Simultaneous Learning of Pivots and Representations for Cross-Domain Sentiment Classification

Liang Li,[1] **Weirui Ye,**[1] **Mingsheng Long,**[1] **(✉) Yateng Tang,**[2] **Jin Xu,**[2] **Jianmin Wang**[1]
[1]School of Software, BNRist, Tsinghua University, China
[1]Research Center for Big Data, Tsinghua University, China
[2]Data Quality Team, Wechat, Tencent Inc., China
{liliang17, ywr16}@mails.tsinghua.edu.cn, {mingsheng, jimwang}@tsinghua.edu.cn, {fredyttang, jinxxu}@tencent.com

Abstract

Cross-domain sentiment classification aims to leverage useful knowledge from a source domain to mitigate the supervision sparsity in a target domain. A series of approaches depend on the pivot features that behave similarly for polarity prediction in both domains. However, the engineering of such pivot features remains cumbersome and prevents us from learning the disentangled and transferable representations from rich semantic and syntactic information. Towards learning the pivots and representations simultaneously, we propose a new Transferable Pivot Transformer (TPT). Our model consists of two networks: a Pivot Selector that learns to detect transferable n-gram pivots from contexts, and a Transferable Transformer that learns to generate domain-invariant representations by modeling the correlation between pivot and non-pivot words. The Pivot Selector and Transferable Transformer are jointly optimized through end-to-end back-propagation. We experiment with real tasks of cross-domain sentiment classification over 20 domain pairs where our model outperforms prior arts.

Introduction

Despite the great progress achieved by previous works in the area of Natural Language Processing (NLP), cross-domain sentiment classification is still a challenging task. The key challenge lies in that data from different domains are drawn from different distributions, and there are a lot of domain-specific words and expressions in practice. There is a dilemma that domain-common words are usually not discriminative enough for distinguishing sentiment polarity, while many sentiment-words are domain-specific and cannot transfer well across domains. Several techniques have been proposed for sentiment domain adaptation. The aim is to bridge the source and target domains by learning domain-invariant feature representations so that a classifier trained on a source domain can be adapted to another target domain. In cross-domain sentiment classification, previous works (Blitzer, Dredze, and Pereira 2007; Pan et al. 2010; Wu and Huang 2016; Yu and Jiang 2016) mainly rely on a basic intuition that non-pivot features could be aligned with the help of pivot features. For sentiment classification, Blitzer *et al.* (2007) set up multiple pivot-prediction tasks to induce a projected joint low-dimensional space that bridges the domains, and these auxiliary tasks are highly correlated with the sentiment classification task.

With the advances of deep learning in NLP, some early studies explored deep models for sentiment domain adaptation (Glorot, Bordes, and Bengio 2011; Chen et al. 2012) by learning cross-domain representations to disentangle the variational factors behind data. Recently, some works integrated pivot selection with deep models. Ziser and Reichart (2016) presented the AE-SCL-SR model to marry pivot based and autoencoder based approaches. They also incorporated the pivot to language modeling, and their PBLM model (Ziser and Reichart 2018; 2019) yielded superiority over many previous approaches. Despite their promising results, they cannot learn pivots during training and completely depend on feature selection methods to *engineer* the pivots, which may be discriminative for source domain task but do not transfer well across domains. Further, these works share another limitation: Akin to left-to-right language modeling, the prediction task conditions on previous words including pivots and non-pivots, which may not sufficiently model the connections between pivot and non-pivot features.

How to detect pivots effectively still remains an important challenge for sentiment domain adaptation. In this paper, we propose the Transferable Pivot Transformer (TPT) for cross-domain sentiment classification. TPT consists of two networks: one Pivot Selector for learning to find transferable n-gram pivots, based on its mutual information and the uncertainty of distinguishing them between domains, and another Transferable Transformer for explicitly modeling the relationship between pivots and non-pivots. Unlike pivot engineering, our pivot learning can be seamlessly combined with representation learning, which enables end-to-end learning of expressive pivots and representations from scratch. The experiments conducted on a number of different source and target domains show that our method achieves the best accuracy compared with a number of strong baselines.

Related Work

Unsupervised Domain Adaptation: In the scenario of unsupervised domain adaptation, we have access to labeled source samples and only unlabeled target samples. Unsupervised representation learning with deep neural networks (DNN) has been explored for feature adaptation (Glorot, Bordes, and Bengio 2011; Chen et al. 2012). **SDA** (Glorot, Bordes, and Bengio 2011) is the first work that automatically learns feature representations of documents from large amounts of data. Chen *et al.* (2012) proposed a Marginalized Stacked Denoising Autoencoder (**mSDA**) to address the speed and scalability problem of SDA for high-dimensional data. Many recent works in vision problems extract domain-invariant representations in deep models via explicit minimization objectives (Tzeng et al. 2014; Long et al. 2015) or adversarially learning the feature representations to confuse a domain discriminator (Ganin and Lempitsky 2015; Tzeng et al. 2015; Ganin et al. 2016; Long et al. 2018).

