Unified Feature and Instance Based Domain Adaptation for Aspect-Based Sentiment Analysis

Chenggong Gong*, **Jianfei Yu***, **and Rui Xia**†
School of Computer Science and Engineering,
Nanjing University of Science and Technology, China
{cggong, jfyu, rxia}@njust.edu.cn

Abstract

The supervised models for aspect-based sentiment analysis (ABSA) rely heavily on labeled data. However, fine-grained labeled data are scarce for the ABSA task. To alleviate the dependence on labeled data, prior works mainly focused on feature-based adaptation, which used the domain-shared knowledge to construct auxiliary tasks or domain adversarial learning to bridge the gap between domains, while ignored the attribute of instance-based adaptation. To resolve this limitation, we propose an end-to-end framework to jointly perform feature and instance based adaptation for the ABSA task in this paper. Based on BERT, we learn domain-invariant feature representations by using part-of-speech features and syntactic dependency relations to construct auxiliary tasks, and jointly perform word-level instance weighting in the framework of sequence labeling. Experiment results on four benchmarks show that the proposed method can achieve significant improvements in comparison with the state-of-the-arts in both tasks of cross-domain End2End ABSA and cross-domain aspect extraction.

1 Introduction

Aspect extraction and aspect sentiment classification are two important sub-tasks in Aspect Based Sentiment Analysis (ABSA) (Liu, 2012; Pontiki et al., 2016), which aim to extract aspect terms and predict the sentiment polarities of the given aspect terms, respectively. Since these two sub-tasks have been well studied in the literature, a number of recent studies focus on the End2End ABSA task by employing a unified tagging scheme to tackle the two sub-tasks in an end-to-end manner (Mitchell et al., 2013; Zhang et al., 2015; Li et al., 2019a). The unified tagging scheme fuses aspect boundary tags {B, I, O} and sentiment polarities {POS, NEG, NEU} together, and formulates End2End ABSA as a sequence labeling problem. For example, given a sentence *"The price is reasonable, although the service is poor."*, the End2End ABSA task aims to jointly extract aspect terms and detect sentiment polarities over them. The extracted pairs in this example are {"price": Positive; "service": Negative}. However, these existing studies heavily rely on supervised learning over a large amount of labeled data, which is usually hard to obtain for ABSA due to the intensive nature of human annotation. Therefore, it will be very attractive to explore the End2End ABSA task in a cross-domain setting, which allows us to train a robust ABSA model for a resource-poor target domain based on enough annotated data in a resource-rich source domain.

Traditional domain adaptation methods primarily focus on coarse-grained sentiment classification (Blitzer et al., 2007; Pan et al., 2010; Glorot et al., 2011; Bollegala et al., 2012; Xia et al., 2013; Yu and Jiang, 2016; Ganin et al., 2016; Li et al., 2018b). Most of these methods can be grouped into two categories: feature-based domain adaptation and instance-based domain adaptation. Feature-based methods focus on finding a new feature representation which could reduce domain discrepancy. Instance-based methods aim to re-weight training samples in source domain which essentially attempts to assign higher weights to instances similar to the target domain, and lower weights to instances different from the target domain.

In contrast, due to the difficulty in fine-grained domain adaptation, only a few approaches have been proposed for cross-domain ABSA. Most of them explored cross-domain ABSA from the feature-based adaptation perspective, aiming to induce domain-invariant representations for each word. Specifically, Ding et al. (2017) and Wang and Pan (2018) proposed to use domain-shared

[*] Equal contribution.
[†] Corresponding author.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 7035–7045,
November 16–20, 2020. ©2020 Association for Computational Linguistics

syntactic knowledge to construct auxiliary tasks to reduce domain disparity. More recently, Li et al. (2019b) used the memory network to model the syntactic relations between words and designed a selective adversarial learning strategy to achieve word-level adaptation. However, all these methods are still based on traditional neural network architectures. As we all know, with the recent trend of pre-training in NLP (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2018), many pre-trained text encoders such as BERT have demonstrated their strong capability for domain-invariant representation learning, which poses new challenges for domain adaptation. Based on our preliminary experiments, we find that simply using BERT without domain adaptation has already obtained indistinguishable performance compared with previous domain adaptation methods. Therefore, it will be more attractive to extend these feature-based adaptation approaches to pre-trained models and further improve the domain adaptation performance.

Apart from the feature-based domain adaptation methods, Jiang and Zhai (2007) pointed out the importance of performing instance-based adaptation for different NLP tasks. As revealed by the theoretical analysis in Jiang and Zhai (2007), the domain discrepancy mainly comes from feature mismatches and instance mismatches, and needs to be jointly modeled from two attributes. However, previous studies only demonstrated the importance of instance-based domain adaptation in coarse-grained sentiment classification (Xia et al., 2014), and it is still unclear how to perform instance adaptation for the ABSA task.

