Adversarial Category Alignment Network for Cross-domain Sentiment Classification

Xiaoye Qu1,* Zhikang Zou1,* Yu Cheng2 Yang Yang3 Pan Zhou1†

1Huazhong University of Science and Technology
2Microsoft AI & Research
3University of Electronic Science and Technology of China
{xiaoye, panzhou}@hust.edu.cn | yu.cheng@microsoft.com
{zhikangzou001, dlyyang}@gmail.com

Abstract
Cross-domain sentiment classification aims to predict sentiment polarity on a target domain utilizing a classifier learned from a source domain. Most existing adversarial learning methods focus on aligning the global marginal distribution by fooling a domain discriminator, without taking category-specific decision boundaries into consideration, which can lead to the mismatch of category-level features. In this work, we propose an adversarial category alignment network (ACAN), which attempts to enhance category consistency between the source domain and the target domain. Specifically, we increase the discrepancy of two polarity classifiers to provide diverse views, locating ambiguous features near the decision boundaries. Then the generator learns to create better features away from the category boundaries by minimizing this discrepancy. Experimental results on benchmark datasets show that the proposed method can achieve state-of-the-art performance and produce more discriminative features.

1 Introduction
Sentiment classification aims to automatically identify the sentiment polarity (i.e., positive or negative) of the textual data. It has attracted a surge of attention due to its widespread applications, ranging from movie reviews to product recommendations. Recently, deep learning-based methods have been proposed to learn good representations and achieved remarkable success. However, the performances of these works are highly dependent on manually annotated training data while annotation process is time-consuming and expensive. Thus, cross-domain sentiment classification, which aims to transfer knowledge learned on labeled data from related domains (called source domain) to a new domain (called target domain), becomes a promising direction.

One key challenge of cross-domain sentiment classification is that the expression of emotional tendency usually varies across domains. For instance, considering reviews about two sorts of products: Kitchen and Electronics. One set of reviews would contain opinion words such as “delicious” or “tasty”, and the other “rubbery” or “blurry”, to name but a few. Due to the small intersection of two domain words, it remains a significant challenge to bridge the two domains divergence effectively.

Researchers have developed many algorithms for cross-domain sentiment classification in the past. Traditional pivot-based works (Blitzer et al., 2007; Yu and Jiang, 2016) attempt to infer the correlation between pivot words, i.e., the domain-shared sentiment words, and non-pivot words, i.e., the domain-specific sentiment words by utilizing multiple pivot prediction tasks. However, these methods share a major limitation that manual selection of pivots is required before adaptation. Recently, several approaches (Sun et al., 2016; Zellinger et al., 2017) focus on learning domain invariant features whose distribution is similar in source and target domain. They attempt to minimize the discrepancy between domain-specific latent feature representations. Following this idea, most existing adversarial learning methods (Ganin et al., 2016; Li et al., 2017) reduce feature difference by fooling a domain discriminator. Despite the promising results, these adversarial methods suffer from inherent algorithmic weakness. Even if the generator perfectly fools the discriminator, it merely aligns the marginal distribution of the two domains and ignores the category-specific decision boundaries. As shown in Figure 1 (left), the generator may generate ambiguous or even mismatched features near the decision boundary, thus
hindering the performance of adaptation.

To address the aforementioned limitations, we propose an adversarial category alignment network (ACAN) which enforces the category-level alignment under a prior condition of global marginal alignment. Based on the cluster assumption in (Chapelle et al., 2009), the optimal predictor is constant on high density regions. Thus, we can utilize two classifiers to provide diverse views to detect points near the decision boundaries and train the generator to create more discriminative features into high-density region. Specifically, we first maximize the discrepancy of the outputs of two classifiers to locate the inconsistent polarity prediction points. Then the generator is trained to avoid these points in the feature space by minimizing the discrepancy. In such an adversarial manner, the ambiguous points are kept away from the decision boundaries and correctly distinguished, as shown in Figure 1 (right).

We evaluate our method on the Amazon reviews benchmark dataset which contains data collected from four domains. ACAN is able to achieve the state-of-the-art results. We also provide analyses to demonstrate that our approach can generate more discriminative features than the approaches only aligning global marginal distribution (Zhuang et al., 2015).

2 Related Work

Sentiment Classification: Deep learning based models have achieved great success on sentiment classification (Zhang et al., 2011). These models usually contain one embedding layer which maps each word to a dense vector, and different network architectures then process combined word vectors to generate a representation for classification. According to diverse network architectures, four categories are divided including Convolutional Neural Networks (CNNs) (Kalchbrenner et al., 2014; Kim, 2014), Recurrent Neural Networks (RNNs) (Yang et al., 2016; Zhou et al., 2016b), Recursive Neural Networks (RecNNs) (Socher et al., 2013) and other neural networks (Iyyer et al., 2015).