Pivot-based Domain Adaptation: The majority of feature adaptation methods for sentiment analysis rely on a key intuition that even though certain opinion words are completely distinct for each domain, they can be aligned if they have a high correlation with some domain-invariant opinion words. Blitzer *et al.* (2007) proposed a method based on structural correspondence learning (**SCL**), which uses pivot feature prediction to induce a projected feature space that works well for both the source and the target domains. Pan *et al.* (2010) proposed the Spectral Feature Alignment (**SFA**) to find an alignment between pivots and non-pivots. These methods are remarkable advances in sentiment domain adaptation, but the pivots are selected by statistical criteria, thus the pivots cannot be learned by the model. Recently, deep neural models are explored to automatically produce superior real-valued feature representations. Yu and Jiang (2016) borrow the idea of pivot prediction from SCL and extend it to a neural network-based solution with auxiliary tasks. Ziser and Reichart (Ziser and Reichart 2018) incorporate the pivot to language modeling, the proposed PBLM and its improved version RF2 (Ziser and Reichart 2019) have demonstrated their superiority over a large number of previous approaches. Despite their promising results, they still cannot learn to select pivots.

Some efforts have been initiated to learn pivots with embedding based model. Li *et al.* (2017; 2018) incorporated attention mechanism into domain-adversarial learning (Ganin and Lempitsky 2015) to automatically identify the pivots. This method opens a new door for selecting pivots with adversarial attention. But the attention learns a weight for each unigram pivot and fails to learn higher-order n-gram pivots which play an important role in sentiment analysis (e.g. the polarity of bigram `not good` is completely different from unigram `good`). Similar problem was also studied in (Wang et al. 2018), uncovering that attention is not capable of inferring the dependency between words. In addition, attention networks require pre-trained word embeddings. As pointed out in recent research (He, Girshick, and Dollar 2018), pre-training may not be necessary for many situations.

Uncertainty of Bayesian Deep Learning: Bayesian deep learning offers a practical framework for understanding uncertainty with deep models (Kendall and Gal 2017; Gal and Ghahramani 2016). In Bayesian modeling, there are two main types of uncertainty (Der Kiureghian and Ditlevsen 2009): aleatoric uncertainty and epistemic uncertainty. Both of them can be estimated in regression and classification tasks. Epistemic uncertainty can be explained away given enough data and is often referred to as model uncertainty. However, aleatoric uncertainty (or heteroscedastic uncertainty specifically) depends on the model inputs and cannot be explained away. It is worth modeling aleatoric uncertainty in domain adaptation, since domain invariant features tend to show higher domain uncertainty.

In this paper, we build a model to learn the pivots and representations simultaneously. We envision the concept of *transferable pivot*, and propose a new approach that utilizes uncertainty (Grandvalet and Bengio 2004) to quantify the transferability of pivots. Our model can be trained from scratch not requiring external pre-trained word embeddings.

Approach

Notations and Model Overview

In this paper, we study sentiment classification in an unsupervised domain adaptation setting. Considering in source domain, we have a set of labeled data $\mathcal{D}_s^l = \{x_i^s, y_i^s\}|_{i=1}^{n_s^l}$ as well as some unlabeled data $\mathcal{D}_s^u = \{x_i^s\}|_{i=n_s^l+1}^{n_s^l+n_s^u}$, where $\mathcal{D}_s = \mathcal{D}_s^l \cup \mathcal{D}_s^u$. In a target domain, only a set of unlabeled data $\mathcal{D}_t = \{x_i^t\}|_{i=1}^{n_t}$ is available. We utilize mutual information (Blitzer, McDonald, and Pereira 2006) to initialize a pivot set V_p with size n_p.

In this section, we present a new approach to sentiment domain adaptation: Transferable Pivot Transformer (TPT). As shown in Figure 1, the Pivot Selector (top right) learns to detect transferable pivots while the Transferable Transformer jointly learns transferable representations with the aid of the selected pivots. The ultimate goal is to build a discriminative classifier on the final representations, that is trained with labeled data $\mathcal{D}_s^l$ and generalize to the target domain.

Transferable Transformer

Previous work PBLM (Ziser and Reichart 2018) conducts a representation learning procedure based on recurrent language modeling. The difference is that the recurrent language model aims to predict the next word conditioned on its previous context, while PBLM predicts whether the next token is a pivot or not. However, this unidirectional language model severely restricts the power of the model to capture contextual cues. Furthermore, the pivots are predicted not only by non-pivot words but also by pivot words, which means that their pivot prediction task may not sufficiently model the crucial relationship between the pivot and non-pivot features. As a consequence, the model tends to learn trivial features for pivot prediction.

To address these limitations, we propose the Transferable Transformer which uses a multi-layer bidirectional Transformer (Vaswani et al. 2017) to learn transferable representations. We set up a masked-pivot prediction task, inspired by BERT (Devlin et al. 2018), to establish the relationship between pivots and non-pivots. Given a sequence $U = \{u_1, \ldots, u_n\}$, we mainly mask the pivot tokens and keep the

