeling task, i.e., using a unified tagging scheme {`BA`,`IA`,`BP`,`IP`,`O`} to label the aspect term, the opinion term and other words, respectively, without explicitly modeling the correlation between them. Thus, we refer it to the aspect term-polarity co-extraction.

[3] For instance, in the sentence "*...the longer you may have to wait*" of the *Restaurant* domain, the opinion term "*longer*" represents longer time, while it denotes durable "***battery***" in the case of the *Laptop* domain. That is, the meaning of the opinion term "*longer*" expresses completely different meanings when it occurs in different domains, i.e. domain-specific properties. Similarly, the aspect term also owns the properties.

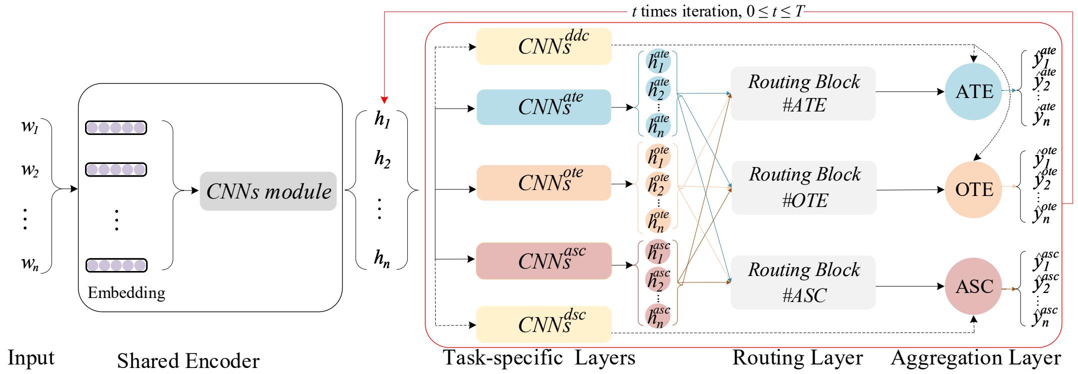

Figure 2: The model architecture of IKTN. ATE: aspect term extraction; OTE: opinion term extraction; ASC: aspect-level sentiment classification. To fully exploit the inter-task correlations among the three aspect-level subtasks for mutual reinforcing, the knowledge from them is mutually transferred to each other via the "Routing Block". Besides, the knowledge from CNN_s^{ddc} is only transferred to the ATE and OTE subtasks. The knowledge from CNN_s^{dsc} is only transferred to the ASC subtask. In summary, all the knowledge transfer processes are iteratively conducted for adequately exploiting the knowledge from all the tasks to enhance the ABSA task.

the ATE and OTE subtasks, we employ the BIO tagging scheme: $\mathcal{Y}^{ate} = \{\mathtt{BA}, \mathtt{IA}, \mathtt{O}\}$ and $\mathcal{Y}^{ote} = \{\mathtt{BP}, \mathtt{IP}, \mathtt{O}\}$ to label the aspect terms and the opinion terms appeared in one sentence, respectively. BA,IA,BP,IP and O denote the beginning and the inside of an **a**spect term, an o**p**inion term and **o**ther words, respectively. For the ASC subtask, we employ $\mathcal{Y}^{asc} = \{\mathtt{pos}, \mathtt{neg}, \mathtt{neu}\}$ to mark the word-level sentimental orientation, which denote *positive*, *negative* and *neutral* sentiment polarities, respectively. Given an input sentence $\mathrm{S} = \{w_1, w_2, \ldots, w_n\}$ with length n, we aim to inference three tag sequences $\mathrm{Y}^{ate} = \{y_1^{ate}, y_2^{ate}, \ldots, y_n^{ate}\}$, $\mathrm{Y}^{ote} = \{y_1^{ote}, y_2^{ote}, \ldots, y_n^{ote}\}$ and $\mathrm{Y}^{asc} = \{y_1^{asc}, y_2^{asc}, \ldots, y_n^{asc}\}$, where $y_i^{ate} \in \mathcal{Y}^{ate}$, $y_i^{ote} \in \mathcal{Y}^{ote}$ and $y_i^{asc} \in \mathcal{Y}^{asc}$, $1 \le i \le n$.

Document-level Tasks. This work contains two document-level subtasks: **d**ocument-level **d**omain **c**lassification (DDC) and **d**ocument-level **s**entiment **c**lassification (DSC). For an input document $\mathrm{D} = \{\mathrm{S}_1, \mathrm{S}_2, \ldots, \mathrm{S}_m\}$ with m sentences, the DDC and DSC aim to predict a domain label $\mathrm{Y}^{ddc} \in \{Laptop, Restaurant\}$ and a sentiment label $\mathrm{Y}^{dsc} \in \mathcal{Y}^{asc}$, respectively. To fully exploit the rich linguistic knowledge from the document-level corpora, we design a novel network to discriminately transfer it to enhance the aspect-level tasks.