Domain Adaption: The fundamental challenge to solve the domain adaptation lies here is that data from the source domain and target domain have different distributions. To alleviate this difference, there are many pivot-based methods (Blitzer et al., 2007; He et al., 2011; Gouws et al., 2012; Yu and Jiang, 2016; Ziser and Reichart, 2018) which try to align domain-specific opinion (non-pivot) words through domain-shared opinion (pivot) words as the expression of emotional tendency usually varies across domains, which is a major reason of the domain difference. However, selecting pivot words for these methods first is very tedious, and the pivot words they find may not be accurate. Apart from pivot-based methods, denoising auto-encoders (Glorot et al., 2011; Chen et al., 2012; Yang and Eisenstein, 2014) have been extensively explored to learn transferable features during domain adaption by reconstructing noise input. Despite their promising results, they are based on discrete representation. Recently, some adversarial learning methods (Ganin et al., 2016; Li et al., 2017, 2018) propose to reduce this difference by minimizing the distance between feature distributions. But these methods solely focus on aligning the global marginal distribution by fooling a domain discriminator, which can lead to the mismatch of category-level features. To solve this issue, we propose to further align the category-level distribution by taking the decision boundary into consideration. Some recent works with class-level alignment have been explored in computer vision applications (Saito et al., 2017, 2018).

Semi-supervised learning: Considering the target samples as unlabeled data, our work is somehow related to semi-supervised learning (SSL). SSL has several critical assumptions, such as cluster assumption that the optimal predictor is constant or smooth on connected high density regions (Chapelle et al., 2009), and manifolds assumption that support set data lies on low-dimensional manifolds (Chapelle et al., 2009; Luo et al., 2017). Our work takes these assumptions to develop the approach.
3 Method

3.1 Problem Definition and Overall Framework

We are given two domains $D_s$ and $D_t$, denoting the source domain and the target domain respectively. $D_s = \{x_i^{(s)}, y_i^{(s)}\}_{i=1}^{n_s}$ are $n_s$ labeled source domain examples, where $x_i^{(s)}$ means a sentence and $y_i^{(s)}$ is the corresponding polarity label. $D_t = \{x_i^{(t)}\}_{i=1}^{n_t}$ are $n_t$ unlabeled target domain examples. In our proposed method, we denote $G$ as a feature encoder that extracts features from the input sentence. Then two classifiers $F_1$ and $F_2$ map these features to soft probabilistic outputs $p_1(y|x)$ and $p_2(y|x)$ respectively.

The goal is to train a model to classify the target examples correctly with the aid of source labeled data and target unlabeled data. To achieve this, we first train $G$, $F_1$ and $F_2$ to obtain global marginal alignment. This step reduces the distance between two domains but generates ambiguous target features near the decision boundary. Thus, $F_1$ and $F_2$ are adjusted to detect them by maximizing prediction discrepancy. After that, $G$ is trained to generate better features avoiding appearing near the decision boundary. The method also regularizes $G$ by taking the target data samples into consideration. In this way, we can achieve the category alignment. The proposed Adversarial Category Alignment Network (ACAN) is illustrated in Figure 2. The detailed training progress is described in Appendix D.

3.2 Marginal Distribution Alignment

To solve the domain adaption problem, we first consider minimize the classification error on the source labeled data for two classifiers:

$$L_{cls} = -\frac{1}{n_s} \sum_{i=1}^{n_s} \sum_{j=1}^{K} y_i^{(s)}(j) \log \tilde{y}_i^{(s)}(j) - \frac{1}{n_s} \sum_{i=1}^{n_s} \sum_{j=1}^{K} y_i^{(s)}(j) \log \tilde{y}_i^{(s)}(j)$$

$$\tilde{y}_i^{1i} = F_1(G(x_i^{(s)})) \quad \tilde{y}_i^{2i} = F_2(G(x_i^{(s)}))$$

where $K$ denotes the number of different polarities. In addition, similar to (Zhuang et al., 2015), our method tries to explicitly minimize the distance between the embedding features from both the source and the target domains. We adopt the Kullback–Leibler (KL) to estimate the distribution divergence:

$$L_{kl} = \sum_{i=1}^{n_s} g_s(i) \log \frac{g_s(i)}{g_t(i)} + \sum_{i=1}^{n_t} g_t(i) \log \frac{g_t(i)}{g_s(i)}$$

$$g_s' = \frac{1}{n_s} \sum_{i=1}^{n_s} G(x_i^{(s)}) \quad g_s = \frac{g_s'}{||g_s'||_1}$$

$$g_t' = \frac{1}{n_t} \sum_{i=1}^{n_t} G(x_i^{(t)}) \quad g_t = \frac{g_t'}{||g_t'||_1}$$

where $g_s$, $g_t \in \mathcal{R}^D$, $|| \cdot ||_1$ denotes L1 normalization. In this way, the latent network representations of two domains are encouraged to be similar. In other words, the marginal distribution is forced to be aligned.
3.3 Category-level Alignment