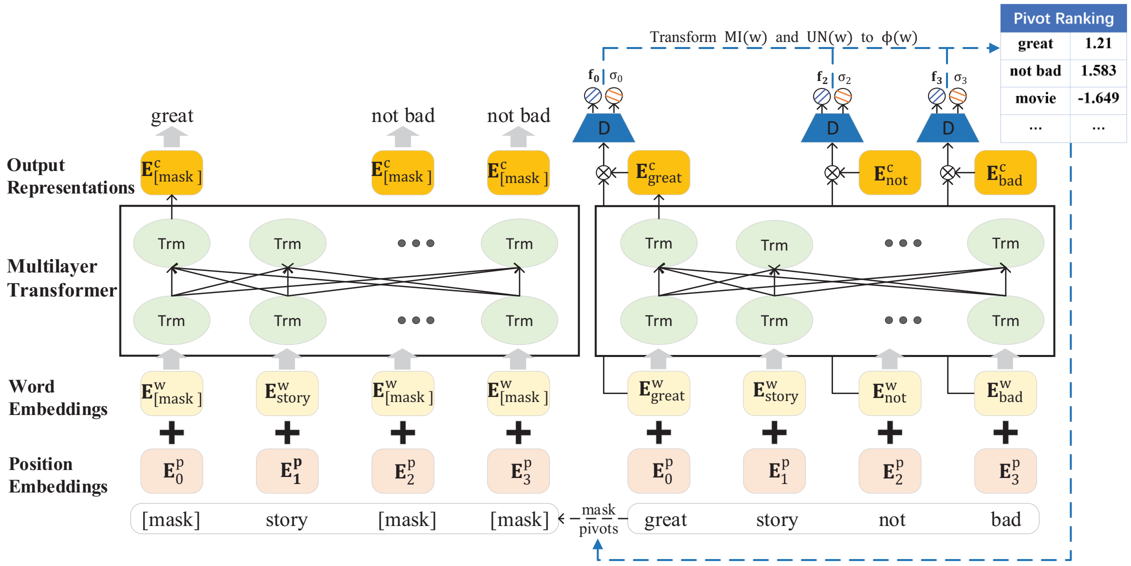


Figure 1: The architecture of the proposed Transferable Pivot Transformer (TPT). (**left**) Training objective used in Representation Learning with masked sequence as input. (**right**) Training Pivot Selector with original input.

remaining non-pivot ones unchanged in order to align the non-pivots with the pivots more strictly. The model applies a multi-head self-attention operation over the masked input context tokens $\widetilde{U}$ followed by position-wise feed-forward layers, and the final hidden vectors corresponding to the masked pivots are fed into an output softmax over the pivot vocabulary V_p. Hence, masked cross-entropy is a natural loss function to reason about all masked pivots included in U:

$$\mathcal{L}_{pivot}(\widetilde{U}) = \frac{1}{\sum m_i} \sum_{i=1}^{n} m_i \cdot L_p(\hat{\mathbf{y}}_i^p, \mathbf{y}_i^p) \qquad (1)$$

where $m_i \in \{0,1\}$ is the mask value and $\mathbf{y}_i^p \in [0,1,2,...,|V_p|]$ is the ground truth pivot index for the i-th transformed input token (0 stands for NONE). $L_p(\hat{\mathbf{y}}_i^p, \mathbf{y}_i^p)$ denotes the cross entropy loss for pivot prediction. Note that m_i and $\mathbf{y}_i^p$ are the same for two consecutive words constituting a bigram pivot.

As explained in (Devlin et al. 2018), the main downside of the bidirectional Transformer is the mismatch between training and inference for the [MASK] tokens. Since the masked pivots carry the most transferable and discriminative information for cross-domain classification, they must be observed during inference time. In this paper, the training scheme is designed as follows:

- For pivot words: most of the time are masked for prediction, with a small probability to be kept unchanged.
- For non-pivot words: most of the time are kept unchanged, with a small probability to be masked for prediction.

We also mask non-pivot tokens with a small probability, which enables the model to distinguish pivot and non-pivot tokens. The tokens for prediction may be replaced with a [MASK] token, or a random token, or the token itself in a predefined probability. Figure 1 provides an illustration of Transferable Transformer. The model predicts the masked pivots great and not bad which convey the relevant sentiment-related information. Note that even though bad is also a unigram pivot, it is regarded as a bigram pivot with higher priority.

Pivot Selector

Pivot Selector is used to learn pivots that transfer better across domains. The original SCL based on the traditional discrete features can help identify the words carrying the most significant sentiment signals in the source. Nevertheless, under the circumstance of adopting real-valued embeddings as word features in deep models, there is no guarantee that the contextual features of selected words act similarly in both domains. In this paper, we envision the idea of *transferable pivot*, as follows.

Definition 1 (Transferable Pivot) *A transferable pivot is an n-gram that behaves similarly for its contextual representation and discriminative learning in both domains.*

More intuitively, when using the powerful embedding based methods, it is very desirable to take the transferability of contextual representations into account. The Pivot Selector addresses these issues by learning to select those transferable pivots. To study the characteristics of transferable pivots, we visualize the Transformer contextual representations of some pivots and non-pivots in Figure 2. Similar to many adversarial domain adaptation approaches, we utilize a conditional domain discriminator learned to distinguish the source from the target domains given a word w. Its error function reflects well the discrepancy between feature condition distributions $P(f|w)$ and $Q(f|w)$. It can be observed that representations of pivots tend to show higher domain classification uncertainty than non-pivots.

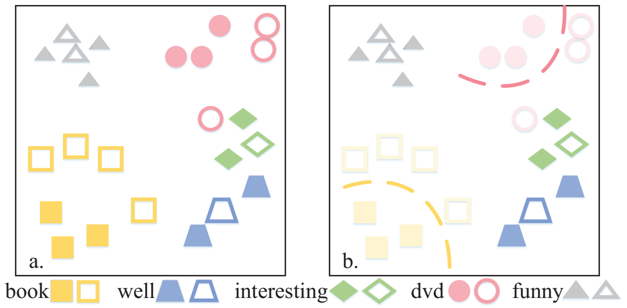


Figure 2: Transferable pivots. Suppose from domain Books (B filled with color) to domain DVDs (D not filled). (a) Patterns of the same color indicate representations of same word in different contexts from both domains. (b) The decision boundary of Bayesian domain discriminator (conditioned on specific word) separates non-pivot distribution easily with lower variance estimated by Bayesian classifier (indicated by the darkness of color) compared to pivot.