3 Model

3.1 Overview

As shown in Figure 2, the IKTN mainly consists of four parts: 1) Shared Encoder, for extracting n-gram features, 2) Task-specific Layers, aiming to convert n-gram features into several task-specific representations, 3) Routing Layer, including three Routing Blocks, controlling to fully transfer knowledge among the aspect-level tasks for mutually reinforcing, and 4) Aggregation Layer, for aggregating the multi-source information for the next iteration.

Note that, knowledge generated by the DDC task only helps ATE and OTE subtasks and knowledge from the DSC task only serves for the ASC subtask. In summary, the IKTN aims to fully exploit the knowledge from all the subtasks to enhance the aspect-level tasks.

3.2 Shared Encoder

To extract n-gram features at different granularities, we adopt a few Convolutional Neural Networks (Kim, 2014) as the feature extractor, where each kernel corresponds to a linguistic feature detector (Kalchbrenner et al., 2014). The encoder is shared by the aspect-level tasks and the document-level tasks for providing common features.

3.3 Task-specific Layers

Based on the Shared Encoder, under the supervised signals of various subtasks, we design three aspect-level task-specific layers: CNN_s^{ate}, CNN_s^{ote} and CNN_s^{asc}, aiming to generate aspect-related knowledge, opinion-related knowledge and sentiment-related knowledge, respectively; and two document-level task-specific layers: CNN_s^{ddc} and CNN_s^{dsc}, for producing domain-specific features and sentimental features, respectively.

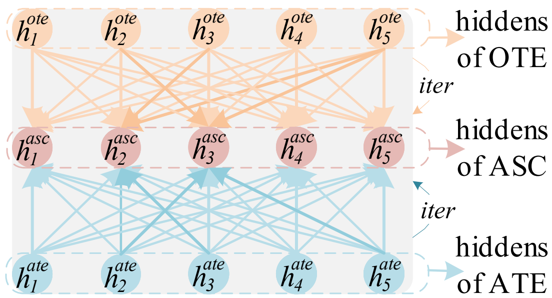

Figure 3: An example of the internal structure of "Routing Block #ASC" as shown in Figure 3. The knowledge from the ATE and OTE subtasks is transferred to the ASC subtask through $iter$ rounds of iteration, that is, the ATE and OTE subtasks will help the ASC subtask.

Algorithm 1 Routing

1: **procedure** ROUTING ALGORITHM($\mathbf{H}^{ote}$, $iter$)
2: $\quad \forall i \in OTE, \forall j \in ASC, 1 \leq i, j \leq n : b_{j|i} \leftarrow 0.$
3: $\quad \hat{\mathbf{u}}_{j|i} = \mathbf{H}_i^{ote}\mathbf{W}_{ij}$
4: $\quad$ **for** $iter$ iterations **do**
5: $\quad\quad b_{j|i} \leftarrow b_{j|i}$
6: $\quad\quad \forall i \in OTE$: $\mathbf{c}_i \leftarrow \text{softmax}(\mathbf{b}_i)$
7: $\quad\quad \forall j \in ASC$: $\mathbf{s}_j \leftarrow \Sigma_i c_{j|i}\hat{\mathbf{u}}_{j|i}$
8: $\quad\quad \forall j \in ASC : \mathbf{v}_j^{ote} \leftarrow \text{squash}(\mathbf{s}_j)$
9: $\quad\quad \forall i \in OTE.\forall j \in ASC : b_{j|i} \leftarrow b_{j|i} + \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j^{ote}$
10: $\quad$ **end for**
11: $\quad$ Return $\mathbf{v}_j^{ote}$
12: **end procedure**

3.4 Routing Layer

We design the routing layer to take full advantages of the inter-task correlations among the three aspect-level subtasks. Particularly, the routing layer consists of three "Routing Blocks" (in Figure 2).

Routing Block. The routing block serves for transferring knowledge among the aspect-level subtasks as shown in the "Routing Layer" part of Figure 2. Taking the "Routing Block #ASC" for example, its internal structure is shown in Figure 3, in which the knowledge from ATE and OTE is transferred to ASC for enhancing the ASC performance via the routing algorithm of dynamic-length. In original routing process (Sabour et al., 2017), the output are fixed number of capsules (e.g., number of categories). However, in our problem, the output are dynamic number of capsules (sentence length), and it is necessary to discern this in practical application. To this end, we propose a new routing algorithm by introducing sin and cos positional encoding (Vaswani et al., 2017) to obtain a shared yet position-aware weight matrix $\mathbf{W}$. The positional encoding functions used are as follows:

$$PE_{(pos,2p)} = sin(pos/10000^{2p/d_{model}})$$
$$PE_{(pos,2p+1)} = cos(pos/10000^{2p/d_{model}})$$

where pos is the position, p is the positional index of the dimension and d_{model} is the input dimension.

Subsequently, taking "transferring knowledge from OTE to ASC" for example, the pseudocode is shown as Algorithm 1, which determines the agreement value $c_{j|i}$. Specifically, the input $\mathbf{H}^{ote}$, $\mathbf{A}$ and $iter$ of Algorithm 1 are the representation of OTE, adjacency matrix (generated by dependency tree) and iteration number, respectively (line 1). The $b_{j|i}$ is the probability indicating that the i-th token representation in OTE agrees to be routed to the j-th token representation in ASC, which is initialized with zero (line 2). The $\hat{\mathbf{u}}_{j|i}$ denotes the resulting knowledge vector generated by multiplying the representation $\mathbf{H}_i^{ote}$ by a weight matrix $\mathbf{W}_{ij}$ (line 3).