**Diverse Views:** Considering the marginal distribution alignment, there could be some ambiguous features near the decision boundary, which are easy to be incorrectly categorized into a specific class. If we alter the boundary of classifier $F_1$ and $F_2$, the samples closer to the decision boundary would have larger change. To explore these samples, we use $F_1$ and $F_2$ to provide diverse guidance. We define a discrepancy between probabilistic outputs of the two classifiers $p_1(y|x)$ and $p_2(y|x)$. The formula is:

$$L_{\text{dis}} = E_{x \sim D_1} [d(p_1(y|x), p_2(y|x))]$$ (3)

where $d(p_1(y|x), p_2(y|x))$ defines the average absolute difference for $K$ classes, which is:

$$d(p_1(y|x), p_2(y|x)) = \frac{1}{K} \sum_{i=1}^{K} |p_1(y|x) - p_2(y|x)|$$ (4)

Specifically, we first fix the generator $G$ and train the classifiers $F_1, F_2$ to detect points near the decision boundary by maximizing their discrepancy. The objective is as follows:

$$\max_{F_1,F_2} E_{x \sim D_1} \left[ \frac{1}{K} \sum_{i=1}^{K} |p_1(y|x) - p_2(y|x)| \right]$$ (5)

Then, this discrepancy is minimized by optimizing $G$ in order to keep these points away from the decision boundary and categorized into correct classes. The objective is as follows:

$$\min_{G} E_{x \sim D_1} \left[ \frac{1}{K} \sum_{i=1}^{K} |p_1(y|x) - p_2(y|x)| \right]$$ (6)

This adversarial step is repeated in the whole training process so that we can continuously locate non-discriminative points and classify them correctly, forcing the model to achieve category-level alignment on two domains.

3.4 Training Steps

The whole training procedure can be divided into three steps. In the first step, we consider both minimizing the classification error and marginal distribution discrepancy to achieve global marginal alignment. The loss function of this step can be written as:

$$L_1 = L_{\text{cls}} + \lambda_1 L_{\text{kl}}$$ (7)

In the second step, we consider increasing the difference of two classifiers $F_1$ and $F_2$ for the fixed $G$, thus the ambiguous features can be located by the diverse views. The loss function is defined as below:

$$L_2 = L_{\text{cls}} - \lambda_2 L_{\text{dis}}$$ (8)

$L_{\text{cls}}$ is used here to ensure the stability of the training process. $\lambda_2$ is a hyper-parameter controlling the range of classifiers. In the third step, the difference of two classifiers should be reduced for the fixed $F_1$ and $F_2$:

$$L_3 = L_{\text{cls}} + \lambda_3 L_{\text{dis}}$$ (9)

$L_{\text{cls}}$ and $\lambda_3$ used here are similar to the second step. We repeat this step $n$ times to balance the generator and two classifiers. After each step, the corresponding part of the network parameters will be updated. Algorithm 1 describes the overall training procedure.