To address the two challenges mentioned above, we first utilize BERT to learn domain-invariant features for the ABSA task, followed by proposing an instance weighting method for cross-domain ABSA. Finally, we integrate them into an end-to-end framework to jointly perform feature and instance adaptation. Specifically, for feature-based adaptation, we use the domain-shared part-of-speech information and dependency relations as self-supervised signals to enhance BERT to learn domain-invariant representation for cross-domain ABSA. For instance-based adaptation, since ABSA is typically modeled as a word-level prediction task, we propose to leverage a domain classier to dynamically learn an importance weight for each word and re-weight different words from the source domain during supervised training. Finally, we propose a unified framework to jointly perform feature and instance-based adaptation via sequential learning and joint learning, respectively. Experimental results on four benchmark datasets show that our method can significantly improve the performance of cross-domain End2End ABSA and cross-domain aspect extraction, and we further carry out ablation studies to quantitatively measure the effectiveness of each component in our unified framework.

The main contributions of this paper can be summarized as follows:

- To the best of our knowledge, we are the first to address both tasks of cross-domain End2End ABSA and cross-domain aspect extraction based on BERT.
- We propose a Unified Domain Adaptation (UDA) framework encompassing both feature-based adaptation and instance-based adaptation, which can significantly improve the performance of the fine-tuned BERT model without domain adaptation.
- Compared with the state-of-the-art domain adaptation method, our UDA approach gains an average improvement of 6.92% on Micro-F1 for cross-domain End2End ABSA.

2 Problem Statement

Following Li et al. (2019b), we model both the End2End ABSA task and the aspect extraction task as sequence labeling problems. The input is a sequence of tokens $w = \{w_1, w_2, ..., w_T\}$, and the output is a sequence of labels $y = \{y_1, y_2, ..., y_T\}$. For the End2End ABSA task, $y_i \in$ {B-POS, I-POS, B-NEG, I-NEG, B-NEU, I-NEU, O}; for the aspect extraction task, $y_i \in$ {B, I, O}. In this paper, we focus on unsupervised domain adaptation, where labeled data are not available in the target domain. Given a set of labeled tokens from a source domain $D_S = \{(w_s^i, y_s^i)\}_{i=1}^{N_S}$, and a set of unlabeled instances from a target domain $D_U = \{w_u^i\}_{i=1}^{N_U}$, our goal is to predict token labels for target test instances: $y_i^T = f_t(w_t^i)$, $D_T = \{w_t^i\}_{i=1}^{N_T}$.

The essential cause of domain adaptation is that the data distribution of the source domain and that of the target domain are different, i.e., $P_s(w, y) \neq P_t(w, y)$. The optimal model f_t^* for the target domain could be obtained by minimizing the following expected loss:

$$f_t^* = \arg\min_{f \in H} \int_{(w,y)} P_t(w, y) L(w, y, f) \qquad (1)$$

In unsupervised domain adaptation, since labeled data are not available in the target domain. We therefore minimize the empirical loss of data drawn from the source domain instead:

$$\begin{aligned} f_t^* &= \arg\min_{f\in H} \int_{(w,y)} P_t(w,y)L(w,y;f) \\ &= \arg\min_{f\in H} \int_{(w,y)} \frac{P_t(w,y)}{P_s(w,y)} P_s(w,y)L(w,y;f) \\ &\approx \arg\min_{f\in H} \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{P_t(w_i^s,y_i^s)}{P_s(w_i^s,y_i^s)} L(w_i^s,y_i^s;f) \\ &= \arg\min_{f\in H} \sum_{i=1}^{N_s} \frac{P_t(y_i^s|w_i^s)P_t(w_i^s)}{P_s(y_i^s|w_i^s)P_s(w_i^s)} L(w_i^s,y_i^s;f) \end{aligned} \tag{2}$$

According to the last line in Equation 2, as $P(w, y)$ can be factored into $P(y|w)P(w)$, an ideal domain adaptation model consider the following two attributes:

- feature-based adaptation, which needs to find a general feature representation w under which $\frac{P_t(y|w)}{P_s(y|w)} \to 1$;
- instance-based adaptation, which uses $r(w) = \frac{P_t(w)}{P_s(w)}$ as weights for sampling the instances in the source domain.

However, most previous domain adaptation methods in ABSA only presume feature-based adaptation which leverage auxiliary tasks or domain adversarial networks to learn domain-invariant feature representations while ignore instance-based adaptation. In this work, we take both attributes into consideration within a joint framework based on BERT for domain adaptation of the ABSA task.

3 Approach

Overview: As discussed above, the domain differences mainly come from two attributes, namely feature discrepancy and instance discrepancy. Therefore, we approach cross-domain End2End ABSA and cross-domain aspect extraction with a Unified Domain Adaptation (UDA) framework encompassing two components, named feature-based and instance-based domain adaptation components, which are showed in Figure 1. To reduce the feature discrepancy, we introduce two auxiliary tasks based on the domain-shared knowledge. To reduce the instance discrepancy, we perform word-level instance weighting to focus more on important words for the target domain. Finally, we unified the two components in a sequential and joint manner.

3.1 Feature-Based Domain Adaptation

Structural correspondence learning (Ando and Zhang, 2005; Blitzer et al., 2007) is the core idea of feature-based domain adaptation, whose goal is to use the structural correspondence to narrow the gap between domains. As a self-supervised learning mechanism based on large-scale corpus, the mask language model task of BERT is essentially a structural correspondence learning method. However, it does not use pivot words as masked objects, but randomly selects words to mask and predict. Based on our preliminary observations, in both tasks of End2End ABSA and aspect extraction, although aspect words vary a lot across domains, there are still some universal language structure correspondence between domains such as part-of-speech tags and dependency relations, which can serve as pivots to connect the domains. Nevertheless, this information has not been explicitly captured by BERT.