During each iteration (line 4), the probability $b_{j|i}$ is summed by $b_{j|i}$ and $A_{j|i}$, which is the prior knowledge indicating whether there is dependency relationship between the i-th token and the j-th token or not (line 5), and the coupling coefficients are obtained by applying the softmax function (line 6) and each sentiment representation $\mathbf{v}_j$ is calculated by aggregating all the opinion vectors for the j-th token, voting for the sentiment polarity of the j-th token, weighted by the agreement values $c_{j|i}$ obtained from $b_{j|i}$ (line 7). The $\text{squash}(\mathbf{s}_j) = \frac{||\mathbf{s}_j||^2}{1+||\mathbf{s}_j||^2}\frac{\mathbf{s}_j}{||\mathbf{s}_j||}$ scales the output $\mathbf{s}_j$ non-linearly (line 8). Once the $\mathbf{v}_j$ is updated in the current iteration, the probability $b_{j|i}$ becomes larger if the dot product $\hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j^{ote}$ is large. That is, when an opinion vector $\hat{\mathbf{u}}_{j|i}$ is more similar to the sentiment representation $\mathbf{v}_j^{ote}$, the dot product is larger, meaning that it is more likely to route this opinion knowledge to the j-th token (line 9) and thus affects its sentimental orientation. Therefore, larger $b_{j|i}$ will lead to a larger agreement value $c_{j|i}$ between the opinion knowledge of the i-th token and the sentiment representation of the j-th token in the next iteration. In contrast, it generates low $c_{j|i}$ when there is no correlation between $\hat{\mathbf{u}}_{j|i}$ and $\mathbf{v}_j^{ote}$. After multiple iterations, agreement values learned via the routing process ensures the opinion knowledge is sent to the appropriate sentiment representation (line 11).

Similarly, we can obtain the knowledge $\mathbf{v}_j^{ate}$ which is transferred from ATE to ASC, indicating which token should be correctly labeled with the sentiment polarity. The knowledge from ATE and OTE subtasks is combined as follows:

$$\mathbf{h}_j^{asc} = Concat(\mathbf{h}_j^{asc}, \mathbf{v}_j^{ate}, \mathbf{v}_j^{ote})$$

where $\mathbf{h}_j^{asc}$ is the j-th hidden of the ASC subtask.

We achieve knowledge transfer in "Routing Block #OTE" and "Routing Block #ATE" in Figure 2 through the process above. In doing so, the three subtasks are interacted with one another to fully exploit the inter-task correlations.

3.5 Aggregation Layer

This layer aims to aggregate the multi-source knowledge, i.e., the task-specific knowledge, the predictions of the aspect-level tasks from the previous iteration, which is proved helpful in (He et al., 2019), and the document-level knowledge. Specifically, the functions used are as follows:

$$\mathbf{h}_i^{q(t+1)} = f_1(\mathbf{h}_i^{q(t)}; \hat{\mathbf{y}}_i^{ate(t)}; \hat{\mathbf{y}}_i^{ote(t)}; \hat{\mathbf{y}}_i^{asc(t)}; a_i^{ddc(t)})$$

$$\mathbf{h}_i^{asc(t+1)} = f_2(\mathbf{h}_i^{asc(t)}; \hat{\mathbf{y}}_i^{ate(t)}; \hat{\mathbf{y}}_i^{ote(t)}; \hat{\mathbf{y}}_i^{asc(t)}; \hat{\mathbf{y}}^{dsc(t)}; a_i^{dsc(t)})$$

where $q \in \{ate, ote\}$, t is the iteration number ($0 \leq t \leq T$), [;] means concatenation operation, f_1 and f_2 are fully-connected layers and $\hat{\mathbf{y}}_i^{o(t)}$ is the prediction on the i-th token at the t-th iteration, $o \in \{ate, ote, asc\}$. Note that the ATE and OTE subtasks only take the domain-specific knowledge from the DDC subtask, i.e. $a_i^{ddc(t)}$, and the ASC subtask only merges the sentiment-related knowledge from the DSC subtask, i.e. $\hat{\mathbf{y}}^{dsc(t)}$ and $a_i^{dsc(t)}$. $a_i^{s(t)}$ ($s \in \{ddc, dsc\}$) is the self-attention weight:

$$a_i^{s(t)} = \frac{exp(\mathbf{h}_i^{s(t)}\mathbf{W}^s)}{\sum_{k=1}^n exp(\mathbf{h}_k^{s(t)}\mathbf{W}^s)}$$

where $\mathbf{W}^s$ is the trainable parameter. The document representation is computed by $\mathbf{h}^{s(t)} = \sum_{i=1}^n a_i^{s(t)}\mathbf{h}_i^{s(t)}$. Then a fully-connected layer with softmax function is applied to map $\mathbf{h}^{s(t)}$ to $\hat{\mathbf{y}}^{s(t)}$.