Algorithm 1 Pseudocode for the first stage of TPT

```
Require: $\mathcal{D}_s$, $\mathcal{D}_t$, $\mathcal{D}_{val}$
Require: $n_{step}$ = #steps to reduce $n_p$ per epoch
Require: $n_{min}$ = minimum size of $V_p$
  for epoch=1 to max-epoch do
    for min-batch $B_s$, $B_t$ in $\mathcal{D}_s$, $\mathcal{D}_t$ do
      $\widetilde{B}_s = \mathrm{MASK}(B_s)$, $\widetilde{B}_t = \mathrm{MASK}(B_t)$
      compute loss $\mathcal{L}_{pivot}$ on $\widetilde{B}_s \cup \widetilde{B}_t$
      update transformer parameters
    end for
    if $\mathcal{L}_{pivot}$ converges on $\mathcal{D}_{val}$ and $n_p > n_{min}$ then
      compute loss $\mathcal{L}_{dom}$ on $\mathcal{D}_s \cup \mathcal{D}_t$
      update discriminator parameters
      compute $\phi(w), w \in V_p$ on $\mathcal{D}_{val}$
      sort $V_p$ with $\phi(w)$ in descending order
      $n_p = n_p - n_{step}$
      $\widetilde{V}_p \leftarrow \{w_i \in V_p$, for $i \in [1, n_p]\}$
    end if
  end for
```

Motivation. These findings reveal that domain-invariant features make it hard to tell which domain they are from (intuitively shown in Figure 2). In other word, the domain classifier will be uncertain for its predictions. It thus motivates us to use the uncertainty of domain discriminator to aid the learning of transferable pivots.

Domain Uncertainty: With new Bayesian deep learning tools, it is possible to capture the uncertainty of deep models (Kendall and Gal 2017). In Bayesian modeling, there are two main types of uncertainty (Der Kiureghian and Ditlevsen 2009): epistemic uncertainty and aleatoric uncertainty. On one hand, out-of-data examples that can be identified with epistemic uncertainty (model uncertainty) are not suitable in this work since the model will be certain for its predictions under transductive transfer setting, where all data are available during training. On the other hand, aleatoric uncertainty (data uncertainty) captures the noise inherent in the observations which implies the domain uncertainty of pivots. We set up our target to choose the pivot such that it is discriminative with respect to sentiment label on the source domain while its contextual features are domain-invariant enough to fool the discriminator.

We achieve this by introducing a Bayesian domain discriminator D without propagating its gradient to the Transformer (top right of Figure 1). According to our definition, we measure the domain discrepancy of a word's context conditioned on its word embedding, which may follow different distributions. It may be better aligned by using a conditional discriminator (Long et al. 2018). Given a sequence of input tokens $U = \{u_1, \ldots, u_n\}$, we only compute the discriminator loss for pivot words indexed by its mask m_i. The discriminator D takes the contextual representation $\mathbf{E}_i^c$ as input with its word embedding $\mathbf{E}_i^w$ as condition, where $\otimes$ is a multilinear map that can be approximated by randomized methods to avoid dimension explosion:

$$\left[f_i, \sigma_i^2\right] = D(\mathbf{E}_i^w \otimes \mathbf{E}_i^c) \tag{2}$$

here f_i, σ_i^2 are the Bayesian network outputs, where f_i is the predictive mean of domain logit while σ_i^2 is the variance. To model the aleatoric uncertainty, the scoring prediction is given by a Gaussian distribution parameterized by f_i, σ_i^2:

$$\hat{d}_{i,t} = f_i + \sigma_i * \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, I) \tag{3}$$

We need to maximize the expected log likelihood:

$$\log \mathbb{E}_{\mathcal{N}(\hat{d}_i; f_i, \sigma_i^2)} \left[\delta(\hat{d}_i)^{d_i}(1 - \delta(\hat{d}_i))^{1-d_i}\right] \tag{4}$$

where $\delta(\cdot)$ is the sigmoid function, $d_i \in \{0, 1\}$ is domain label indicating source or target. As there is no analytic solution to integrate out this Gaussian distribution, we approximate the objective through Monte Carlo integration (Kendall and Gal 2017). The aleatoric uncertainty loss $L_a(\hat{d}_i, d_i)$ is then estimated as $-\log \frac{1}{T} \sum_t \left(\delta(\hat{d}_{i,t})^{d_i}(1 - \delta(\hat{d}_{i,t}))^{1-d_i}\right)$, where T is the number of Monte Carlo Simulations. The objective function of the Bayesian domain discriminator is formulated as follows:

$$\mathcal{L}_{dom}(U) = \frac{1}{\sum m_i} \sum_{i=1}^{n} m_i \cdot L_a(\hat{d}_i, d_i) \tag{5}$$

The aleatoric uncertainty is naturally derived from the predictive variance σ_i^2 of the discriminator. We define a new function $\phi(\cdot)$ incorporating both aleatoric uncertainty and mutual information (MI) to quantify pivot transferability:

$$\mathrm{UN}(w) = \frac{\sum_{(u_i = w)} \sigma_i^2}{\sum_i \mathbb{I}(u_i = w)} \tag{6}$$

$$\phi(w) = \lambda \cdot \widehat{\mathrm{UN}}(w) + \widehat{\mathrm{MI}}(w) \tag{7}$$

where λ is a hyper-parameter controlling the relative weight of the two criteria to tradeoff discriminability and domain uncertainty. $\widehat{\mathrm{UN}}(w)$ and $\widehat{\mathrm{MI}}(w)$ are standardized values respectively to avoid numerical scale differences.

