Transferable End-to-End Aspect-based Sentiment Analysis with Selective Adversarial Learning

Zheng Li[1], **Xin Li**[2], **Ying Wei**[3], **Lidong Bing**[4], **Yu Zhang**[5], **Qiang Yang**[1]
[1]The Hong Kong University of Science and Technology, Hong Kong
[2]The Chinese University of Hong Kong, Hong Kong
[3]Tencent AI Lab, Shenzhen, China
[4]R&D Center Singapore, Machine Intelligence Technology, Alibaba DAMO Academy
[5]Southern University of Science and Technology, Shenzhen, China
{zlict, qyang}@cse.ust.hk, lixin@se.cuhk.edu.hk, judywei@tencent.com
l.bing@alibaba-inc.com, yu.zhang.ust@gmail.com

Abstract

Joint extraction of aspects and sentiments can be effectively formulated as a sequence labeling problem. However, such formulation hinders the effectiveness of supervised methods due to the lack of annotated sequence data in many domains. To address this issue, we firstly explore an unsupervised domain adaptation setting for this task. Prior work can only use common syntactic relations between aspect and opinion words to bridge the domain gaps, which highly relies on external linguistic resources. To resolve it, we propose a novel Selective Adversarial Learning (SAL) method to align the inferred correlation vectors that automatically capture their latent relations. The SAL method can dynamically learn an alignment weight for each word such that more important words can possess higher alignment weights to achieve fine-grained (word-level) adaptation. Empirically, extensive experiments[1] demonstrate the effectiveness of the proposed SAL method.

1 Introduction

End-to-End Aspect-Based Sentiment Analysis (E2E-ABSA) aims to jointly detect the aspect terms explicitly mentioned in sentences and predict the sentiment polarities over them (Liu, 2012; Pontiki et al., 2014). For example, in the sentence "*The **AMD Turin Processor** seems to always perform much better than **Intel***", the user mentions two aspect terms, i.e., "***AMD Turin Processor***" and "***Intel***", and expresses positive and negative sentiments over them, respectively.

Typically, prior work formulates E2E-ABSA as a sequence labeling problem over a unified tagging scheme (Mitchell et al., 2013; Zhang et al., 2015; Li et al., 2019a). The unified tagging scheme connects a set of *aspect boundary tags* (e.g., {`B`, `I`, `E`, `S`, `O`} denotes the beginning of, inside of, end of, single-word, and no aspect term), and *sentiment tags* (e.g. {`POS`, `NEG`, `NEU`} denotes positive, negative or neutral sentiment) together to constitute a joint label space for each word. As such, "***AMD Turin Processor***" and "***Intel***" should be tagged with {`B-POS`, `I-POS`, `E-POS`} and {`S-NEG`}, respectively, while the remaining words are tagged with `O`. This formulation makes two sub-tasks joint modeling easier, and meanwhile, tend to be low-resource. There usually exist few annotated data for each new domain, where labeling each word with a unified tag could be more time-consuming and expensive.

To alleviate the dependence on domain supervisions, we explore an unsupervised domain adaptation setting for E2E-ABSA, which aims to leverage knowledge from a labeled source domain to improve the sequence learning in an unlabeled target domain. The challenges in fulfillment of this setting are two-fold: (1) there exists a large feature distribution shift between domains since aspect terms in different domains are usually disjoint. For example, users usually mention "***pizza***" in the *Restaurant* domain while "***camera***" is often discussed in the *Laptop* domain; (2) Unlike domain adaptation in traditional sentiment classification (Blitzer et al., 2007) that learns shared sentence or document representations, we need to learn fine-grained (word-level) representations to be domain-invariant for sequence prediction.

Consider the first problem, i.e., what to transfer? Even though aspect terms from different domains behave distinctly, some association patterns between aspect and opinion words are common across domains; e.g., "*The **pizza** is great.*" from the *Restaurant* domain and "*The **camera** is excellent.*" from the *Laptop* domain. Both of them share the same syntactic pattern (*as-*

[1]The code is available at https://github.com/hsqmlzno1/Transferable-E2E-ABSA

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 4589–4599, Hong Kong, China, November 3–7, 2019.

pect words→nsubj→opinion words). Inspired by this, existing studies use general syntactic relations as the pivot to bridge the domain gaps for cross-domain aspect extraction (Jakob and Gurevych, 2010; Ding et al., 2017), or aspect and opinion co-extraction (Li et al., 2012; Wang and Pan, 2018). Unfortunately, these methods highly rely on prior knowledge (e.g., manually-designed rules) or external linguistic resources (e.g., dependency parsers), which are inflexible and prone to bringing in knowledge errors. Instead, we introduce a multi-hop Dual Memory Interaction (DMI) mechanism to automatically capture the latent relations among aspect and opinion words. The DMI iteratively infers the correlation vectors of each word by interacting its local memory (LSTM hidden state) with both the global aspect and opinion memories, such that the inter-correlations between aspects and opinions, and the intra-correlations in aspects or opinions can be derived.

Second, how to transfer for this sequence prediction task? One straightforward way is to apply domain adaption methods to align all words within the sentence, however, it is observed that it will not yield significant improvements. Actually, not all the words contribute equally to the domain-invariant feature space though fine-grained adaptation is required. Thus, we propose a novel Selective Adversarial Learning (SAL) method to dynamically learn an alignment weight for each word, where more important words can possess higher alignment weights to achieve a local semantic alignment based on adversarial training. Empirically, the proposed model outperforms the state-of-the-art fine-grained adaptation methods by a large margin on four benchmark datasets. We also conduct extensive ablation studies to quantitatively and qualitatively demonstrate the effectiveness of the selectivity of adversarial learning.

Overall, our main contributions are summarized as: (1) to the best of our knowledge, an unsupervised domain adaptation setting is firstly explored for E2E-ABSA; (2) an effective SAL method is proposed to conduct a local semantic alignment for fine-grained domain adaptation; (3) extensive experiments verify the effectiveness of the proposed SAL method.

2 Task Definition

Single-domain: E2E-ABSA involves both aspect detection (**AD**) and aspect sentiment (**AS**) classification tasks, which are formulated as a unified sequence labeling problem. Given an input sequence of words $\mathbf{x}=\{w_1, w_2, ..., w_T\}$ and its word embeddings $\mathbf{e}=\{e_1, e_2, ..., e_T\}$, the goal is to predict a tag sequence $\mathbf{y}=\{y_1, y_2, ..., y_T\}$ over the *unified tags*, with $y_i \in \mathcal{Y}^{\mathcal{U}} = \{$`B-POS`, `I-POS`, `E-POS`, `S-POS`, `B-NEG`, `I-NEG`, `E-NEG`, `S-NEG`, `B-NEU`, `I-NEU`, `E-NEU`, `S-NEU`, `O`$\}$. **Cross-domain:** Here we are performing in a more challenging unsupervised domain adaptation setting. Given a set of labeled data $D_s = \{(\mathbf{x}_s^i, \mathbf{y}_s^i)\}_{i=1}^{N_s}$ from a source domain and a set of unlabeled data $D_t = \{(\mathbf{x}_t^j)\}_{j=1}^{N_t}$ from a target domain, we aim to transfer the knowledge of D_s to improve the sequence learning in D_t.