Overall, the IKTN can fully perform knowledge transfer via the routing algorithm and incorporate the document-level knowledge to discriminately enhance the corresponding aspect-level tasks through such T rounds of iteration.

3.6 Training

For training, we minimize the loss on each token of aspect-level tasks and each instance of document-level tasks with the cross-entropy function. The aspect-level loss functions are written as follows:

$$\mathcal{J}_a = \lambda_1\mathcal{L}_{ate} + \lambda_2\mathcal{L}_{ote} + \lambda_3\mathcal{L}_{asc}$$

$$\mathcal{L}_o = \frac{1}{n}\sum_{i=1}^{n}(min(-\sum_{r=0}^{C_1}\mathbf{y}_{i,r}^o \log(\hat{\mathbf{y}}_{i,r}^{o(T)}))$$

where λ_1, λ_2 and λ_3 are discount coefficients, $o \in \{ate, ote, asc\}$, n is the sentence length, C_1 is the class number, $\mathbf{y}_{i,r}^o$ denotes the ground-truth and $\hat{\mathbf{y}}_{i,r}^{o(T)}$ denotes the predictions with T times iteration, respectively. The document-level loss functions are formulated as follows:

$$\mathcal{J}_d = \lambda_4\mathcal{L}_{ddc} + \lambda_5\mathcal{L}_{dsc}$$

$$\mathcal{L}_s = min(-\sum_{r=0}^{C_2}\mathbf{y}_r^s \log(\hat{\mathbf{y}}_r^{s(T)}))$$

where λ_4 and λ_5 are discount coefficients, $s \in \{ddc, dsc\}$, C_2 is the class number, $\mathbf{y}_r^s$ denotes the ground-truth and $\hat{\mathbf{y}}_r^{s(T)}$ denotes the predictions with T times iteration, respectively.

For the whole model training, we first pretrain the model with document-level tasks for a few epochs to generate reasonable features for aspect-level tasks. Subsequently, we train the network on aspect-level corpus and document-level corpus alternately, to minimize the corresponding loss.

4 Experiments

4.1 Experiment Settings

Datasets. Table 1 is the data statistics from SemEval 2014 (Pontiki et al., 2014) and SemEval 2015 (Pontiki et al., 2015). The opinion terms of these three datasets are annotated by (Wang et al., 2016a). We adopt two document-level datasets from (He et al., 2018, 2019), which include 30k instances of Yelp restaurant domain and 30k instances of Amazon electronic domain, respectively. We merge the two datasets with domain labels for domain classification. We use the Yelp data when training on D1 and D3, and use the Amazon data for D2, due to the domain-specific properties.

Implementation Details. Following (He et al., 2019), we use 300d GloVe[4] released by (Pennington et al., 2014) as general-specific embeddings and the embeddings released by (Xu et al., 2018) as domain-specific embeddings. Our models[5] are trained by Adam optimizer (Kingma and Ba, 2014), with learning rate $\eta_0 = 10^{-4}$, and batch size is set to 32. When training, we randomly sample 20%

[4]https://nlp.stanford.edu/projects/glove/

[5]Code: https://github.com/XL2248/IKTN

Dataset			Sentence	Aspect Term	Opinion Term
D1	Restaurant14	train	3,044	3,699	3,484
		test	800	1,134	1,008
D2	Laptop14	train	3,048	2,373	2,504
		test	800	654	674
D3	Restaurant15	train	1,315	1,199	1,210
		test	685	542	510

Table 1: Statistics of the aspect-level datasets.

of each training data as the development set and the remaining 80% as training set. We tune the iteration number T and the routing number $iter$ on each development set. The tuning details and more implementation details are given in Appendix A, B.

Evaluation Metrics. Following (He et al., 2019), five metrics are applied for evaluation and the average score over 5 runs are reported with random initialization in all experiments. We use F1 score denoted as **F1-I** to measure the aspect term-polarity co-extraction performance, where an extracted aspect term is taken as correct only when the span and the sentiment are both correct. For the ATE and OTE subtasks, we use F1 to measure the performance denoted as **F1-a** and **F1-o**, respectively. For the ASC subtask, we adopt accuracy and macro-F1 denoted as **acc-s** and **F1-s**, respectively, which are computed based on the correctly extracted aspect term from the ATE instead of the golden ones.

4.2 Compared Models

- {**CMLA**, **DECNN**}-{**ALSTM**, **dTrans**}: CMLA (Wang et al., 2017) focuses on aspect term and opinion term co-extraction through modeling their inter-dependencies. DECNN (Xu et al., 2018) proposes double embeddings for the ATE subtask. ALSTM (Wang et al., 2016b) is an attention-based structure and the dTrans (He et al., 2018) introduces a large document-level corpus to improve the ASC performance.

- **PIPELINE-IMN**: It is the pipeline setting of IMN (He et al., 2019), which performs ATE, OTE and ASC subtasks separately.