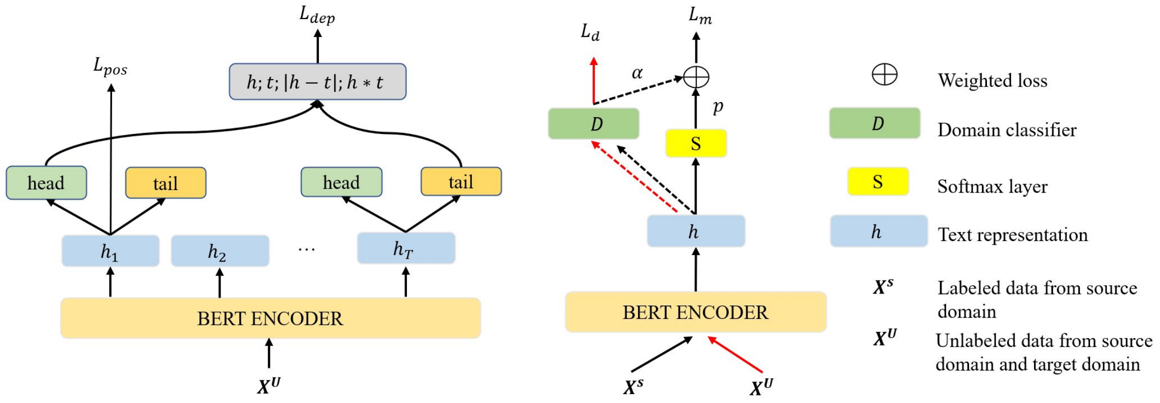
Motivated by this, we propose to use part-of-speech information and dependency relations as self-supervised signals to fine-tune BERT to learn the structural correspondence between domains for cross-domain ABSA. The overall architecture for our feature-based domain adaptation component is shown in Figure 1(a).

3.1.1 Masked POS Tag Prediction

We first convert the word sequences $w = \{w_1, w_2, ..., w_T\}$ into continuous embedding $e = \{e_1, e_2, ..., e_T\}$. The embedding of each word is the sum of four type embeddings $e_i = [t_i, s_i, p_i, tag_i]$. $t_i \in R^d$ is the word embedding of w_i. $s_i \in R^d$ is the segment embedding, which is used as a segmentation mark between sentences. $p_i \in R^d$ is the embedding for the absolute position of a word. $tag_i \in R^d$ is the POS tag embedding.

The first three kinds of embedding are the same as those defined in Devlin et al. (2018), and are initialized using the pre-trained BERT embedding. The POS tag embedding matrix is randomly initialized and trained with unlabeled data from the source and target domains. Since BERT uses sub-word tokenizer, we assume that sub-words share the same POS tags. The word embedding sequences $e = \{e_1, e_2, ..., e_T\}$ were converted into a context-aware representation $H = \{h_1, h_2, ..., h_T\}$ through a multi-layer transformer as follows:

$$H = \text{transformer}(E)$$



(a) Feature-Based Domain Adaptation Component

(b) Instance-Based Domain Adaptation Component

Figure 1: Two components in our Unified Domain Adaptation (UDA) approach. Figure 1(a) shows two auxiliary tasks we proposed to learn a domain invariant representation for cross-domain ABSA. Figure 1(b) shows an illustration of our word-level instance weight method. The black line represents the flow of training data from the source domain to optimize L_m (Equation 10). The red line represents the unlabeled data from the source and target domain, which is used to optimize L_d (Equation 8). The dotted line indicates that there is no back propagation during training.

To prepare the input for masked POS tag prediction task, we randomly select about 25% of tokens and replaced the original tokens and POS tags with [MASK]. After being encoded by transformer, the masked feature in H is fed into the softmax layer, and converted to the probability over POS tag types p_i^{pos} as follows:

$$p_i^{pos} = \mathrm{softmax}(W_p h_i + b_p)$$

where $p_i^{pos} \in R^{n\_tags}$, n_tags is the number of POS tag type, W_p and b_p are the weight matrix and bias vector of the softmax layer. We only use the masked features for prediction and we use cross-entropy loss for optimization:

$$L_{pos} = \sum_{D_U} \sum_{i}^{T} I(i) l(p_i^{pos}, y_i^{pos}) \tag{3}$$

where $I(i)$ is an indicator function, which is equal to 1 if masked, otherwise 0, and y_i^{pos} is the real POS tag type of the i-th token.