3 Model Description

Overview: As shown in Figure 1, we adopt two stacked bi-directional LSTMs as the base model (Li et al., 2019a) for E2E-ABSA. The upper LSTM$^{\mathcal{U}}$ is for the high-level **ADS** (AD+AS) task and it predicts the *unified tags* as output, while the lower LSTM$^{\mathcal{B}}$ is for the low-level **AD** task and predicts the *aspect boundary tags* as the guidance. To adapt the base model, we design different components in terms of the two problems, i.e., what to transfer and how to transfer, respectively.

(1) To automatically capture the latent relations between aspect and opinion words as transferable knowledge across domains, we introduce a multi-hop Dual Memory Interaction (DMI) mechanism between the two LSTMs. At each hop, e.g., the 1st hop, each local memory $\mathbf{h}_i^{\mathcal{B}}$ will interact with both the global aspect and opinion memories, i.e., $\mathbf{m}_a^1$ and $\mathbf{m}_o^1$ based on the DMI, to produce two correlation vectors for aspect and opinion words co-detection, where the opinion detection is used as an auxiliary task for the AD task. The "local" memory denotes the hidden representation (LSTM$^{\mathcal{B}}$ hidden state) of each word within the sentence. Whereas the two "global" memories are globally shared by all input sentences, which are commonly used in memory networks (Sukhbaatar et al., 2015; Kumar et al., 2016) and can be seen as high-level representations for aspect and opinion words, respectively. The A-attention and O-attention then aggregate most relevant aspect or opinion words information to refine the two global memories for the next hop.

(2) To adapt these relations across domains, we propose a Selective Adversarial Learning (SAL)

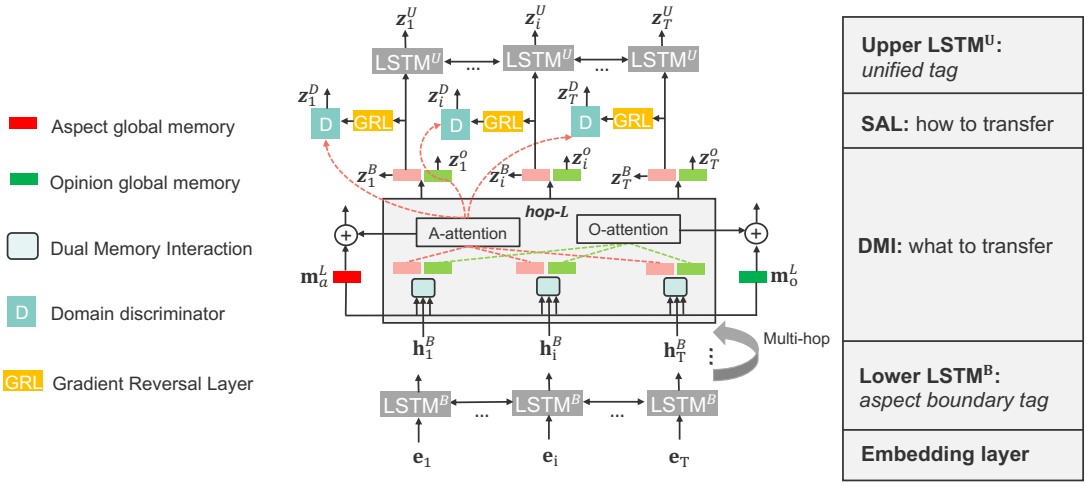

Figure 1: The framework of the proposed model.

method to dynamically focus on aligning the aspect words between domains. This is because the informative aspect words contribute more to the shared feature space than the unmeaning words tagged with O in the sentence (Zhou et al., 2019b). As such, an aspect tagger trained on a source domain can work well when applied to a target domain. Specifically, at the final hop, we adopt a domain discriminator for each word with a gradient reversal layer (Ganin et al., 2016) to perform domain adversarial learning over its correlation vector (alignment). While the A-attention module provides an aspect attention distribution as a selector to control a learnable alignment weight for each word (selectivity). Finally, each aligned correlation vector will be used to predict *aspect boundary tags* (AD task) and fed to the LSTM$^{\mathcal{U}}$ for the *unified tags* prediction (ADS task). In the following sections, we detail each component.

3.1 Base Model

We adopt two stacked bi-directional LSTMs as the base model. We link these two LSTM layers so that the hidden representations generated by the LSTM$^{\mathcal{B}}$ can be fed to LSTM$^{\mathcal{U}}$ as the guidance information. Specifically, their hidden representations $\mathbf{h}_i^{\mathcal{B}} \in \mathbb{R}^{\dim_h^{\mathcal{B}}}$ and $\mathbf{h}_i^{\mathcal{U}} \in \mathbb{R}^{\dim_h^{\mathcal{U}}}$ at the i-th time step ($i \in [1, T]$) are calculated as follows:

$$\mathbf{h}_i^{\mathcal{B}} = [\overrightarrow{\text{LSTM}}^{\mathcal{B}}(\mathbf{e}_i); \overleftarrow{\text{LSTM}}^{\mathcal{B}}(\mathbf{e}_i)],$$
$$\mathbf{h}_i^{\mathcal{U}} = [\overrightarrow{\text{LSTM}}^{\mathcal{U}}(\mathbf{h}_i^{\mathcal{B}}); \overleftarrow{\text{LSTM}}^{\mathcal{U}}(\mathbf{h}_i^{\mathcal{B}})].$$

The probability scores $\mathbf{z}_i^{\mathcal{B}} \in \mathbb{R}^{|\mathcal{Y}^{\mathcal{B}}|}$ over the *aspect boundary tags* $\mathcal{Y}^{\mathcal{B}} = \{\texttt{B}, \texttt{I}, \texttt{E}, \texttt{S}, \texttt{O}\}$ are calculated by a fully-connected softmax layer:

$$\mathbf{z}_i^{\mathcal{B}} = \mathbf{p}(\mathbf{y}_i^{\mathcal{B}}|\mathbf{h}_i^{\mathcal{B}}) = \text{Softmax}(\mathbf{W}_{\mathcal{B}}\mathbf{h}_i^{\mathcal{B}} + \mathbf{b}_{\mathcal{B}}).$$

Similarly, the scores $\mathbf{z}_i^{\mathcal{U}} \in \mathbb{R}^{|\mathcal{Y}^{\mathcal{U}}|}$ over the *unified tags* $\mathcal{Y}^{\mathcal{U}}$ defined in Section 2 are obtained as:

$$\mathbf{z}_i^{\mathcal{U}} = \mathbf{p}(\mathbf{y}_i^{\mathcal{U}}|\mathbf{h}_i^{\mathcal{U}}) = \text{Softmax}(\mathbf{W}_{\mathcal{U}}\mathbf{h}_i^{\mathcal{U}} + \mathbf{b}_{\mathcal{U}}).$$