TPT for Domain Adaptation

The above Transferable Transformer and Pivot Selector finally form our TPT (Transferable Pivot Transformer) model, which learns transferable pivots and the representations simultaneously. The ultimate goal is to build a sentiment classifier, which can be trained on the labeled source data and generalize well to the unlabeled target data. Like many previous unsupervised methods, TPT also follows a two-stage training procedure to be detailed as follows.

In the first stage of representation learning, TPT is trained with all data from the source and target domains. Algorithm 1 illustrates the training procedure. Firstly, we divide the whole data into training set and development set and maintain the size of the pivot set during training. For each epoch on training data, the discriminator is jointly trained if the current pivot set size n_p is larger than the threshold n_{min} and the pivot loss has converged on $\mathcal{D}_{val}$. On development data, we rank pivots from the original pivot set V_p by computing $\phi(w)$ which forms a newly generated pivot set $\widetilde{V}_p$. The selected pivots $\widetilde{V}_p$, presumed to be better aligned in both domains, facilitate the Transferable Transformer to learn domain-invariant representations by the masked-pivot prediction task $\mathcal{L}_{pivot}$. We stop this ranking process when n_p is smaller than the predefined threshold n_{min}. In the second stage, we feed the features from the Transferable Transformer to predict sentiment polarity. To validate the efficacy of the representations learned by TPT, we adopt a one-layer text CNN classifier (Kim 2014) as it is used in previous sentiment classification (Yu and Jiang 2016; Ziser and Reichart 2018) tasks for a fair comparison.

Experiments and Results

Experimental Setup

To facilitate direct comparison with previous work we experiment with the product review domains (Blitzer, Dredze, and Pereira 2007) of – Books (B), DVDs (D), Electronic (E), Kitchen (K) and airline services reviews (A) (20 ordered domain pairs), replicating the experimental setup (Ziser and Reichart 2018) (including baselines, design, and hyperparameter details). The airline service reviews have a larger domain gap with the product reviews. There are 1000 positive and 1000 negative labeled reviews in each domain and the remaining reviews form the unlabeled set. All the models are trained with the data from the source and the unlabeled data from the target. We test our model with 2000 labeled target domain data. The experiments are conducted in three setups: 12 domain pairs between amazon product datasets, 4 product to airline and 4 airline to product setups. The statistics of datasets are summarized in table 1.

Implementation Details

In our implementation, we follow the same pivot selection criterion used by previous work (Ziser and Reichart 2018). For example, on each domain pair of Amazon Product datasets, top 800 pivots ranked by mutual information with sentiment label of source domain are kept as initialized pivots for TPT and the minimum threshold n_{min} is set to 500. Due to the scale of our datasets, our Transferable Transformer is a 4-layer (128 or 256 dimensional self-attention states) structure. For the position-wise feed-forward networks, the inner states are 4 times the size of self-attention states. The word embedding matrix W_e and position embedding matrix W_p are randomly initialized from a uniform distribution $U[-0.2, 0.2]$. We use 50 Monte Carlo integration samples and keep fixing $\lambda = 0.1$ throughout all experiments. The CNN classifier takes the features from Transformer as input. We train TPT using RMSprop optimizer with learning rate set to 7e-4 and use Adam (Kingma and Ba 2014) for text convolutional network fine-tuning.

Models for Comparison

- **Source-only:** We consider LSTM (Hochreiter and Schmidhuber 1997) and CNN (Kim 2014) as two non-domain-adaptation baselines where the word embeddings are trained from scratch. As for LSTM, word embeddings are fed to the LSTM and the final hidden state is used for classification. As for CNN, 1-d convolution is applied to these embeddings and the features after max-pooling are used for classification.
- **LSTM-LM-CNN:** LSTM is pretrained with language model objective on unlabeled data. The CNN structure is the same as above.
- **SCL-MI (Blitzer, Dredze, and Pereira 2007):** SCL aims to learn a low-dimensional domain-invariant feature representation. The pivots are those words with the highest mutual information to the sentiment labels in the source domain. The same pivot and non-pivot selection criterion is employed for AE-SCL-SR and PBLM models.
- **MSDA (Chen et al. 2012):** This is one of the state-of-the-art method based on discrete input features, which learns a shared hidden representation by reconstructing pivot features from corrupted inputs.
- **MSDA-DAN (Ganin et al. 2016):** Ganin *et al.* have also applied their shallow version of DANN on the feature representation generated by MSDA. The new representation is the concatenation of the hidden features of MSDA and the original input.
- **AE-SCL-SR (Ziser and Reichart 2016):** AE-SCL-SR learns a non-linear function from non-pivot features to pivot features. The reconstruction matrix of the autoencoder is initialized with a word embedding model.
- **AMN (Li et al. 2017):** AMN learns document representation based on memory network and adversarial training, and requires well pre-trained word embeddings.