3.1.2 Dependency Relation Prediction

To reconstruct the syntactic relation in H that is useful for ABSA, we feed the context-aware representation H to two non-linear transformation functions to obtain $H^{head} = \{h_1^{head}, h_2^{head}, ..., h_T^{head}\}$ and $H_{tail} = \{h_1^{tail}, h_2^{tail}, ..., h_T^{tail}\}$ as:

$$h_i^{head} = \tanh(W_1 h_i + b_1), \tag{4}$$

$$h_i^{tail} = \tanh(W_2 h_i + b_2), \tag{5}$$

where $h_i^{head} \in R^{\frac{d}{4}}$ and $h_i^{tail} \in R^{\frac{d}{4}}$, and W_1 and W_2 are learnable parameters. h_i^{head} and h_i^{tail} can be viewed as the representations of the head token and child token in the dependency tree, respectively. Suppose the i-th and j-th words in the input sequence are connected in the dependency tree and represent the head node and the child node, respectively. We use o_{ij} to predict their dependency relation:

$$o_{ij} = [h_i^{head}; h_j^{tail}; h_i^{head} - h_j^{tail}; h_i^{head} \odot h_j^{tail}]$$

where $[;]$ indicates concatenation operation, $-$ and $\odot$ indicate element-wise subtraction and multiplication, respectively. The o_{ij} was converted into p_{ij}^{dep} by a softmax layer.

$$p_{ij}^{dep} = \mathrm{softmax}(W_d o_{ij} + b_d)$$

where $W_d \in R^{d \times n_{arc}}$ is the weight matrix for relation classification, and n_{arc} is the number of relation classes. We use token pairs that are directly connected in the dependency tree to construct training examples. $I(ij)$ indicates whether token pairs (i, j) have a direct edge in dependency tree or not. If they are connected in the dependency tree, we predict their dependency relation. The optimization objective is as follows:

$$L_{dep} = \sum_{D_U} \sum_{i}^{T} \sum_{j}^{T} I(ij) l(p_{ij}^{dep}, y_{ij}^{dep}) \tag{6}$$

We perform feature-based domain adaptation through two auxiliary tasks, and optimize the following objective function for feature-based domain adaptation:

$$L_{feature} = L_{pos} + \lambda L_{dep} \tag{7}$$

where λ is a trade-off hyper-parameter to control the contributions of two tasks, and L_{pos} and L_{dep} are defined in Equation 3 and Equation 6 respectively.

3.2 Instance-Based Domain Adaptation

As analyzed above, instances-based domain adaptation aims to use $\frac{p_t}{p_s}$ to re-weight instances in the source domain to reduce the gap across domains. However, unlike the coarse-grained domain adaptation, our fine-grained ABSA tasks are modeled as sequence labeling tasks, which are essentially word-level classification problems. Since each sentence has domain-invariant words and domain-specific words, we need to obtain the domain distribution of each word and re-weight it at the word level.

Specifically, while training the main task, we also train a word-level domain classifier based on unlabeled data, whose goal is to identify whether each word is from the source domain or the target domain. The output of transformer H was then send to a softmax layer to get the domain distribution probability of the i-th word w_i as follows:

$$p_i^D = \text{softmax}(W_d h_i + b_d)$$

where $p_i^D \in R^{|y^{n\text{-}d}|}$ is the domain distribution probability and $y^{n\text{-}d} = \{source, target\}$. The domain classifier D is trained by the cross entropy loss between p_i^D and the ground-truth y_i^D as follows:

$$L_d = \sum_{D_U} \sum_{i=1}^{T} l(p_i^D, y_i^D) \tag{8}$$

Through the domain classifier D, we can get the domain distribution of each word, and we use the ratio of its target-domain probability to its source-domain probability, i.e., $\frac{p_{i,t}^D}{p_{i,s}^D}$, as the weight of each word during training the main task. Since the training of domain classifiers will make it difficult to generalize across domains, we cut off the gradient back pass, so that L_d only optimizes the parameters W_d and b_d in the softmax layer. As shown in Figure 1(b), when training D, the red dashed line represents the feed-forward calculation, but there is no gradient return. The main task is optimized with the weighted cross entropy loss as follows:

$$p_i^m = \text{softmax}(W_m h_i + b_m) \tag{9}$$

$$L_m = \sum_{D_S} \sum_{i}^{T} \alpha_i * l(p_i^m, y_i^m) \tag{10}$$

where α_i (i.e., the weight of each word) is computed based on the re-normalization over $\frac{p_{i,t}^D}{p_{i,s}^D}$ of all the T tokens, and the probability $p_{i,t}^D$ and $p_{i,s}^D$ are obtained by the domain classifier D.

Although AD-SAL (Li et al., 2019b) also learns an importance weight for each word, our method is significantly different. First of all, AD-SAL still essentially belongs to feature-based domain adaptation, and our method belongs to instance-based domain adaptation. For AD-SAL, the goal is to learn domain-invariant representations for each word through domain adversarial learning. As aspect words are the core of ABSA (this is also consistent with our motivation), AD-SAL introduces aspect attention weights in domain adversarial learning to learn domain-invariant representations for aspect words. In contrast, our method uses domain classifier to automatically learn the importance of each word for the target domain, so that it pays more attention to words (including aspect words and opinion words) that are closer to the target domain during the main task training process. Secondly, the training process of SAL is independent of the main task. In contrast, in our method, the weight of each word is learned through the domain classifier, and the learning process is combined with the main task, which will make the model automatically learn which words are more important for the target domain and the main task.