- **SPAN-pipeline** (Hu et al., 2019b): It utilizes BERT as backbone networks for ATE and ASC subtasks, which is a strong baseline.

- **MNN** (Wang et al., 2018) and **INABSA** (Li et al., 2019a): The two models all handle the aspect term-polarity co-extraction as a sequence labeling problem by using a unified tagging scheme.

- **BERT+GRU** (Li et al., 2019b): It focuses on exploring the potential of BERT for this task.

- **DOER** (Luo et al., 2019b): This model jointly perform aspect term and polarity co-extraction with a cross-shared unit.

- **IMN** (He et al., 2019): It uses an interactive multi-task architecture and message-passing mechanism to pass the document-level knowledge to aspect-level tasks, which is the current state-of-the-art method for end-to-end ABSA task. IMN^{-d} is the variant of IMN, where $^{-d}$ denotes without using the document-level corpora.

4.3 Results and Analysis

Main Results. Table 2 and Table 3 are the results of ours and baseline models for the end-to-end ABSA task[6]. Results suggest that our IKTN consistently outperforms all baseline models by a large margin in most cases under the setting without BERT. We can conclude from Table 2 and 3:

1) For all three aspect-level subtasks (F1-a, F1-o, F1-s and acc-s), Table 2 shows that our IKTN can significantly surpass other baselines in most cases on three subtasks. This suggests that the inter-task correlations and document-level knowledge have an overall positive impact on these subtasks, and demonstrates the superiority of our model.

2) For aspect term-polarity co-extraction (F1-I), Table 2 shows that our IKTN can performs the best than other baselines. Specifically, IKTN outperforms the best F1-I results of IMN by **1.56%**, **2.23%**, and **2.13%** on D1, D2 and D3, respectively, indicating that the IKTN indeed can benefit from the knowledge transfer at both the token level and the document level. We achieve further improvements by using *BERT* features (**+2.21%**, **+3.97%**, and **+3.15%** compared with IMN, respectively).

3) We compare with other models in Table 3, which adopt different dataset settings from ours. We find that the IKTN can surpass the DOER under the setting without BERT. And our model

[6] Peng et al. (2019) also obtain good results for this task, while they focus on the limited scenario where the aspect term and the opinion term are paired in one sentence.

	Methods	D1					D2					D3				
		F1-a	F1-o	F1-s	acc-s	F1-I	F1-a	F1-o	F1-s	acc-s	F1-I	F1-a	F1-o	F1-s	acc-s	F1-I
Pipeline Models	CMLA-ALSTM*	82.45	82.67	68.70	77.46	63.87	76.80	77.33	66.67	70.25	53.68	68.55	71.07	58.91	81.03	54.79
	CMLA-dTrans*	82.45	82.67	72.23	79.58	65.34	76.80	77.33	69.52	72.38	55.56	68.55	71.07	66.45	82.27	56.09
	DECNN-ALSTM*	83.94	85.60	68.50	77.79	65.26	78.38	78.81	66.78	70.46	55.05	68.32	71.22	57.25	80.32	55.10
	DECNN-dTrans*	83.94	85.60	73.31	80.04	67.25	78.38	78.81	70.63	73.10	56.60	68.32	71.22	69.58	82.65	56.28
	PIPELINE-IMN*	83.94	85.60	69.59	79.56	66.53	78.38	78.81	68.12	72.29	56.02	68.32	71.22	59.53	82.27	55.96
Integrated Models	MNN*	83.05	84.55	68.45	77.17	63.87	76.94	77.77	65.98	70.40	53.80	70.24	69.38	57.90	80.79	56.57
	INABSA*	83.92	84.97	68.38	79.68	66.60	77.34	76.62	68.24	72.30	55.88	69.40	71.43	58.81	82.56	57.38
Joint Models	IMN^{-d*}	84.01	85.64	71.90	81.56	68.32	78.46	78.14	69.92	73.21	57.66	69.80	72.11	60.65	83.38	57.91
	IMN*	83.33	85.61	75.66	83.89	69.54	77.96	77.51	72.02	75.36	58.37	70.04	71.94	71.76	85.64	59.18
	IKTN^{-d} (Ours)	84.59	84.89	73.20	81.32	68.64	80.10	74.22	74.47	69.95	59.36	71.25	71.85	66.76	81.96	58.25
	IKTN (Ours)	83.91	84.65	**76.66**	**84.93**	71.10	79.19	76.80	73.13	76.92	60.60	70.96	72.48	**72.39**	86.67	61.31
	IKTN+BERT (Ours)	**86.13**	**86.62**	74.35	83.47	**71.75**	**80.89**	**78.90**	**73.42**	**77.51**	**62.34**	**71.63**	**76.79**	69.85	**87.10**	**62.33**

Table 2: Model comparison. The results with symbol "*" refer to IMN (He et al., 2019). "$^{-d}$" represents not using document-level corpus. "+BERT" denotes exploiting *BERT-Base* features on our architecture. Average results over 5 runs with random initialization are reported.