Domain	#Train	#Dev	#Unlabeled	Avg.Length
Books	1600	400	6000	156
DVD	1600	400	34741	171
Electronics	1600	400	13153	108
Kitchen	1600	400	16785	91
Airline	1600	400	39396	117

Table 1: Statistics of the Amazon Product and Airline reviews datasets.

Method	D→B	E→B	K→B	B→D	E→D	K→D	B→E	D→E	K→E	B→K	D→K	E→K	Avg
							Our method						
TPT	**83.3**	**76.2**	**78.2**	**85.8**	**81.1**	**81.9**	**81.2**	**80.5**	**88.2**	**86.1**	**83.4**	**88.3**	**82.9**
							Previous Work Models						
RF2	-	-	76.2	84.1	80.2	-	-	-	-	**86.1**	-	-	-
PBLM-LSTM	80.5	70.8	73.5	82.6	77.6	78.6	74.5	80.4	85.4	80.9	83.3	87.1	79.6
PBLM-CNN	82.5	71.4	74.2	84.2	75.0	79.8	77.6	79.6	87.1	82.5	83.2	87.8	80.4
AMN	76.9	73.2	72.9	83.5	75.2	77.8	76.4	77.9	85.5	79.1	78.8	85.2	78.5
AE-SCL-SR	77.3	71.1	73.0	81.1	74.5	76.3	76.8	78.1	84.0	80.1	80.3	84.6	78.1
MSDA	76.1	71.9	70.0	78.3	71.0	71.4	74.6	75.0	82.4	78.8	77.4	84.5	75.9
MSDA-DAN	75.0	71.0	71.2	79.7	73.1	73.8	74.7	74.5	82.1	75.4	77.6	85.0	76.1
SCL-MI	73.2	68.5	69.3	78.8	70.4	72.2	71.9	71.5	82.2	77.2	74.0	82.9	74.3
							No Domain Adaptation						
LSTM-LM-CNN	76.4	66.4	71.3	76.2	72.7	74.8	72.8	74.6	84.8	77.9	78.7	85.8	76.0
LSTM	69.2	67.9	67.5	72.8	68.1	66.2	65.9	68.3	78.2	72.1	70.5	80.6	70.6
CNN	71.2	65.6	66.5	73.6	67.1	70.8	69.6	69.7	79.9	72.7	72.6	80.6	71.6

Table 2: Accuracy of adaption between product review domains.

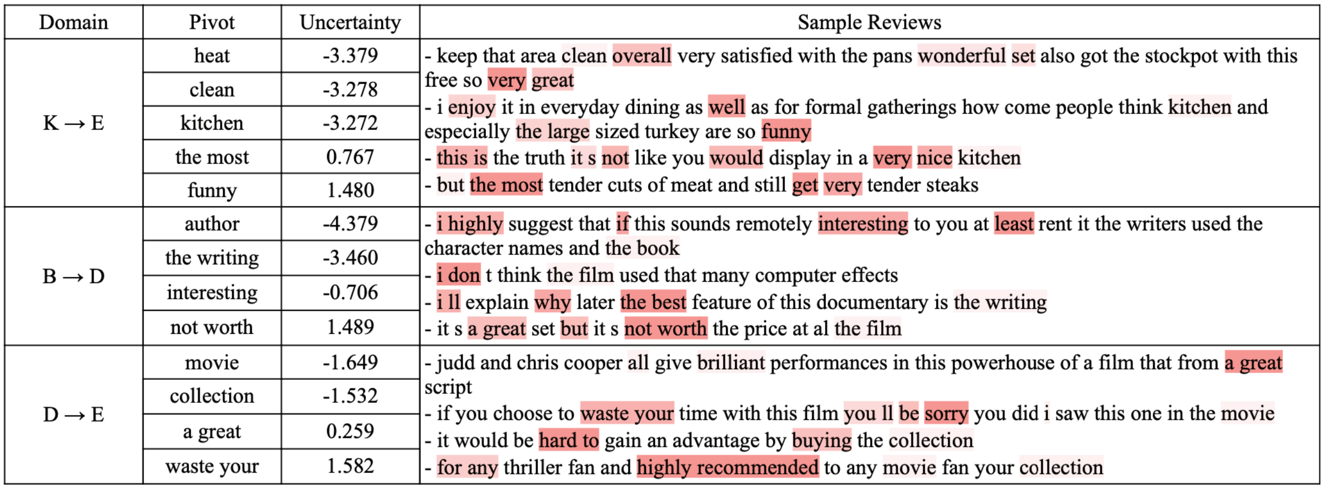
Method	B→A	D→A	E→A	K→A	Avg (P-Air)	A→B	A→D	A→E	A→K	Avg (Air-P)
					Our method					
TPT	**84.9**	**82.5**	**87.9**	86.9	**85.6**	**73.0**	**72.1**	**81.2**	**82.7**	**77.3**
					Previous Work Models					
RF2	-	-	-	86.1	-	72.3	-	-	-	-
PBLM-LSTM	83.7	81.0	87.7	**87.4**	85.0	70.3	71.1	80.5	82.6	76.1
PBLM-CNN	83.8	78.3	86.5	86.1	83.7	70.6	71.3	81.1	81.8	76.2
AMN	81.2	80.3	84.5	82.1	82.0	66.4	69.2	78.3	77.4	72.8
AE-SCL-SR	79.1	76.1	82.6	76.9	78.7	60.5	66.0	74.4	71.7	68.1
MSDA	72.2	73.3	75.1	76.8	74.3	58.5	61.0	70.6	69.0	64.8
MSDA-DAN	73.5	73.9	76.3	76.6	75.0	59.5	60.7	71.0	71.7	65.7
SCL	70.9	69.0	80.2	72.3	73.0	61.7	62.1	72.3	69.7	66.4
					No Domain Adaptation					
LSTM-LM-CNN	71.2	69.8	74.5	71.3	71.7	58.9	60.3	72.5	72.2	66.0
LSTM	68.3	65.0	72.1	68.6	67.3	56.7	57.3	66.2	65.0	61.3
CNN	67.6	66.7	72.0	70.0	69.1	56.3	59.0	66.0	66.6	62.0