**Algorithm 1** Training procedure of ACAN

**Require:** $D_s, D_t, G, F_1, F_2$

**Require:** $\lambda_1, \lambda_2, \lambda_3$, iteration number $n$

for $i \in [1, \text{max-epochs}]$

for minibatch $B^{(s)}, B^{(t)} \in D^{(s)}, D^{(t)}$

compute $L_{\text{cls}}$ on $[x_i \in B^{(s)}, y_i \in B^{(s)}]$

compute $L_{\text{kl}}$ on $[x_i \in B^{(t)}, x_j \in B^{(t)}]$

$L_1 = L_{\text{cls}} + \lambda_1 L_{\text{kl}}$

update $G, F_1, F_2$ by minimizing $L_1$

compute $L_{\text{cls}}$ on $[x_i \in B^{(s)}, y_i \in B^{(s)}]$,

compute $L_{\text{dis}}$ on $[x_i \in B^{(t)}, x_j \in B^{(t)}]$,

$L_2 = L_{\text{cls}} - \lambda_2 L_{\text{dis}}$

fix $G$, update $F_1, F_2$ by minimizing $L_2$

for $j \in [1, n]$

compute $L_{\text{dis}}$ on $[x_i \in B^{(s)}, y_i \in B^{(s)}]$,

compute $L_{\text{dis}}$ on $[x_i \in B^{(t)}, x_j \in B^{(t)}]$,

$L_3 = L_{\text{cls}} + \lambda_3 L_{\text{dis}}$

fix $F_1, F_2$, update $G$ by minimizing $L_3$

end for

end for

3.5 Generator Regularizer

To further enhance the feature generator, we introduce to regularize $G$ with the information of unlabeled target data. Generally, the mapping of $G(\cdot)$ can be seen a low-dimensional feature of the input. According to the manifolds assumption (Chapelle et al., 2009), this feature space is
Then the expected error on the source samples inputs of domain adaptation in (Ben-David et al., 2010). Specifically, the regularizer is formulated as follows:

$$R(G) = \sum_{x \in D_t} l_G(x_i, x_j)$$

where the expected error $\epsilon_t(h)$ is bounded by three terms: (1) the expected error on the source samples $\epsilon_s(h)$; (2) the divergence between the distributions $D_s$ and $D_t$; (3) the combined error of the ideal joint hypothesis $\lambda$.

For fixed $G$, sup can be replaced by max. Therefore, $F_1$ and $F_2$ are trained to maximize the discrepancy of their outputs and we expect $G$ to minimize this discrepancy. So we obtain

$$\min_G \max_{F_1, F_2} E_{x \sim D_t} [F_1 \circ G(x) - F_2 \circ G(x)]$$

The maximization of $F_1$ and $F_2$ is to provide diverse views, to find ambiguous points near the decision boundary, and the minimization of $G$ is to keep these points away from the decision boundary. To optimize Eq. 18, we assist the model to capture the whole feature space on the target domain better and achieve lower errors.

4 Experiments

4.1 Data and Experimental Setting

We evaluate the proposed ACAN on the Amazon reviews benchmark datasets collected by Blitzer (2007). It contains reviews from four different domains: Books (B), DVDs (D), Electronics (E), Kitchen appliances (K). There are 1000
### 4.2 Training Details and Hyper-parameters

In our implementation, the feature encoder $G$ consists of three parts including a 300-dimensional word embedding layer using GloVe (Pennington et al., 2014), a one-layer CNN with ReLU activation function adopted in (Yu and Jiang, 2016; He et al., 2018) and a max-over-time pooling through which final sentence representation is obtained. Specifically, the convolution filter and the window size of this one-layer CNN are 300 and 3 separately. Similarly, the classifier $F_1$ and $F_2$ can be decomposed into one dropout layer and one fully connected output layer. For the fully connected layer, we constrain the l2-norm of the weight vector, setting its max norm to 3. For the implementation of generator regularizer, we apply doubly stochastic sampling approximation due to the computational complexity.

The margin $m$ is set to 1 in this procedure. During training period, $\lambda_1, \lambda_2, \lambda_3, \lambda_4$, and $n$ are set to 5.0, 0.1, 0.1, 1.5, 2. Similar to (He et al., 2018), we parametrize $\lambda_4$ as a dynamic weight $\exp[-5(1 - \frac{t}{t_{max\text{-epochs}}}^2)]\lambda_4$. This is to minimize the effort of the regularizer as the predictor is not good at the beginning of training. We train 30 epochs for all our experiments with batch-size 50 and dropout rate 0.5. RMSProp (Tieleman and Hinton, 2012) optimizer with learning rate set to 0.0001 is used for all experiments.

### 4.3 Methods for Comparison

We consider the following approaches for comparisons (The URLs of previous methods code and data we use are in Appendix A):

- **SVM** (Fan et al., 2008): This is a non-domain-adaptation method, which trains a linear SVM on the raw bag-of-words representation of the labeled source domain.
- **AuxNN** (Yu and Jiang, 2016): This method uses two auxiliary tasks to learn sentence embeddings that works well across two domains. For fair comparison, we replace the neural model in this work with our CNN encoder.
- **DANN** (Ganin et al., 2016): This method exploits a domain classifier to minimize the discrepancy between two domains via adversarial training manner. We replace its encoder with our CNN-based encoder.
- **PBLM** (Ziser and Reichart, 2018): This is a representation learning model that exploits the structure of the input text. Specifically, we choose CNN as the task classifier.
- **DAS** (He et al., 2018): This method employs two regularizations: entropy minimization and self-ensemble bootstrapping to refine the classifier while minimizing the domain divergence.