3.2 Global-Local Memory Interaction

Before detailing the DMI module, we firstly introduce Global-Local Memory Interaction (GLMI) that describes the interaction between a local memory $\mathbf{h}_i \in \mathbb{R}^{\dim_h}$ and a global memory $\mathbf{m} \in \mathbb{R}^{\dim_h}$. Formally, we parameterize the GLMI $f(\mathbf{h}_i, \mathbf{m}; \mathbf{\Theta}, \mathbf{G})$, with $\mathbf{\Theta} = \{\mathbf{W}, \mathbf{b}\}$ and $\mathbf{G}$, which consists of a residual transformation and a tensor product operation. Specifically, we firstly incorporate the global memory information $\mathbf{m}$ into each local position with a residual transformation as $\tilde{\mathbf{h}}_i = \mathbf{h}_i + \text{ReLU}(\mathbf{W}[\mathbf{h_i}:\mathbf{m}] + \mathbf{b})$, where $[:]$ denotes the vector concatenation. As such, the global memory can distill more correlated local information and they are mapped into the same space. Then we compute a correlation vector $\mathbf{r}_i \in \mathbb{R}^K$ that encodes the strength of correlations between the global memory $\mathbf{m}$ and the transformed local memory $\tilde{\mathbf{h}}_i$ through a tensor product operation as:

$$\mathbf{r}_i = \mathbf{m}^T \mathbf{G} \tilde{\mathbf{h}}_i^{\mathcal{B}},$$

where the tensor $\mathbf{G} \in \mathbb{R}^{\dim_h \times \dim_h \times K}$ can be seen as multiple bilinear matrice that model K kinds of latent relations between two objects. The k-th slice of the $\mathbf{G}$, i.e., $\mathbf{G}_k \in \mathbb{R}^{\dim_h \times \dim_h}$ denotes one type of latent relation that interacts with 2 vectors to constitute one type of composition.

3.3 Dual Memory Interaction

Following the notations in Section 3.2, we further define a global aspect memory $\mathbf{m}_a \in \mathbb{R}^{\dim_h^{\mathcal{B}}}$, a

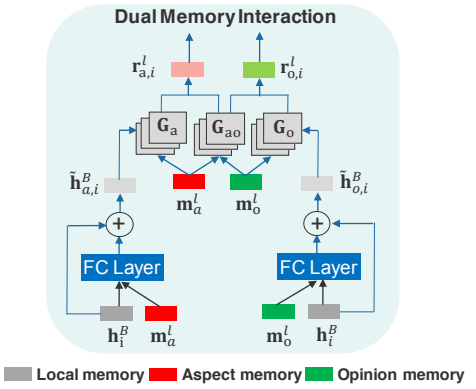

Figure 2: The Dual Memory Interaction (DMI).

global opinion memory $\mathbf{m}_o \in \mathbb{R}^{\dim_h^{\mathcal{B}}}$, and LSTM$^{\mathcal{B}}$ hidden states $\mathbf{H}^{\mathcal{B}} = \{\mathbf{h}_i^{\mathcal{B}}\}_{i=1}^{T}$ as the local memories. The global aspect and opinion memories are able to capture highly correlated aspect or opinion words from the local memories, respectively. Based on the observation that aspect words are often collocated with opinion words across domains, thus their associations can act as the pivot information to bridge the domain gaps. To automatically capture their latent relations within the sentences, at the l-th hop, each local memory $\mathbf{h}_i^{\mathcal{B}}$ will interact with the global memories $\mathbf{m}_a^l$ and $\mathbf{m}_o^l$ by the Dual Memory Interaction (DMI) shown in Figure 2, to produce two correlation vectors for aspect and opinion co-detection:

$$\mathbf{r}_{a,i}^l = [f(\mathbf{h}_i^{\mathcal{B}}, \mathbf{m}_a^l; \mathbf{\Theta}_a, \mathbf{G}_a) : f(\mathbf{h}_i^{\mathcal{B}}, \mathbf{m}_o^l; \mathbf{\Theta}_o, \mathbf{G}_{ao})],$$
$$\mathbf{r}_{o,i}^l = [f(\mathbf{h}_i^{\mathcal{B}}, \mathbf{m}_o^l; \mathbf{\Theta}_o, \mathbf{G}_o) : f(\mathbf{h}_i^{\mathcal{B}}, \mathbf{m}_a^l; \mathbf{\Theta}_a, \mathbf{G}_{ao}^{\top})],$$

where $\mathbf{G}_a, \mathbf{G}_o$ and $\mathbf{G}_{ao}$ denote the composition tensors of modeling the latent relations between aspect and aspect, opinion and opinion, and aspect and opinion, respectively.

The correlation vector measures the association strength between local and global memories; e.g., If $\mathbf{h}_i^{\mathcal{B}}$ for the word w_i is both highly intra-correlated with the aspect memory $\mathbf{m}_a$ and inter-correlated with the opinion memory $\mathbf{m}_o$, w_i is more likely to be an aspect term. Then the two correlation vectors can be transformed to a scalar **a**spect attention (**A**-attention) and **o**pinion attention (**O**-attention) weight $\alpha_{p,i}^l$, respectively, with $p \in \{a, o\}$ denoting the aspect or opinion, which indicates the possibility of each word in the sentence being an aspect word or an opinion word as:

$$\alpha_{p,i}^l = \frac{\exp(\mathbf{W}_p \mathbf{r}_{p,i}^l)}{\sum_{j=1}^{T} \exp(\mathbf{W}_p \mathbf{r}_{p,j}^l)},$$

where $\mathbf{W}_p$ is the weight of the attention module. The aspect or opinion attention weight $\alpha_{p,i}^l$ will summarize the local memories to update the global aspect and opinion memories, respectively, for the next hop, i.e., $\mathbf{m}_p^{l+1} = \mathbf{m}_p^l + \sum_{i=1}^{T} \alpha_{p,i}^l \mathbf{h}_i^{\mathcal{B}}$. The updates gradually refine the global memories to incorporate more relevant candidates based on the attention mechanism. In the DMI, all parameters are shared in different hops and domains.