Table 3: Accuracy of adaption between product (P) review domains and the airline (A) review domain.

- **PBLM-LSTM/CNN (Ziser and Reichart 2018):** The bottom two layers are Pivot Based Language Model (PBLM) built on LSTM, pre-trained before classification. The last layer is a task specific classifier (CNN or LSTM).
- **RF2 (Ziser and Reichart 2019):** RF2 is the improved version of PBLM-CNN enhanced by a curriculum learning algorithm. It iteratively trains the PBLM model, gradually increasing the information exposed about each pivot. Besides, The design of task specific classifier is the same as PBLM-CNN.

Main Results

Table 2 reports the classification accuracies of different methods on the Amazon product reviews dataset. Previous work (Ziser and Reichart 2018; 2019) provided very comprehensive results on typical sentiment domain adaptation methods under rigorous evaluation protocols, we thus adopted their published results for a direct and fair comparison. The proposed TPT model consistently achieves the best performance on 12 domain pairs. Results show that pivots matter a great deal to sentiment domain adaptation. The non-adapted CNN and LSTM baselines perform poorly compared with other methods, confirming that information from unlabeled data is beneficial for domain adaptation. To explore the importance of pivot, we pre-trained a LSTM language model and its classifier is also a CNN like non-adapted baseline methods. The experiment also shows that it is important to incorporate domain adaptation techniques in embedding based models. The results of non-adapted methods are not comparable with adapted baselines based on discrete word features. Compared with the previous best baseline PBLM, TPT outperforms PBLM-CNN and PBLM-LSTM by 2.5% and 3.3% on average. One reason is that we utilize bidirectional Transformer to capture the relationship between pivot words and non-pivot words other than recurrent language model, resulting in better transferable features across domains. Besides, the word-level domain discriminator can help detect more transferable pivots that are superior to previous pivots selected only by mutual information. The ablation study demonstrates that both model designs contribute to the final results. In the more challenging product to airline and airline to product setups, the above observations are similar. The lower performance of the non-adapted base-



Domain	Pivot	Uncertainty	Sample Reviews
K → E	heat	-3.379	- keep that area clean overall very satisfied with the pans wonderful set also got the stockpot with this free so very great - i enjoy it in everyday dining as well as for formal gatherings how come people think kitchen and especially the large sized turkey are so funny - this is the truth it s not like you would display in a very nice kitchen - but the most tender cuts of meat and still get very tender steaks
	clean	-3.278	
	kitchen	-3.272	
	the most	0.767	
	funny	1.480	
B → D	author	-4.379	- i highly suggest that if this sounds remotely interesting to you at least rent it the writers used the character names and the book - i don t think the film used that many computer effects - i ll explain why later the best feature of this documentary is the writing - it s a great set but it s not worth the price at al the film
	the writing	-3.460	
	interesting	-0.706	
	not worth	1.489	
D → E	movie	-1.649	- judd and chris cooper all give brilliant performances in this powerhouse of a film that from a great script - if you choose to waste your time with this film you ll be sorry you did i saw this one in the movie - it would be hard to gain an advantage by buying the collection - for any thriller fan and highly recommended to any movie fan your collection
	collection	-1.532	
	a great	0.259	
	waste your	1.582	

Figure 3: Visualization of aleatoric uncertainty on learned pivots and corresponding standardized values.

line reflects the larger domain gaps between product and airline reviews. TPT performs consistently better in 7 domain pairs, as reported in Table 3.

Ablation Study

We conduct further experiments to investigate the impact of word embeddings, Transformer, and transferable pivots. The results are shown in Table 4. The subscript $w2v$ indicates that the model embeddings are initialized with the well pre-trained word2vec[1]. First, we observe that word embeddings pre-trained from large scale datasets improve the performance of the non-adapted baseline, but little help for TPT since the embeddings and Transformer are jointly learned during training with highly correlated tasks for sentiment classification. Second, even TPT (w/o learning transferable pivot) outperforms all previous results, demonstrating the effectiveness of our Transferable Transformer in modeling the correlation between pivots and non-pivots. But increasing the pivot number does not seem to improve the classification accuracy. Besides, it is straightforward to see that learning representation with random mask (not only on pivots) is superior to recurrent language modeling. However, predicting pivots with random mask is prone to learn trivial features, making the pivot-prediction task suffer. In addition, we evaluate the impact of learning transferable pivots by applying the pivot learning strategy in transformer or using our selected pivots on the SCL baseline. The performance boost proves the effectiveness of learning transferable pivots.