3.3 Training Mechanism

As analyzed before, our work mainly contains two components: feature-based and instance-based component, which was corresponding to the two attributes of domain adaptation respectively. To dynamically learn a weight for the instance-based component, L_d (Equation 8) and L_m (Equation 10) update jointly. The training objective of instance-based domain adaptation is as follows:

$$L = L_m + L_d \tag{11}$$

The feature-based domain adaptation aims to learn a shared feature space for the target domain, which could be trained separately from the main task. Thus, we can merge the instance-based component and the feature-based component in a sequential or joint training way.

Sequential Training: In the sequential training, we first train the auxiliary tasks to learn a shared feature space, and the training objective is given in Equation 7. Based on the learned shared feature space, we then perform instance-based domain adaptation, and the training objective is given in Equation 11.

Joint Training: We can also merge the two components in a joint way, i.e., training auxiliary tasks and the main task in a multi-task manner. The training objective is:

$$L = L_m + L_d + L_{pos} + \lambda L_{dep} \qquad (12)$$

As revealed by Ando and Zhang (2005) and Blitzer et al. (2007), the success of the target task comes from multiple related tasks to help discover common structures between domains. As they are trained jointly, the information from auxiliary tasks could be propagated to the main task.

4 Experiment

4.1 Data & Experiment Setup

Datasets: We conduct experiments on four benchmark datasets: Laptop (L), Restaurant(R), Device (D), and Service (S). L contains reviews from the laptop domain in SemEval-2014 ABSA challenge (Pontiki et al., 2014). Following the setup in Li et al. (2019a), R is the union set of the restaurant datasets from SemEval ABSA challenge 2014, 2015, and 2016 (Pontiki et al., 2014, 2015, 2016). D is a combination of device reviews from Toprak et al. (2010) and S is introduced by Hu and Liu (2004) containing reviews from web services. Detailed statistics are shown in Table 1.

Dataset	Domain	Sentences	Training	Testing
L	Laptop	3845	3045	800
R	Restaurant	6035	3877	2158
D	Device	3836	2557	1279
S	Service	2239	1492	747

Table 1: Statistics of the datasets.

Settings & Implementation Details: We conduct experiments on 10 source→target pairs using the four domains above. Following the setup in (Li et al., 2019b), we removed D→L and L→D, as D and L are similar. For each source→target pair, the training data consists of the training data in the source domain and the unlabeled training data in the target domain. The evaluation results are obtained based on the test data from the target domain. We use Spacy to extract part-of-speech tags and dependency relations, and finally used 54 types of part-of-speech tags and 47 types of dependency relation.

For our proposed UDA approach, since it is a general DA framework, we can potentially use any pre-trained BERT model or their variants as our base model. In this work, we adopt two kinds of base models: BERT_B and BERT_E. For BERT_B, it refers to the uncased BERT_base model pre-trained by Devlin et al. (2018)[1]. For BERT_E, it refers to an extended version of BERT_B, which further incorporates the domain knowledge (Xu et al., 2019) by fine-tuning the pre-trained BERT_B model with the BERT language model on product reviews from a combination of Yelp Challenge Datasets[2] and the Electronics dataset from Amazon[3] (He and McAuley, 2016). Note that for the BERT language model fine-tuning, we use 32 bit floating point computations using the Adam optimizer (Kingma and Ba, 2014). The batch size is set to 32, and the learning rate is set to $3 \cdot 10^{-5}$. For training downstream tasks, we set λ to 0.1, and use the Adam optimizer. We perform grid search over a learning rate of $2 \cdot 10^{-5}$, $3 \cdot 10^{-5}$, $5 \cdot 10^{-5}$ and a batch size of 16, 32, 64. We tune all these parameters on the validation set, which is composed by 10% samples from the training set [4].

Evaluation Metric: The evaluation metric we used is Micro-F1. Following the setting in existing work, only exact match could be counted as correct. All experiments are repeated 5 times and we report the average results over 5 runs.

4.2 Baselines & Main Results

We compare our Unified Domain Adaptation (UDA) approach with several highly competitive DA methods as follows:

[1] We make use of the uncased BERT_base model as part of the pytorch-transformers library: `https://github.com/huggingface/pytorch-transformers`

[2] `https://www.yelp.com/dataset/challenge`

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

[4] The source code and corpus can be obtained at `https://github.com/NUSTM/BERT-UDA`

	Source→Target Pairs										
Methods	S→R	L→R	D→R	R→S	L→S	D→S	R→L	S→L	R→D	S→D	AVG
Hier-Joint†	31.10	33.54	32.87	15.56	13.90	19.04	20.72	22.65	24.53	23.24	23.72
RNSCN†	33.21	35.65	34.60	20.04	16.59	20.03	26.63	18.87	33.26	22.00	26.09
AD-SAL†	41.03	43.04	41.01	28.01	27.20	26.62	34.13	27.04	35.44	33.56	33.71
BERT$_\text{B}$	44.66	40.38	40.32	19.48	25.78	30.31	31.44	30.47	27.55	33.96	32.44
BERT$_\text{B}$-DANN	45.84	41.73	34.68	21.60	25.10	18.62	30.41	31.92	34.41	23.97	30.79
BERT$_\text{B}$-UDA	47.09	45.46	42.68	**33.12**	27.89	28.03	33.68	34.77	34.93	32.10	35.98
BERT$_\text{E}$	51.34	45.40	42.62	24.44	23.28	28.18	39.72	35.04	33.22	33.22	35.65
BERT$_\text{E}$-DANN	50.31	47.39	42.20	28.35	26.69	28.77	38.83	34.29	33.42	37.14	36.74
BERT$_\text{E}$-UDA	**53.97**	**49.52**	**51.84**	30.67	**27.78**	**34.41**	**43.95**	**35.76**	**40.35**	**38.05**	**40.63**