At the final L-th hop, we use $\mathbf{r}_{a,i}^L$ for the AD task and feed it to the LSTM$^{\mathcal{U}}$ for the ADS task. For the auxiliary opinion detection task, we feed $\mathbf{r}_{o,i}^L$ into a softmax layer for predicting the probability scores $\mathbf{z}_i^{\mathcal{O}} \in \mathbb{R}^{|\mathcal{Y}^{\mathcal{O}}|}$ over the opinion labels[2] $\mathcal{Y}^{\mathcal{O}}$, i.e., a word is an opinion word or not, as:

$$\mathbf{z}_i^{\mathcal{O}} = \mathbf{p}(\mathbf{y}_i^{\mathcal{O}} | \mathbf{r}_{o,i}^L) = \mathrm{Softmax}(\mathbf{W}_{\mathcal{O}} \mathbf{r}_{o,i}^L + \mathbf{b}_{\mathcal{O}}).$$

3.4 Selective Adversarial Learning

To adapt the captured relations to be domain-invariant, we propose a Selective Adversarial Learning (SAL) method to dynamically align the words with high probability to fall into the aspect boundaries, i.e., being an aspect word. Specifically, we introduce a domain discriminator for each word, which aims to identify the domain label $\mathbf{y}_i^{\mathcal{D}} \in \mathbb{R}^{|\mathcal{Y}^{\mathcal{D}}|}$ of the input word, i.e., the word in the sentence is from the source or the target domain. While the feature extractor is to produce the domain-invariant correlation vector $\mathbf{r}_{a,i}^L$ that cannot be distinguished by the domain discriminator via a Gradient Reversal Layer (GRL) (Ganin et al., 2016). Mathematically, we formulate the GRL as a 'pseudo-function' $R_\lambda(\mathbf{x}) = \mathbf{x}$ with a reversal gradient $\frac{\partial R_\lambda(\mathbf{x})}{\partial \mathbf{x}} = -\lambda I$, where λ is the adaptation rate. The correlation vector $\mathbf{r}_{a,i}^L$ will be fed to the GRL before the domain discriminator, which is used to predict the probability scores $\mathbf{z}_i^{\mathcal{D}} \in \mathbb{R}^{|\mathcal{Y}^{\mathcal{D}}|}$ over the domain labels $\mathcal{Y}^{\mathcal{D}}$ as:

$$\mathbf{z}_i^{\mathcal{D}} = \mathbf{p}(\mathbf{y}_i^{\mathcal{D}} | \mathbf{r}_{a,i}^L) = \mathrm{Softmax}(\mathbf{W}_{\mathcal{D}} R_\lambda(\mathbf{r}_{a,i}^L) + \mathbf{b}_{\mathcal{D}}).$$

And meanwhile, the aspect attention weight $\alpha_{a,i}^L$ at the final hop serves as a selector to be a learnable alignment weight for each word. Thus, the selective domain adversarial loss is a weighted cross-entropy loss ℓ for all the words from the labeled source data D_s and unlabeled target data D_t:

$$\mathcal{L}_{\mathcal{D}} = \sum_{D_s \cup D_t} \sum_{i=1}^{T} \alpha_{a,i}^L \ell(\mathbf{z}_i^{\mathcal{D}}, \mathbf{y}_i^{\mathcal{D}}). \quad (1)$$

[2]The opinion lexicon (http://mpqa.cs.pitt.edu/) is used to provide opinion labels for both domains.

Existing studies (Yosinski et al., 2014; Mou et al., 2016) have already shown some evidence that low-level neural layer features (i.e., low-level task) are more easily transferred to different tasks or domains. Thus, we choose the $\mathbf{r}_{a,i}^{L}$ from the low-level AD task to be aligned instead of the feature $\mathbf{h}_i^{\mathcal{U}}$ from the high-level ADS task to transfer. Our ablation studies also confirm this assumption.

3.5 Alternating Training

The primary task loss consists of the cross-entropy losses ℓ for both the guided AD and main ADS tasks for the labeled source data D_s:

$$\mathcal{L}_{\mathcal{M}} = \sum_{D_s} \sum_{\mathcal{Q}\in\{\mathcal{B},\mathcal{U}\}} \sum_{i=1}^{T} \ell(\mathbf{z}_i^{\mathcal{Q}}, \mathbf{y}_i^{\mathcal{Q}}). \quad (2)$$

The auxiliary opinion detection loss is the cross-entropy loss for the labeled source data D_s and unlabeled target data D_t as follows:

$$\mathcal{L}_{\mathcal{O}} = \sum_{D_s \cup D_t} \sum_{i=1}^{T} \ell(\mathbf{z}_i^{\mathcal{O}}, \mathbf{y}_i^{\mathcal{O}}). \quad (3)$$

Traditionally, we can directly optimize the joint loss of Eqs. (1)-(3), i.e., $E = \mathcal{L}_{\mathcal{M}} + \rho\mathcal{L}_{\mathcal{O}} + \gamma\mathcal{L}_{\mathcal{D}}$ to obtain both discriminative and domain-invariant word representations, where ρ and γ are the trade-off factors. However, we found the optimization process tends to be unstable since it may be hard to jointly optimize many objectives. Thus, we propose an empirically alternating strategy to train the $\mathcal{L}_{\mathcal{M}} + \rho\mathcal{L}_{\mathcal{O}}$ and $\mathcal{L}_{\mathcal{D}}$ iteratively, which separates the whole word representation learning into a *discriminative* stage and a *domain-invariant* stage. Let $\boldsymbol{\theta}_f$, $\boldsymbol{\theta}_w$, $\boldsymbol{\theta}_d$ denote the parameters for feature learning of each word, word predictors for AD, ADS and opinion detection tasks, and domain discriminators, respectively. Based on our strategy, we are seeking the parameters $(\hat{\boldsymbol{\theta}}_f^{(1)}, \hat{\boldsymbol{\theta}}_f^{(2)}, \hat{\boldsymbol{\theta}}_w, \hat{\boldsymbol{\theta}}_d)$ that deliver a saddle point of E among two stages:

$$(\hat{\boldsymbol{\theta}}_f^{(1)}, \hat{\boldsymbol{\theta}}_w) = \arg\min_{\boldsymbol{\theta}_f, \boldsymbol{\theta}_w} \mathcal{L}_{\mathcal{M}} + \rho\mathcal{L}_{\mathcal{O}}$$

$$(\hat{\boldsymbol{\theta}}_f^{(2)}, \hat{\boldsymbol{\theta}}_d) = \arg\min_{\boldsymbol{\theta}_d} \max_{\boldsymbol{\theta}_f} \mathcal{L}_{\mathcal{D}}.$$

At the saddle point, the feature learning parameters $\boldsymbol{\theta}_f$ minimize the word prediction losses (i.e., the features are discriminative) for the first stage. For the second stage, the domain classification loss is minimized by the domain discriminator parameters $\boldsymbol{\theta}_d$ while maximized by the feature learning parameters $\boldsymbol{\theta}_f$ via GRL (i.e., the features are domain-invariant). As such, we can achieve easier and more stable optimization for feature learning.

Dataset	Domain	Sentences	Training	Testing
$\mathbb{L}$	Laptop	1,869	1,458	411
$\mathbb{R}$	Restaurant	3,900	2,481	1,419
$\mathbb{D}$	Device	1,437	954	483
$\mathbb{S}$	Service	2,153	1,433	720

Table 1: Statistics of the datasets.