Visualization

In order to validate that our model is able to discover the potential transferability of pivots, we visualize the aleatoric uncertainty of the pivot and its context. Figure 3 lists some example reviews and the darkness of the color indicates the uncertainty in Eq. (6). We take some pivots that transfer well or poorly as examples for demonstration. It can be observed that some pivots like `but`, `great` and `the best`, whose semantics are usually more consistent across domains, tend to have higher domain uncertainty. Some pivots like `movie`, `kitchen`, `the film` and `author` are either biased to the source or the target, therefore, the domain discriminator may be more certain on them. Compared to original mutual information, those frequent words are very high up the pivot list ranked by MI but may be filtered by the discriminator (e.g. `clean` is ranked 29 by MI, but 796 by uncertainty). Also, it is interesting to observe that the discriminator tends to be uncertain on bigram pivots. One reasonable explanation is that bigrams may convey more informative transferability than unigrams.

Model	Avg. Acc
CNN	71.6
CNN$_{w2v}$	73.8
Transformer (random mask)	79.5
TPT (w/o transferable pivot, n_p=500)	82.3
TPT (w/o transferable pivot, n_p=500)$_{w2v}$	82.1
TPT (w/o transferable pivot, n_p=800)	82.2
SCL (transferable pivot)	75.0
TPT (n_p=800, n_{min}=500)	**82.9**

Table 4: Comparison between different model variants on the Amazon Production Reviews Datasets.

Conclusion

Deep neural networks are widely used in sentiment classification but suffer from their dependency on large-scale labeled training data in a specific domain. In this paper, we incorporate pivots into representation learning and propose the TPT model for cross-domain sentiment classification. Unlike the previous works, TPT can simultaneously learn pivot and contextual representations, resulting in robust transfer performance. We have demonstrated through multiple experiments that it can better leverage unlabeled data compar-

[1] https://code.google.com/p/word2vec/

ing previous works, which shows the effectiveness of TPT.

Acknowledgments

This work was supported by the National Key R&D Program of China (2017YFC1502003) and Natural Science Foundation of China (61772299 and 71690231).

References

Blitzer, J.; Dredze, M.; and Pereira, F. 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*, 440–447.

Blitzer, J.; McDonald, R.; and Pereira, F. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, 120–128. Association for Computational Linguistics.

Chen, M.; Xu, Z.; Weinberger, K.; and Sha, F. 2012. Marginalized denoising autoencoders for domain adaptation. *arXiv preprint arXiv:1206.4683*.

Der Kiureghian, A., and Ditlevsen, O. 2009. Aleatory or epistemic? does it matter? *Structural Safety* 31(2):105–112.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Gal, Y., and Ghahramani, Z. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, 1050–1059.

Ganin, Y., and Lempitsky, V. S. 2015. Unsupervised domain adaptation by backpropagation. *international conference on machine learning* 1180–1189.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research* 17(1):2096–2030.

Glorot, X.; Bordes, A.; and Bengio, Y. 2011. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, 513–520.

Grandvalet, Y., and Bengio, Y. 2004. Semi-supervised learning by entropy minimization. 529–536.

He, K.; Girshick, R. B.; and Dollar, P. 2018. Rethinking imagenet pre-training. *arXiv: Computer Vision and Pattern Recognition*.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Kendall, A., and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, 5574–5584.

Kim, Y. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kingma, D., and Ba, J. 2014. Adam: A method for stochastic optimization. *Computer Science*.

Li, Z.; Zhang, Y.; Wei, Y.; Wu, Y.; and Yang, Q. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In *IJCAI*, 2237–2243.

Li, Z.; Wei, Y.; Zhang, Y.; and Yang, Q. 2018. Hierarchical attention transfer network for cross-domain sentiment classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. I. 2015. Learning transferable features with deep adaptation networks. *international conference on machine learning* 97–105.

Long, M.; Cao, Z.; Wang, J.; and Jordan, M. I. 2018. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, 1640–1650.

Pan, S. J.; Ni, X.; Sun, J.-T.; Yang, Q.; and Chen, Z. 2010. Cross-domain sentiment classification via spectral feature alignment. In *Proceedings of the 19th international conference on World wide web*, 751–760. ACM.

Tzeng, E.; Hoffman, J.; Zhang, N.; Saenko, K.; and Darrell, T. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474*.

Tzeng, E.; Hoffman, J.; Darrell, T.; and Saenko, K. 2015. Simultaneous deep transfer across domains and tasks. *international conference on computer vision* 4068–4076.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Wang, S.; Mazumder, S.; Liu, B.; Zhou, M.; and Chang, Y. 2018. Target-sensitive memory networks for aspect sentiment classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, 957–967.

Wu, F., and Huang, Y. 2016. Sentiment domain adaptation with multiple sources. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, 301–310.

Yu, J., and Jiang, J. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 236–246.

Ziser, Y., and Reichart, R. 2016. Neural structural correspondence learning for domain adaptation. *arXiv preprint arXiv:1610.01588*.

Ziser, Y., and Reichart, R. 2018. Pivot based language modeling for improved neural domain adaptation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, 1241–1251.

Ziser, Y., and Reichart, R. 2019. Task refinement learning for improved accuracy and stability of unsupervised domain adaptation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5895–5906.