#	Models	D2
0	DOER (Luo et al., 2019b)♮	59.48
1	IKTN (Ours)	**60.60**
2	BERT+GRU (Li et al., 2019b)♮	60.42
3	SPAN-pipeline (Hu et al., 2019b)♮	61.84
4	IKTN+BERT (Ours)	**62.34**

Table 3: F1-I (%) scores. Since the dataset D2 is our common corpus, so we conduct experiments on it for fair comparison. "♮" indicates that the results are generated by running their released code. Rows 2∼4 are the results with the assistance of *BERT-Base, Uncased* model, where we establish new state-of-the-art results.

#	Models	D1		
		F1-a	F1-o	F1-s
0	IKTN (Coarse)	83.45	83.75	75.03
1	IKTN (This work)	**83.91**	**84.65**	**76.66**
2	– aspect knowledge transfer	83.53	84.10	74.07
3	– opinion knowledge transfer	83.64	83.65	74.01
4	– sentiment knowledge transfer	83.33	83.49	75.40
5	– knowledge transfer from DDC	83.14	83.34	75.13
6	– knowledge transfer from DSC	83.67	84.57	72.76

Table 4: F1-a (%), F1-o (%) and F1-s (%) scores of ablation study on D1, and the results on D2 and D3 are given in Appendix C, which aim to investigate the impacts of the different knowledge. The "Coarse" is the usage from IMN (He et al., 2019), meaning the knowledge from two document-level subtasks is merged to indistinguishably enhance the aspect-level tasks. The "– aspect knowledge transfer" represents removing the aspect knowledge transfer and its interactions with other subtasks correspondingly. Similarly for rows 3∼6.

with BERT (row 4) can also outperform the SPAN-pipeline (row 3) and "BERT+GRU" (row 2), which suggests the effectiveness of our approach.

Ablation Study. We investigate the impact of different knowledge in Table 4, where we remove one knowledge at a time. And we conclude that: 1) our fine-grained application of the document-level knowledge has better effect than the "Coarse" one (row 1 vs. row 0); 2) once any of the aspect-level subtask knowledge transfer is removed (row 1 vs. rows 2∼4), scores on three subtasks decrease to some extent, and a more evident decline occurs on the ASC subtask (F1-s), which shows that the three aspect-level subtasks are highly semantically correlated and thus can incrementally boost one another; 3) we also observe obvious drops, especially on the ATE and OTE subtasks (F1-a and F1-o) when removing the document-level knowledge from the DDC subtask (row 5), and on the ASC subtask (F1-s) when removing it from the DSC subtask (row 6), which shows that discriminately transferring the document-level knowledge can significantly benefits the corresponding aspect-level tasks (row 1 vs. rows 5∼6), leading to better performances.

Case Study and Visualization. To provide an understanding of how the knowledge transfer works, we take the knowledge transfer from OTE and ATE to ASC for example to visualize the agreement value $c_{j|i}$ in Figure 4. Figure 4(a) and 4(c) are the case of transferring knowledge from OTE to ASC, which shows that the knowledge from the OTE subtask can be effectively sent to the ASC subtask, indicating that the opinion word determines the sentimental polarity, i.e., the former (ATE) is naturally correlated with the latter (ASC). Particularly, in Figure 4(c), the negation information can be effectively transferred to the aspect term "*prices*" via the routing algorithm and affects its sentimental polarity. Figure 4(b) and 4(d) are the case of transferring knowledge from ATE to ASC, which suggests that the aspect-related knowledge is mainly

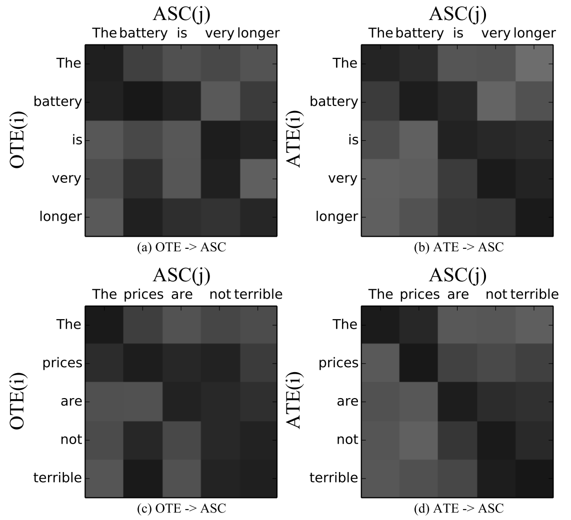

Figure 4: Visualization of $c_{j|i}$. The darker the color is, the more knowledge is transferred.

transferred to corresponding word and points out whether it is a aspect term or not. Therefore, it can help the aspect-level sentiment classification to judge whether the word should own sentimental polarity or not.