4 Experiments

4.1 Experimental Setup

Datasets: Our experiments are conducted on four benchmark datasets: Laptop ($\mathbb{L}$), Restaurant ($\mathbb{R}$), Device ($\mathbb{D}$), and Service ($\mathbb{S}$). $\mathbb{L}$ contains reviews from the laptop domain in SemEval ABSA challenge 2014 (Pontiki et al., 2014). Following the setup in (Li et al., 2019a), $\mathbb{R}$ is the union set of the restaurant datasets from SemEval ABSA challenge 2014, 2015, and 2016 (Pontiki et al., 2014, 2015, 2016). $\mathbb{D}$ is a combination of device reviews from 5 different digital products provided by (Hu and Liu, 2004). $\mathbb{S}$ is introduced by (Toprak et al., 2010) and contains reviews from web services. Detailed statistics are shown in Table 1.

Settings: We construct 10 transfer pairs like $D_s \rightarrow D_t$ with the four domains mentioned above, and we do not use the pairs $\mathbb{L}\rightarrow\mathbb{D}$ and $\mathbb{D}\rightarrow\mathbb{L}$ as these two domains are very similar. Note that for the unsupervised domain adaptation setting, no labels are available for the target domain. Therefore, for each transfer pair, its training dataset is the combination of the labeled training data of the source domain and the unlabeled training data of the target domain. Meanwhile, it employs the testing data of the source domain with labels as the validation set and the testing data of the target domain as the evaluation set. We report the results for both AD and ADS tasks. The evaluation metric is the Micro-F1 score under the **exact** match, which means that an output segment is considered to be correct only if it exactly matches with the gold standard span of the aspect term for the AD task or the aspect term and its sentiment for the ADS task. All experiments are repeated 5 times and we report the average results over 5 runs.

4.2 Implementation details

The word embeddings are 100-dimensional *word2vec* (Mikolov et al., 2013) vectors pre-trained on the combination of the Yelp Challenge

Transfer Pair	TCRF		RAP		Hier-Joint		Hier-Joint$^+$		RNSCN		RNSCN$^+$		**Ours**	
	AD	ADS	AD	ADS	AD	ADS	AD	ADS	AD	ADS	AD	ADS	AD	ADS
$\mathbb{S}\rightarrow\mathbb{R}$	-	14.84	-	25.41	-	32.81	46.39	31.10	-	30.56	48.89	33.21	**52.05**	**41.03**
$\mathbb{L}\rightarrow\mathbb{R}$	-	16.06	-	31.05	-	31.90	48.61	33.54	-	31.85	52.19	35.65	**56.12**	**43.04**
$\mathbb{D}\rightarrow\mathbb{R}$	-	17.05	-	28.37	-	30.03	42.96	32.87	-	31.41	50.39	34.60	**51.55**	**41.01**
$\mathbb{R}\rightarrow\mathbb{S}$	-	15.20	-	13.17	-	15.20	27.18	15.56	-	23.31	30.41	20.04	**39.02**	**28.01**
$\mathbb{L}\rightarrow\mathbb{S}$	-	12.34	-	13.72	-	15.33	25.22	13.90	-	16.73	31.21	16.59	**38.26**	**27.20**
$\mathbb{D}\rightarrow\mathbb{S}$	-	13.49	-	16.80	-	18.74	29.28	19.04	-	18.93	35.50	20.03	**36.11**	**26.62**
$\mathbb{R}\rightarrow\mathbb{L}$	-	14.59	-	15.69	-	19.17	34.11	20.72	-	25.54	**47.23**	26.63	45.01	**34.13**
$\mathbb{S}\rightarrow\mathbb{L}$	-	9.56	-	12.38	-	21.80	33.02	22.65	-	19.15	34.03	18.87	**35.99**	**27.04**
$\mathbb{R}\rightarrow\mathbb{D}$	-	19.84	-	17.50	-	22.91	34.81	24.53	-	32.43	**46.16**	33.26	43.76	**35.44**
$\mathbb{S}\rightarrow\mathbb{D}$	-	13.43	-	15.74	-	20.04	35.00	23.24	-	19.98	32.41	22.00	**41.21**	**33.56**
Average	-	14.64	-	18.98	-	22.79	35.66	23.72	-	24.99	40.84	26.09	**43.91**†	**33.71**†
(Δ)	-	(19.07)	-	(14.73)	-	(10.92)	(8.25)	(9.99)	-	(8.72)	(3.07)	(7.62)	-	-

Table 2: Main results (%). Δ refers to the improvements of the full model over baseline methods. The marker † means that our model significantly outperforms the best baseline **RNSCN**$^+$ with p-value < 0.01.

dataset[3] and the electronics dataset from Amazon reviews[4]. For out-of-vocabulary words, we randomly initialized them with a uniform distribution $U(-0.25, 0.25)$. The dimensions of two LSTM layers $\dim_h^{\mathcal{B}}$ and $\dim_h^{\mathcal{U}}$ are all set to 100. The number of hops L is set to 2. The number of bilinear interactions K is set to 50. The weight matrices are initialized with a uniform distribution $U(-0.2, 0.2)$. The adaptation rate λ is 0.1 and the trade-off factor ρ is 1.0. For training, the model is optimized by the Adam algorithm (Kingma and Ba, 2014) with the initial learning rate 0.001. The batch size is 64, with a half coming from the source and target domains, respectively. Gradients with the ℓ_2 norm larger than 40 are normalized to be 40. To alleviate the overfitting, we apply the dropout on the word embeddings $\mathbf{e}_i$ and the learned word representations $\mathbf{r}_{a,i}^l$, $\mathbf{r}_{o,i}^l$, and $\mathbf{h}_i^{\mathcal{U}}$ with dropout rate 0.5. We use the same hyper-parameters, which are tuned on 10% randomly held-out training data of the source domain in $\mathbb{R}\rightarrow\mathbb{L}$, for all transfer pairs.

4.3 Baselines

We compare with several state-of-the-art fine-grained adaptation methods.

- **TCRF** (Jakob and Gurevych, 2010): Transferable CRF that uses a linear-chain CRF for sequence prediction based on shared non-lexical features across domains, e.g., POS tags and dependency relations.
- **RAP** (Li et al., 2012): A cross-domain Relational Adaptive Bootstrapping method that iteratively expands target aspect and opinion lexicons according to common opinion words and syntactic relations.
- **Hier-Joint** (Ding et al., 2017): A recurrent neural network (RNN) with manually designed rule-based auxiliary tasks based on common syntactic relations among aspect and opinion words.
- **RNSCN** (Wang and Pan, 2018): a recursive neural structural correspondence network that incorporates syntactic structures and exploits an auto-encoder to denoise relation labels generated from the parser.

As the first to address cross-domain E2E-ABSA, we have to adapt all the baselines which are originally proposed for cross-domain aspect detection, or aspect and opinion co-detection to return the ADS results by replacing their *aspect boundary tags* with the *unified tags*. Absent of the proposed stacking architecture, all the baselines fail to accomplish the auxiliary AD task meantime. Thus, we only report their ADS results. Besides, E2E-ABSA aims to simultaneously learn aspect terms along with their sentiments. Thus, the ADS is exactly our main task while the AD is only an auxiliary task used for the guidance.