Table 2: Comparison results for cross-domain End2End ABSA based on Micro-F1. The results marked by † are extracted from Li et al. (2019b). It is worth noting that different from Li et al. (2019b), we did not remove training/test samples where all the tokens are labeled as 'O' in our experiments, because a moderate amount of product reviews only contain implicit aspects in real scenarios. If we remove these samples, we can get an extra improvement of around 5% on Micro-F1 for all the BERT-based methods in our preliminary experiments.

	Source→Target Pairs										
Methods	S→R	L→R	D→R	R→S	L→S	D→S	R→L	S→L	R→D	S→D	AVG
Hier-Joint†	46.39	48.61	42.96	27.18	25.22	29.28	34.11	33.02	34.81	35.00	35.66
RNSCN†	48.89	52.19	50.39	30.41	31.21	35.50	47.23	34.03	46.16	32.41	40.84
AD-SAL†	52.05	**56.12**	51.55	**39.02**	**38.26**	36.11	45.01	35.99	**43.76**	**41.21**	43.91
BERT$_\text{B}$	54.29	46.74	44.63	22.31	30.66	33.33	37.02	36.88	32.03	38.06	37.60
BERT$_\text{B}$-DANN	54.32	48.34	44.63	25.45	29.83	26.53	36.79	39.89	33.88	38.06	37.77
BERT$_\text{B}$-UDA	56.08	51.91	50.54	34.62	32.49	34.52	46.87	43.98	40.34	38.36	42.97
BERT$_\text{E}$	57.56	50.42	45.71	26.50	25.96	30.40	44.18	41.78	35.98	35.13	39.36
BERT$_\text{E}$-DANN	58.55	52.40	45.21	31.29	30.16	30.86	46.90	40.43	36.32	39.17	41.13
BERT$_\text{E}$-UDA	**59.07**	55.24	**56.40**	34.21	30.68	**38.25**	**54.00**	**44.25**	42.40	40.83	**45.53**

Table 3: Comparison results for cross-domain Aspect Extraction (AE) based on Micro-F1.

- Hier-Joint (Ding et al., 2017): A recurrent neural network (RNN) with manually designed rule-based auxiliary tasks.
- RNSCN (Wang and Pan, 2018): A recursive neural structural correspondence network that incorporates syntactic structures.
- AD-SAL (Li et al., 2019b): A recent deep model that achieves state-of-the-art performance on End2End ABSA across domains.
- BERT$_\text{B}$ (Devlin et al., 2018) and BERT$_\text{E}$ (Xu et al., 2019): directly fine-tuning the two kinds of pre-trained models on the down-stream task.
- BERT$_\text{B}$-DANN and BERT$_\text{E}$-DANN: We respectively use BERT$_\text{B}$ and BERT$_\text{E}$ as the base models, and simultaneously perform adversarial training on each word, which can be viewed as the BERT version of the widely used DANN approach proposed by Ganin et al. (2016).

The overall comparison results on cross-domain End2End ABSA are shown in Table 2. On the one hand, we can observe that BERT$_\text{B}$-UDA generally performs better than the state-of-the-art DA approach (i.e., AD-SAL) on most transfer pairs for cross-domain End2End ABSA. Moreover, with BERT$_\text{E}$ as the base model, our BERT$_\text{E}$-UDA approach can significantly boost the average performance of BERT$_\text{B}$-UDA from 35.75% to 40.63%, which outperforms AD-SAL by 6.92% on average. On the other hand, by comparing BERT-based approaches, we can clearly see that simply performing adversarial training (i.e., DANN) for each word does not give satisfactory improvements over BERT$_\text{B}$ and BERT$_\text{E}$, whereas our UDA approach can significantly outperform all the BERT-based baselines and consistently achieve the best performance on all the transfer pairs. All these observations demonstrate the effectiveness of our UDA framework.

We also report the results on cross-domain AE in Table 3. Clearly, we can find that the overall trend of the performance of each approach is similar to their performance in cross-domain End2End ABSA. But the results of End2End ABSA are much lower than those of AE, which is reasonable as AE is one of its sub-tasks. Compared with AD-SAL, our BERT$_\text{E}$-UDA approach is 1.62% higher in terms of the average performance of all transfer pairs for the task of cross-domain AE. Compared with cross-domain End2End ABSA, the improve-

ABSA	Only Feature	Only Instance	Sequential	Joint
S→R	53.09	51.55	53.56	**53.97**
L→R	**49.79**	48.08	49.47	49.52
D→R	50.67	46.22	**52.13**	51.84
R→S	27.09	25.01	28.02	**30.67**
L→S	24.51	25.92	26.73	**27.78**
D→S	**35.89**	34.21	34.89	34.41
R→L	41.93	40.52	42.46	**43.95**
S→L	35.17	34.33	34.52	**35.76**
R→D	37.79	39.27	**40.42**	40.35
S→D	**38.45**	36.70	37.85	38.05
AVG	39.44	38.18	40.01	**40.63**

Table 4: Ablation study of our UDA approach based on BERT_E for cross-domain End2End ABSA.