5 Related Work

Aspect-based Sentiment Analysis. Existing models typically handle the ABSA task independently or jointly. Apparently, separately treating each subtask can not exploit the inter-task correlations, leading to restricted performances, such as ATE (Qiu et al., 2011; Liu et al., 2013, 2014, 2015; Yin et al., 2016; Li and Lam, 2017; Li et al., 2018; Angelidis and Lapata, 2018; Ma et al., 2019) and ASC (Dong et al., 2014; Nguyen and Shirai, 2015; Vo and Zhang, 2015; Tang et al., 2016; Wang et al., 2016b; De Clercq et al., 2017; Wang et al., 2019; Chen et al., 2017; Ma et al., 2018; Hu et al., 2019a; Bao et al., 2019; Tang et al., 2019; Wang et al., 2019; Sun et al., 2019a; Zhang et al., 2019; Sun et al., 2019b; Luo et al., 2019a; Liang et al., 2019; Hou et al., 2019). By contrast, the joint or integrated methods[7], such as aspect term and opinion term co-extraction (Wang et al., 2016a, 2017; Dai and Song, 2019), aspect term-polarity co-extraction (Mitchell et al., 2013; Zhang et al., 2015; Li and Lu, 2017; Wang et al., 2018; Schmitt et al., 2018; Li et al., 2019a,b; Luo et al., 2019b; He et al., 2019; Hu et al., 2019b) and the three subtasks (Peng et al., 2019), can model the semantic correlations and thus achieve promising results.

However, these methods still fail to fully model the mutual correlations among all the three subtasks. For instance, Peng et al. (2019) use a unified tagging scheme[8] to solve the aspect term-polarity problem and thus can not explicitly model the correlations between them, and they only transfer aspect term knowledge for opinion term extraction. Different from all work above, we focus on exploiting the inter-task correlations among all the three subtasks and thus incrementally boost one another. Besides, we observe the task characteristics and thus propose to use the document-level corpus to discriminately help the corresponding aspect-level tasks, achieving better performances.

Capsule Network. Capsule network (Sabour et al., 2017) has been widely applied in many natural language processing tasks. In the ABSA field, for example, Wang et al. (2019) focus on building multiple capsules for aspect category sentiment analysis, which do not employ the routing procedure. Chen and Qian (2019) construct a transfer capsule network for transferring semantic knowledge from DSC to ASC via sharing encoder, which utilizes the vanilla capsule network only for the ASC subtask. Du et al. (2019) combine capsule network with an interactive attention to model the semantic relationship between the given aspect term and context for the ASC subtask. Jiang et al. (2019) release a new large-scale Multi-Aspect Multi-Sentiment (MAMS) dataset and use capsule network building a strong baseline. Unlike these methods, we focus on the end-to-end ABSA task rather than the individual subtask, and we propose a dynamic-length routing algorithm, which can efficiently perform knowledge transfer.

6 Conclusions

In this work, we propose an iterative knowledge transfer network for the ABSA task, which can fully exploit the inter-task correlations among the three aspect-level subtasks with the proposed routing algorithm. Furthermore, we design a more fine-grained method enabling our model to incorporate the document-level knowledge for discriminately enhancing the corresponding aspect-level

[7]The integrated methods can implicitly model the correlation between tasks through a unified tags and achieve better results than separate ones. But they can not surpass the joint ones because the tasks are linked only by tags, which is insufficient to exploit the correlation (He et al., 2019).

[8]$\{$B,I,E,S$\}$-$\{$neg,pos,neu$\} \cup \{$O$\}$ denote the beginning, the inside, the end and the single of an aspect term with negative, positive, neutral sentiment and other words, respectively.

tasks. Experimental results on three benchmark datasets demonstrate the effectiveness of our approach, which achieves new state-of-the-art results.

Acknowledgements

Liang, Xu and Chen are supported by the National Natural Science Foundation of China (Contract 61370130, 61976015, 61473294 and 61876198), and the Beijing Municipal Natural Science Foundation (Contract 4172047), and the International Science and Technology Cooperation Program of the Ministry of Science and Technology (K11F100010).

References

Stefanos Angelidis and Mirella Lapata. 2018. Summarizing opinions: Aspect extraction meets sentiment prediction and they are both weakly supervised. In *Proceedings of EMNLP*, pages 3675–3686, Brussels, Belgium.

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

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

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

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

Orphée De Clercq, Els Lefever, Gilles Jacobs, Tijl Carpels, and Véronique Hoste. 2017. Towards an integrated pipeline for aspect-based sentiment analysis in various domains. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 136–142, Copenhagen, Denmark. Association for Computational Linguistics.

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 *Proceedings of ACL*, pages 49–54, Baltimore, Maryland.

Chunning Du, Haifeng Sun, Jingyu Wang, Qi Qi, Jianxin Liao, Tong Xu, and Ming Liu. 2019. Capsule network with interactive attention for aspect-level sentiment classification. In *Proceedings of EMNLP*, pages 5492–5501, Hong Kong, China.

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

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

Xiaochen Hou, Jing Huang, Guangtao Wang, Kevin Huang, Xiaodong He, and Bowen Zhou. 2019. Selective attention based graph convolutional networks for aspect-level sentiment classification.

Mengting Hu, Shiwan Zhao, Li Zhang, Keke Cai, Zhong Su, Renhong Cheng, and Xiaowei Shen. 2019a. CAN: Constrained attention networks for multi-aspect sentiment analysis. In *Proceedings of EMNLP*, pages 4593–4602, Hong Kong, China.