To be more convincing, we extend neural models (i.e., Hier-Joint and RNSCN) to more powerful baselines named **Hier-Joint**$^+$ and **RNSCN**$^+$ with the proposed stacking architecture, respectively. Both of them stack an additional RNN layer on top of the original framework to produce the *unified tags* while the lower layer is to predict the *aspect boundary tags* as the guidance. The validity of such extensions is guaranteed by the fact that the extended versions achieve even better AD performances than the original versions. We use the source code of the baselines for experiments. For

[3] http://www.yelp.com/datasetchallenge

[4] http://jmcauley.ucsd.edu/data/amazon/links.html

Transfer Pair	Lower bound		Ablation Models						Full Model		Upper bound	
	Base Model (SO)		Base Model+DMI		AD-AL		ADS-SAL		AD-SAL		Base Model (TO)	
	AD	ADS	AD	ADS	AD	ADS	AD	ADS	AD	ADS	AD	ADS
$\mathbb{S}\rightarrow\mathbb{R}$	30.32	19.74	45.68	37.10	48.28	37.65	51.29	41.03	**52.05**	**41.03**	81.84	67.26
$\mathbb{L}\rightarrow\mathbb{R}$	33.99	28.34	46.25	36.49	51.79	38.63	55.50	42.00	**56.12**	**43.04**		
$\mathbb{D}\rightarrow\mathbb{R}$	31.59	27.25	46.56	36.89	46.39	37.34	46.43	38.35	**51.55**	**41.01**		
$\mathbb{R}\rightarrow\mathbb{S}$	15.63	8.61	21.88	16.85	25.13	18.61	37.11	25.84	**39.02**	**28.01**	68.28	41.12
$\mathbb{L}\rightarrow\mathbb{S}$	22.45	16.07	28.67	21.53	28.18	20.74	30.35	23.73	**38.26**	**27.20**		
$\mathbb{D}\rightarrow\mathbb{S}$	16.79	9.49	31.91	22.14	32.88	24.89	32.51	21.45	**36.11**	**26.62**		
$\mathbb{R}\rightarrow\mathbb{L}$	38.45	23.40	42.27	30.52	40.52	28.77	44.56	33.34	**45.01**	**34.13**	75.95	52.62
$\mathbb{S}\rightarrow\mathbb{L}$	24.69	14.48	36.38	27.48	32.96	25.16	33.87	24.22	**35.99**	**27.04**		
$\mathbb{R}\rightarrow\mathbb{D}$	34.87	25.79	36.90	27.71	41.61	31.88	**43.97**	34.50	43.76	**35.44**	70.37	57.62
$\mathbb{S}\rightarrow\mathbb{D}$	27.73	17.73	38.03	31.21	39.54	32.28	40.40	33.26	**41.21**	**33.56**		
Average	27.65	19.09	37.45	28.79	38.73	29.60	41.60	31.77	**43.91**†	**33.71**†	74.11	54.66
(Δ)	(16.26)	(14.62)	(6.46)	(4.92)	(5.18)	(4.11)	(2.31)	(1.94)	-	-	-	-

Table 3: Ablation results (%). Δ refers to the improvements of the full model over ablation methods. The marker † means that the full model significantly outperforms the best ablation model **ADS-SAL** with p-value < 0.01.

fair comparison, all baselines use the same pre-trained word embeddings and the baselines that require opinion labels use the same opinion lexicon.

4.4 Main Results

Based on the results in Table 2, we have the following observations:

- Our model consistently and significantly achieves the best results on almost all transfer pairs, outperforming the strongest baseline RNSCN^{+} by 3.07% and 7.62% Micro-F1 on average for the AD and ADS tasks, respectively. Our model can automatically model complicated relations among aspect and opinion words via the DMI as transferable knowledge. Besides, the proposed SAL method can dynamically learn an alignment weight for each word to achieve a local semantic alignment, which distills a better shared feature space and further improves the performances.

- Traditional non-neural methods like TCRF and RAP perform very poorly due to the reliance on hand-crafted features. Our model outperforms Hier-Joint and RNSCN, which are neural models, by 10.92% and 8.72% Micro-F1 on average for the ADS task, respectively. Both of them rely on the dependency parser to exploit syntactic relations, which are inflexible due to the no end-to-end manner and may bring in external errors.

- The extended version Hier-Joint^{+} and RNSCN^{+} can further improve the performances, which shows the benefits of the guidance from the low-level AD task. However, our model can still outperform them by a large margin, which demonstrates the effectiveness of the proposed methods.

4.5 Ablation Study

To investigate the effectiveness of each component, we conduct the ablation study to compare our full model with different ablation variants:

- **Base Model (SO / TO)**: it uses two stacked Bi-LSTMs as the Base Model. SO (Source Only) and TO (Target Only) denote that the base model is only trained on the labeled data from the source and target domain, respectively. We usually refer to them as a lower bound and a upper bound, respectively.

- **Base Model+DMI**: it uses two stacked Bi-LSTMs with a multi-hop dual memory interaction (DMI) between them.

- **AD-AL**: it performs pure adversarial learning (removing the selective weight $\alpha^{L}_{a,i}$ from the Eq. (1)) on each correlation vector $\mathbf{r}^{L}_{a,i}$ for the low-level AD task.

- **AD-SAL**: it advances the AD-AL by conducting selective adversarial learning.

- **ADS-SAL**: it conducts selective adversarial learning on each word representation $\mathbf{h}^{\mathcal{U}}_{i}$ for the high-level ADS task.

Note that, the AD-AL, AD-SAL (**Full model**) and ADS-SAL all use the same architecture as the Base Model+DMI. Based on the Table 3, we have the following observations to give us evidences:

- **No DMI** v.s. **DMI**: Base Model+DMI outperforms the Base Model (SO) by 9.80% and 9.70% Micro-F1 on average for the AD and ADS tasks, respectively. This demonstrates