S→R	contentious, bearing, hated, beauty, ##mi, amazement, ##ant, canned, mistake, madden, accused, nicely, employee, proud, difficulty, impressive, likely, catalogue, ##working
L→R	enjoying, lesson, strongly, reality, comfortably, artwork, food, loving, dissatisfaction, spice, ##kind, fork, appears, weary, desk, projects, monster, covering, recipients, purchases
D→R	displayed, desk, robust, lightly, capable, waking, satisfactory, birthday, releasing, kitchen, noises, appearing, experiences, sophisticated, extreme, providing, nuts, interaction, recommendations

Table 5: Words with higher instance weights in the instance-based adaptation component of our UDA approach.

ment of our approach is not that huge, probably due to the inherent difficulty of cross-domain AE, where most aspect words in different domains do not intersect.

4.3 Ablation Study

Since our UDA framework includes two components, i.e., feature-based and instance-based domain adaptation, we further conduct experiments over different variants of the proposed model in Table 4 to show the effect of each component. Only Feature and Only Instance represent the feature-based domain adaptation and the instance-based domain adaptation on basis of BERT_E, respectively. Compared with BERT_E, both components have achieved much better F1 scores on most transfer pairs. This indicates that our proposed two components have effectively reduced the domain discrepancy. Besides, we also merge the two components in a sequential and joint way, denoted by Sequential and Joint respectively. It is easy to see that Joint performs slightly better than Sequential, which shows the advantages of joint optimization.

To qualitatively show the effect of our word-level instance weighting method, we show the most important words for the target domain on three transfer pairs in Table 5. The results show that the common opinion words (e.g., *beauty*, *amazement* and *satisfactory*) or aspect words (e.g., *employee*, *desk* and *kitchen*) gain more weight in the word-level instance weighting.

5 Related Work

Aspect extraction and aspect-level sentiment classification are two important subtasks in Aspect-Based Sentiment Analysis (ABSA), which aim to extract aspect terms and identify the sentiment orientations towards them, respectively (Liu, 2012). As two fundamental tasks, aspect extraction (Qiu et al., 2011; Liu et al., 2015; Poria et al., 2016; Wang et al., 2016a, 2017; Li et al., 2018a; Xu et al., 2018) and aspect-level sentiment classification (Dong et al., 2014; Tang et al., 2016; Wang et al., 2016b; Ma et al., 2017; Wang et al., 2018; Li et al., 2019c) have been extensively studied in the literature.

Since these two tasks are strongly related with each other, a number of previous studies propose to tackle them together in an end-to-end manner (Mitchell et al., 2013; Zhang et al., 2015). Some recent studies have further demonstrated that a unified tagging scheme can effectively eliminate the error propagation issue of traditional pipeline methods, and thus achieve the state-of-the-art performance. However, since annotating each word with fine-grained label is time-consuming, it is next to impossible to obtain enough annotated data for the ABSA task in every new domain. Therefore, in this work, we resort to transfer learning, and focus on proposing an effective domain adaptation approach for the ABSA task.

Existing domain adaptation studies in sentiment analysis primarily focus on coarse-grained domain adaptation problem. Most of them can be grouped into two categories: feature-based methods (Blitzer et al., 2007; Pan et al., 2010; Chen et al., 2012; Zhuang et al., 2015; Yu and Jiang, 2017; Ganin et al., 2016; Li et al., 2018b) and instance-based methods (Jiang and Zhai, 2007; Bickel et al., 2007; Xia et al., 2013, 2014). The former one attempts to learn a domain-invariant representation with auxiliary tasks or domain adversarial learning, while the latter one tries to re-weight source instances in order to assign higher weights to instances similar to the target domain and lower weights to instances different from the target domain.

Due to the challenges in fine-grained domain adaptation, only a few studies have explored the ABSA task in cross-domain settings. Ding et al. (2017) and Wang and Pan (2018) used domain general syntactic relations to construct auxiliary task to bridge the domains. Li et al. (2019b) proposed a selective adversarial learning method to learn domain-invariant representations for aspect words. However, these methods are still based on traditional networks such as LSTM, but fail to resort to recent pre-trained text encoders such as BERT. Moreover, all these methods only perform feature-based adaptation, but ignore instance-based adaptation. In contrast, our work aims to propose a unified feature and instance-based method based on BERT for cross-domain ABSA. Besides, it is worth noting that Rietzler et al. (2019) explored BERT for cross-domain aspect sentiment classification, where the aspect terms or categories are provided for both source and target domains. Different from their work, we primarily focus on the cross-domain End2End ABSA task in this work, which aims to first extract aspect terms followed by identifying the sentiment towards each detected aspect term.