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

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of EMNLP*, pages 6280–6285, Hong Kong, China. Association for Computational Linguistics.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In *Proceedings of ACL*, pages 655–665, Baltimore, Maryland.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of EMNLP*, pages 1746–1751.

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

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

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019a. A unified model for opinion target extraction and target sentiment prediction. In *Proceedings of AAAI*, pages 6714–6721.

Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. 2018. Aspect term extraction with history attention and selective transformation. In *Proceedings of IJCAI*, pages 4194–4200.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019b. Exploiting BERT for end-to-end aspect-based sentiment analysis. In *Proceedings of EMNLP*, pages 34–41, Hong Kong, China.

Xin Li and Wai Lam. 2017. Deep multi-task learning for aspect term extraction with memory interaction. In *Proceedings of EMNLP*, pages 2886–2892, Copenhagen, Denmark.

Yunlong Liang, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. A novel aspect-guided deep transition model for aspect based sentiment analysis. In *Proceedings of EMNLP*, pages 5572–5584, Hong Kong, China.

Kang Liu, Liheng Xu, and Jun Zhao. 2013. Syntactic patterns versus word alignment: Extracting opinion targets from online reviews. In *Proceedings of ACL*, pages 1754–1763, Sofia, Bulgaria.

Kang Liu, Liheng Xu, and Jun Zhao. 2014. Extracting opinion targets and opinion words from online reviews with graph co-ranking. In *Proceedings of ACL*, pages 314–324, Baltimore, Maryland.

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of EMNLP*, pages 1433–1443, Lisbon, Portugal.

Huaishao Luo, Tianrui Li, Bing Liu, Bin Wang, and Herwig Unger. 2019a. Improving aspect term extraction with bidirectional dependency tree representation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(7):1201–1212.

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

Dehong Ma, Sujian Li, Fangzhao Wu, Xing Xie, and Houfeng Wang. 2019. Exploring sequence-to-sequence learning in aspect term extraction. In *Proceedings of ACL*, pages 3538–3547, Florence, Italy.

Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *Proceedings of AAAI*.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of EMNLP*, pages 1643–1654, Seattle, Washington, USA.

Thien Hai Nguyen and Kiyoaki Shirai. 2015. PhraseRNN: Phrase recursive neural network for aspect-based sentiment analysis. In *Proceedings of EMNLP*, pages 2509–2514, Lisbon, Portugal.

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. *arXiv preprint arXiv:1911.01616*.

Minlong Peng, Qi Zhang, Yu-gang Jiang, and Xuanjing Huang. 2018. Cross-domain sentiment classification with target domain specific information. In *Proceedings of ACL*, pages 2505–2513, Melbourne, Australia.

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

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of SemEval 2015*, pages 486–495, Denver, Colorado.

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

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.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In *Proceedings of NIPS*, pages 3856–3866.

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 *Proceedings of EMNLP*, pages 1109–1114, Brussels, Belgium.

Chi Sun, Luyao Huang, and Xipeng Qiu. 2019a. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. In *Proceedings of NAACL*, pages 380–385, Minneapolis, Minnesota.

Kai Sun, Richong Zhang, Samuel Mensah, Yongyi Mao, and Xudong Liu. 2019b. Aspect-level sentiment analysis via convolution over dependency tree. In *Proceedings of EMNLP*, pages 5683–5692, Hong Kong, China.

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

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 *Proceedings of ACL*, pages 557–566, Florence, Italy.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Proceedings of IJCAI*.

F. Wang, M. Lan, and W. Wang. 2018. Towards a one-stop solution to both aspect extraction and sentiment analysis tasks with neural multi-task learning. In *Proceedings of IJCNN*, pages 1–8.

Jin Wang, Liang-Chih Yu, K. Robert Lai, and Xuejie Zhang. 2019. Investigating dynamic routing in tree-structured LSTM for sentiment analysis. In *Proceedings of EMNLP*, pages 3423–3428, Hong Kong, China.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016a. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of EMNLP*, pages 616–626, Austin, Texas.

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

X. Wang, G. Xu, J. Zhang, X. Sun, L. Wang, and T. Huang. 2019. Syntax-directed hybrid attention network for aspect-level sentiment analysis. *IEEE Access*, 7:5014–5025.

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

Yequan Wang, Aixin Sun, Minlie Huang, and Xiaoyan Zhu. 2019. Aspect-level sentiment analysis using as-capsules. In *Proceedings of WWW*, pages 2033–2044, New York, NY, USA. ACM.

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Double embeddings and CNN-based sequence labeling for aspect extraction. In *Proceedings of ACL*, pages 592–598, Melbourne, Australia. Association for Computational Linguistics.

Yichun Yin, Furu Wei, Li Dong, Kaimeng Xu, Ming Zhang, and Ming Zhou. 2016. Unsupervised word and dependency path embeddings for aspect term extraction. *CoRR*, abs/1605.07843.

Chen Zhang, Qiuchi Li, and Dawei Song. 2019. Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of EMNLP*, pages 4560–4570, Hong Kong, China.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *Proceedings of EMNLP*, pages 612–621, Lisbon, Portugal.