Input: (Target domain $\mathbb{L}$)	Base model+DMI		AD-AL		AD-SAL	
	AD	ADS	AD	ADS	AD	ADS
1. This laptop has only 2 [*usb ports*]$_{\text{NEG}}$, and they are both on the same side .	*ports*(✗), *side* (✗)	NONE(✗)	NONE(✗)	NONE(✗)	*usb ports*	[*usb ports*]$_{\text{NEG}}$
2. It is very easy to integrate [*bluetooth devices*]$_{\text{POS}}$, and [*usb devices*]$_{\text{POS}}$ are recognized almost instantly .	*devices* (✗), *devices* (✗)	[*devices*]$_{\text{POS}}$ (✗), [*devices*]$_{\text{POS}}$ (✗)	NONE(✗)	NONE(✗)	*bluetooth devices*, *usb devices*	[*bluetooth devices*]$_{\text{POS}}$, [*usb devices*]$_{\text{POS}}$
3. I also wanted [*windows 7*]$_{\text{POS}}$, which this one has .	NONE(✗)	NONE(✗)	NONE(✗)	NONE(✗)	*windows 7*	[*windows 7*]$_{\text{POS}}$
4. The [*speed*]$_{\text{POS}}$, the [*simplicity*]$_{\text{POS}}$, the [*design*]$_{\text{POS}}$ it is lightyears ahead of any pc i have ever owned .	*speed*, *design*	[*speed*]$_{\text{POS}}$, [*design*]$_{\text{POS}}$	*speed*, *design*, *pc* (✗)	[*speed*]$_{\text{POS}}$, [*design*]$_{\text{POS}}$, [*pc*]$_{\text{POS}}$ (✗)	*speed*, *design*, *simplicity*	[*speed*]$_{\text{POS}}$, [*design*]$_{\text{POS}}$, [*simplicity*]$_{\text{POS}}$
6. The [*battery life*]$_{\text{POS}}$ is excellent , the [*display*]$_{\text{POS}}$ is excellent and [*downloading apps*]$_{\text{POS}}$ is a breeze .	*battery* (✗), *display*, *apps* (✗)	[*battery*]$_{\text{POS}}$ (✗), [*display*]$_{\text{POS}}$, [*apps*]$_{\text{POS}}$ (✗)	*battery* (✗), *display*, *apps* (✗)	[*battery*]$_{\text{POS}}$ (✗), [*display*]$_{\text{POS}}$, [*apps*]$_{\text{POS}}$ (✗)	*battery life*, *display*, *downloading apps*	[*battery life*]$_{\text{POS}}$, [*display*]$_{\text{POS}}$, [*downloading apps*]$_{\text{POS}}$

Table 4: Case analysis for the $\mathbb{R}\rightarrow\mathbb{L}$ pair. Note that we only show the sentiment part of the unified labels (i.e., POS, NEG, and NEU) and use brackets to indicate the boundary. The marker ✗ denotes an incorrect prediction.

that the original word hidden representations (LSTM$^{\mathcal{B}}$ hidden states) are not suitable for transfer. Thus, we need to resort to the correlation vectors inferred by the DMI that models the transferable latent relations between aspect and opinion words.

- **No SAL** v.s. **SAL**: AD-SAL significantly and consistently exceeds Base Model+DMI by 6.46% and 4.92% Micro-F1 on average for the AD and ADS tasks, respectively. Without any adaptation, the captured relations by the DMI may not work well across domains, while the proposed SAL method can effectively align these latent relations to be domain-invariant.

- **No Selectivity** v.s. **Selectivity**: AD-SAL outperforms AD-AL by 5.18% and 4.11% Micro-F1 on average for the AD and ADS tasks, respectively. This proves the necessity to conduct the selective alignment. The SAL method can dynamically learn to control an alignment weight for each word to achieve a local semantic alignment, which captures a better domain-invariant feature space than pure adversarial learning that treats all words equally for the fine-grained adaptation.

- **Low-level** v.s. **High-level**: AD-SAL exceeds ADS-SAL by 2.31% and 1.94% Micro-F1 on average for the AD and ADS tasks, respectively. The label space of the *unified tags* for the high-level ADS task is more complicated than that of the *aspect boundary tag* for the low-level AD task. This gives the evidence that low-level neural features are more easily to transfer than high-level features.

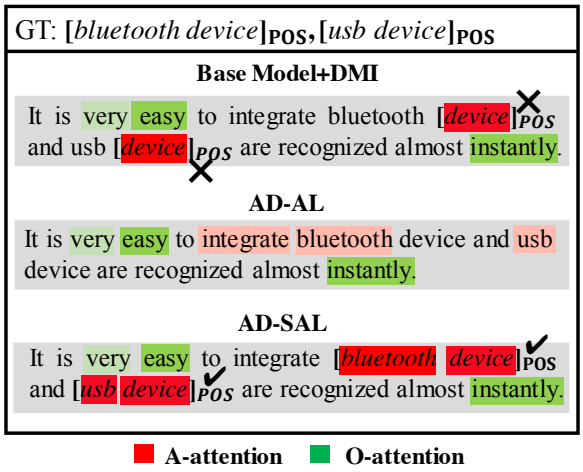

Figure 3: Visualization of attention for the $\mathbb{R}\rightarrow\mathbb{L}$ pair.

4.6 Case Analysis

As illustrated in Table 4, we perform case analysis of the $\mathbb{R}\rightarrow\mathbb{L}$ pair for the Base model+DMI, AD-AL, and the full model AD-SAL to demonstrate the necessity to conduct selective alignment for the fine-grained adaptation. The Base model+DMI can identify some domain-specific aspect words (e.g., *battery*, *ports*) without any supervision from the target domain. This is because the DMI can infer relational representations that capture some common latent relations between aspect and opinion words. However, it still cannot completely capture the multi-word aspect terms (e.g. *bluetooth device*, *battery life*) and sometimes it totally ignores them (e.g. *window 7*). The AD-AL performs pure adversarial learning for aligning all words in a sentence. Even though the domain adaptation method is adopted, it does not yield significant improvements and sometimes it becomes worse. For example, the AD-AL cannot even identify aspect words that can be captured

by the Base Model+DMI (e.g., *ports*, *devices*), or wrongly identifies some non-aspect words (e.g., *pc*). The reason is that pure adversarial learning treats all the words equally for the alignment, which may bring in noises of uninformative words into the shared feature space. To solve that, the full model AD-SAL performs a local semantic alignment to dynamically focus on aligning aspect words that contribute more to the domain-invariant feature space. Hence, AD-SAL model can precisely and completely identity all the aspect terms and make correct unified tag predictions.

Moreover, in Figure 3, we visualize the attentions from these models, where deeper colors denote larger weights. Compared with the Base model+DMI, the full model AD-SAL can precisely attend the complete aspect words from the target domain (A-attention), i.e., *bluetooth device* and *usb device*, and make correct predictions, while the AD-AL cannot achieve that. The AD-AL can only align all the words equally, which hinders the model to attend the aspect words, while the A-attention of the AD-SAL model can be used for both discriminative word tags predictions and acting as a learnable alignment weight for each word. This shows that the proposed SAL method can learn to align important aspect words to improve the transferability of the model for the fine-grained adaptation.