6 Conclusion

In this paper, we explored the potential of BERT to domain adaptation, and proposed a unified feature and instance-based adaptation approach for both tasks of cross-domain End2End ABSA and cross-domain aspect extraction. In feature-based domain adaptation, we use domain-shared syntactic relations and POS tags to construct auxiliary tasks, which can help learn domain-invariant representations for domain adaptation. In instance-based domain adaptation, we employ a domain classifier to learn to assign appropriate weights for each word. Extensive experiments on four benchmark datasets demonstrate the superiority of our Unified Domain Adaptation (UDA) approach over existing methods in both cross-domain End2End ABSA and cross-domain aspect extraction.

Acknowledgments

We would like to thank three anonymous reviewers for their insightful comments and helpful suggestions. This work was supported by the Natural Science Foundation of China (No. 61672288, 62076133, 62006117, and 72001102), and the Natural Science Foundation of Jiangsu Province for Young Scholars (SBK2020040749) and Distinguished Young Scholars (SBK2020010154). Chenggong Gong and Jianfei Yu contributed equally to this paper. Rui Xia is the corresponding author.

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6(Nov):1817–1853.

Steffen Bickel, Michael Brückner, and Tobias Scheffer. 2007. Discriminative learning for differing training and test distributions. In *Proceedings of the 24th international conference on Machine learning*.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*.

Danushka Bollegala, David Weir, and John Carroll. 2012. Cross-domain sentiment classification using a sentiment sensitive thesaurus. *IEEE transactions on knowledge and data engineering*, 25(8):1719–1731.

Minmin Chen, Zhixiang Xu, Kilian Q Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Ying Ding, Jianfei Yu, and Jing Jiang. 2017. Recurrent neural networks with auxiliary labels for cross-domain opinion target extraction. In *Thirty-First AAAI Conference on Artificial Intelligence*.

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 the 52nd annual meeting of the association for computational linguistics (volume 2: Short papers)*.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Domain adaptation for large-scale sentiment classification: a deep learning approach. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*.

Ruining He and Julian McAuley. 2016. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In *proceedings of the 25th international conference on world wide web*.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*.

Jing Jiang and ChengXiang Zhai. 2007. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th annual meeting of the association of computational linguistics*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Xin Li, Lidong Bing, Piji Li, and Wai Lam. 2019a. A unified model for opinion target extraction and target sentiment prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33.

Xin Li, Lidong Bing, Piji Li, Wai Lam, and Zhimou Yang. 2018a. Aspect term extraction with history attention and selective transformation. *arXiv preprint arXiv:1805.00760*.

Zheng Li, Xin Li, Ying Wei, Lidong Bing, Yu Zhang, and Qiang Yang. 2019b. Transferable end-to-end aspect-based sentiment analysis with selective adversarial learning. *arXiv preprint arXiv:1910.14192*.

Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018b. Hierarchical attention transfer network for cross-domain sentiment classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Zheng Li, Ying Wei, Yu Zhang, Xiang Zhang, and Xin Li. 2019c. Exploiting coarse-to-fine task transfer for aspect-level sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33.

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

Pengfei Liu, Shafiq Joty, and Helen Meng. 2015. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

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

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*.

Sinno Jialin Pan, Xiaochuan Ni, Jian-Tao Sun, Qiang Yang, and Zheng Chen. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad Al-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, et al. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *10th International Workshop on Semantic Evaluation (SemEval 2016)*.

Maria Pontiki, Dimitrios Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Auresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*.

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

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.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/ language understanding paper.pdf*.

Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. 2019. Adapt or get left behind: Domain adaptation through bert language model finetuning for aspect-target sentiment classification. *arXiv preprint arXiv:1908.11860*.

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 *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Shuai Wang, Sahisnu Mazumder, Bing Liu, Mianwei Zhou, and Yi Chang. 2018. Target-sensitive memory networks for aspect sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Wenya Wang and Sinno Jialin Pan. 2018. Recursive neural structural correspondence network for cross-domain aspect and opinion co-extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016a. Recursive neural conditional random fields for aspect-based sentiment analysis. *arXiv preprint arXiv:1603.06679*.

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

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016b. Attention-based lstm for aspect-level sentiment classification. In *EMNLP 2016: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Rui Xia, Xuelei Hu, Jianfeng Lu, Jian Yang, and Chengqing Zong. 2013. Instance selection and instance weighting for cross-domain sentiment classification via pu learning. In *Twenty-Third International Joint Conference on Artificial Intelligence*.

Rui Xia, Jianfei Yu, Feng Xu, and Shumei Wang. 2014. Instance-based domain adaptation in nlp via in-target-domain logistic approximation. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2018. Double embeddings and cnn-based sequence labeling for aspect extraction. *arXiv preprint arXiv:1805.04601*.

Hu Xu, Bing Liu, Lei Shu, and Philip S Yu. 2019. Bert post-training for review reading comprehension and aspect-based sentiment analysis. *arXiv preprint arXiv:1904.02232*.

Jianfei Yu and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *EMNLP 2016: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Jianfei Yu and Jing Jiang. 2017. Leveraging auxiliary tasks for document-level cross-domain sentiment classification. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing*.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Fuzhen Zhuang, Xiaohu Cheng, Ping Luo, Sinno Jialin Pan, and Qing He. 2015. Supervised representation learning: Transfer learning with deep autoencoders. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.