5 Related Works

E2E-ABSA can be broken into two sub-tasks, namely, aspect detection and aspect sentiment classification. The aspect detection aims to extract the aspect terms mentioned in the text and it has been actively studied (Qiu et al., 2011; Liu et al., 2015; Poria et al., 2016; Wang et al., 2016a; He et al., 2017; Wang et al., 2017; Majumder et al., 2018; Li et al., 2018b; Xu et al., 2018). The aspect sentiment classification is to predict the sentiment polarities of the given aspect terms and has also received a lot of attention recently (Dong et al., 2014; Tang et al., 2016; Wang et al., 2016b; Ma et al., 2017; Chen et al., 2017; Ma et al., 2018; He et al., 2018b; Li et al., 2018a, 2019b). For practical applications, a typical way is to pipeline these two sub-tasks together, which becomes ineffective due to the accumulated errors across tasks. These two sub-tasks have strong couplings and thus a unified formulation (Mitchell et al., 2013; Zhang et al., 2015; Li et al., 2019a) to handle them together in an end-to-end manner becomes a more promising direction. Despite its importance, existing studies are only exploring the performance in a single domain, while ignoring the transferability across domains. To address this problem, unsupervised domain adaptation methods can be applied. While existing methods focus on traditional cross-domain sentiment classification to learn shared representations for sentences or documents, including pivot-based methods (Blitzer et al., 2007; Pan et al., 2010; Bollegala et al., 2013; Yu and Jiang, 2016), auto-encoders (Glorot et al., 2011; Chen et al., 2012; Zhou et al., 2016), domain adversarial networks (Ganin et al., 2016; Li et al., 2017, 2018c), or semi-supervised methods (He et al., 2018a). Due to the difficulties in fine-grained adaptation, there exist very few methods for cross-domain aspect extraction (Jakob and Gurevych, 2010; Ding et al., 2017), which acts as a sub-task of E2E-ABSA, or aspect and opinion co-extraction (Li et al., 2012; Wang and Pan, 2018) that focuses on detecting aspect and opinion words, while E2E-ABSA needs to analyze more complicated correspondences between them. Besides, those methods can only rely on general syntactic relations between aspect and opinion words to bridge the domains. Different from them, our method leverages the attention mechanism (Bahdanau et al., 2014; Sukhbaatar et al., 2015; Shen et al., 2018, 2019) as a dynamic selector to automatically achieve the selective alignment.

6 Conclusion

The effectiveness of supervised methods for E2E-ABSA is limited due to the data scarcity. Our wok is the first attempt to resolve cross-domain E2E-ABSA by leveraging knowledge from other related domains to enhance the sequence learning of the target domain. Extensive experiments show the effectiveness of the proposed SAL method. Ablation studies also prove the necessity to perform selective alignment. In the future, the proposed SAL method can be potentially extended to other domain adaptation methods and applied to more general sequence labeling tasks including named entity recognition (Zhou et al., 2019c), part-of-speech tagging (Zhou et al., 2019a), etc.

Acknowledgement

We thank the support of Hong Kong CERG grants (16209715 & 16244616) and NSFC 61673202.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, pages 440–447.

Danushka Bollegala, David Weir, and John Carroll. 2013. Cross-domain sentiment classification using a sentiment sensitive thesaurus. *TKDE*, pages 1719–1731.

Minmin Chen, Zhixiang Xu, Fei Sha, and Kilian Q Weinberger. 2012. Marginalized denoising autoencoders for domain adaptation. In *ICML*, pages 1627–1634.

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

Ying Ding, Jianfei Yu, and Jing Jiang. 2017. Recurrent neural networks with auxiliary labels for cross-domain opinion target extraction. In *AAAI*, pages 3436–3442.

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 *ACL*, pages 49–54.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *JMLR*, pages 2096–2030.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *ICML*, pages 513–520.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2017. An unsupervised neural attention model for aspect extraction. In *ACL*, pages 388–397.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2018a. Adaptive semi-supervised learning for cross-domain sentiment classification. In *EMNLP*, pages 3467–3476.

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.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *KDD*, pages 168–177.

Niklas Jakob and Iryna Gurevych. 2010. Extracting opinion targets in a single-and cross-domain setting with conditional random fields. In *EMNLP*, pages 1035–1045.

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

Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. 2016. Ask me anything: Dynamic memory networks for natural language processing. In *ICML*, pages 1378–1387.

Fangtao Li, Sinno Jialin Pan, Ou Jin, Qiang Yang, and Xiaoyan Zhu. 2012. Cross-domain co-extraction of sentiment and topic lexicons. In *ACL*, pages 410–419.

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. 2019a. A unified model for opinion target extraction and target sentiment prediction. In *AAAI*, pages 6714–6721.

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

Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018c. Hierarchical attention transfer network for cross-domain sentiment classification. In *AAAI*.

Zheng Li, Ying Wei, Yu Zhang, Xiang Zhang, Xin Li, and Qiang Yang. 2019b. Exploiting coarse-to-fine task transfer for aspect-level sentiment classification. In *AAAI*, pages 4253–4260.

Zheng Li, Yu Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In *IJCAI*, pages 2237–2243.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*.

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

Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2017. Interactive attention networks for aspect-level sentiment classification. *arXiv preprint arXiv:1709.00893*.

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

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.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

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

Lili Mou, Zhao Meng, Rui Yan, Ge Li, Yan Xu, Lu Zhang, and Zhi Jin. 2016. How transferable are neural networks in NLP applications? In *EMNLP*, pages 479–489.

Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. 2010. Cross-domain sentiment classification via spectral feature alignment. In *WWW*, pages 751–760.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, AL-Smadi Mohammad, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *SemEval*, pages 19–30.

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

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.

Soujanya Poria, Erik Cambria, and Alexander F. Gelbukh. 2016. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowl.-Based Syst.*, 108:42–49.

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

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2018. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *AAAI*, pages 5446–5455.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Tensorized self-attention: Efficiently modeling pairwise and global dependencies together. In *NAACL*, pages 1256–1266.

Sainbayar Sukhbaatar, arthur szlam, Jason Weston, and Rob Fergus. 2015. End-to-end memory networks. In *NIPS*, pages 2440–2448.

Duyu Tang, Bing Qin, and Ting Liu. 2016. Aspect level sentiment classification with deep memory network. *arXiv preprint arXiv:1605.08900*.

Cigdem Toprak, Niklas Jakob, and Iryna Gurevych. 2010. Sentence and expression level annotation of opinions in user-generated discourse. In *ACL*, pages 575–584.

Wenya Wang and Sinno Jialin Pan. 2018. Recursive neural structural correspondence network for cross-domain aspect and opinion co-extraction. In *ACL*, pages 2171–2181.

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

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *AAAI*, pages 3316–3322.

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

Hu Xu, Bing Liu, Lei Shu, and Philip S. Yu. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction. In *ACL*, pages 592–598.

Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. How transferable are features in deep neural networks? In *NIPS*, pages 3320–3328.

Jianfei Yu and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *EMNLP*, pages 236–246.

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

Guangyou Zhou, Zhiwen Xie, Jimmy Xiangji Huang, and Tingting He. 2016. Bi-transferring deep neural networks for domain adaptation. In *ACL*, pages 322–332.

Joey Tianyi Zhou, Hao Zhang, Di Jin, and Xi Peng. 2019a. Dual adversarial transfer for sequence labeling. *TPAMI*, pages 1–1.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Xi Peng, Yang Xiao, and Zhiguo Cao. 2019b. Roseq: Robust sequence labeling. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–11.

Joey Tianyi Zhou, Hao Zhang, Di Jin, Hongyuan Zhu, Meng Fang, Rick Siow Mong Goh, and Kenneth Kwok. 2019c. Dual adversarial neural transfer for low-resource named entity recognition. In *ACL*, pages 3461–3471.