### Table 1: Accuracy of adaptation on Amazon benchmark

All results are the averaged performance of each neural model by a 5-fold cross-validation protocol.

| Source $\rightarrow$ Target | Previous Work Models | ACAN Models |
|-----------------------------|----------------------|-------------|
|                             | SVM | AuxNN | DANN | PBLM | DAS | Baseline | ACAN-KL | ACAN-KM | ACAN |
| D $\rightarrow$ B           | 75.20 | 80.80 | 81.70 | 82.50 | 82.05 | 81.30 | 83.00 | 82.85 | 82.35 |
| E $\rightarrow$ B           | 68.85 | 78.00 | 78.55 | 71.40 | 80.00 | 79.50 | 80.30 | 79.80 | 79.75 |
| K $\rightarrow$ B           | 70.00 | 77.85 | 79.25 | 74.20 | 80.05 | 79.05 | 79.10 | 79.60 | **80.80** |
| B $\rightarrow$ D           | 77.15 | 81.75 | 82.30 | **84.20** | 82.75 | 82.50 | 83.85 | 83.25 | 83.45 |
| E $\rightarrow$ D           | 69.50 | 80.65 | 79.70 | 75.00 | 80.15 | 79.25 | 81.00 | 80.80 | **81.75** |
| K $\rightarrow$ D           | 71.40 | 78.90 | 80.45 | 79.80 | 81.40 | 79.10 | 80.15 | **82.25** | 82.10 |
| B $\rightarrow$ E           | 72.15 | 76.40 | 77.60 | 77.60 | 81.15 | 77.80 | 78.80 | 80.85 | **81.20** |
| D $\rightarrow$ E           | 71.65 | 77.55 | 79.70 | 79.60 | 81.55 | 78.00 | 81.30 | 82.75 | **82.80** |
| K $\rightarrow$ E           | 79.75 | 84.05 | 86.65 | **87.10** | 85.80 | 84.35 | 84.70 | 86.20 | 86.60 |
| B $\rightarrow$ K           | 73.50 | 78.10 | 76.10 | 82.50 | 82.25 | 78.00 | 77.30 | 81.00 | **83.05** |
| D $\rightarrow$ K           | 72.00 | 80.05 | 77.35 | **83.20** | 81.50 | 74.65 | 73.05 | 77.65 | 78.60 |
| E $\rightarrow$ K           | 82.80 | 84.15 | 83.95 | **87.80** | 84.85 | 81.05 | 83.70 | 83.70 | 83.35 |

Average: 73.66 | 79.85 | 80.29 | 80.40 | 81.96 | 79.55 | 80.48 | 81.78 | **82.15**
Baseline: Our baseline model is a non-adaptive CNN similar to (Kim, 2014), trained without using any target domain information, which is a variant of our model by setting $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ to zeros. ACAN-KL: ACAN-KL is a variant of our model which minimizes the distance between the features of two domains by minimizing the KL divergence. (set $\lambda_2 = \lambda_3 = \lambda_4 = 0$) ACAN-KM: ACAN-KM introduces the adversarial category mapping based on ACAN-KL without the regularizer. (set $\lambda_4 = 0$). ACAN: It is our full model.

4.4 Results

Table 1 shows the classification accuracy of different methods on the Amazon reviews, and we can see that the proposed ACAN outperforms all other methods generally. It is obvious to see that SVM performs not well in domain transferring task, beaten by Baseline. We can notice that exploring the structure of the input text (AuxNN and PBLM) brings some improvements over Baseline. However, these two pivot-based methods present relatively lower ability than DAS, which jointly minimizes global feature divergence and refines classifier. Compared to DAS, our proposed ACAN can improve 0.19% on the average accuracy. This can be explained by that we deal with the relationship between target features distribution and classifier more precisely. Finally, we conduct experiments on the variants of the ACAN. It is clear that the performances of Baseline, ACAN-KL, ACAN-KM and ACAN present a growing trend in most cases. Compared with ACAN-KL, ACAN achieves large gain from 80.48% to 82.15%, showing the effectiveness of category-level alignment.

4.5 Case Study

To better understand the results of different models, we conduct experiments on task B $\rightarrow$ E. For each sentiment polarity, we first extract the most related CNN filters according to the learned weights of the output layer in classifier $F_1$. Since all listed models use a window size of 3, the outputs of CNN with the highest activation values correspond to the most useful trigrams. As shown in Table 2, we identify the top trigrams from 10 most related CNN filters on the target domain. It is obvious that Baseline and ACAN-KL are more likely to capture the domain-independent words, such as “pointless”, “disappointing” and “great”. Thus, the performance of these two models drops much when applied to the target domain. Besides, DAS can capture more words of the target domain, but it is limited to nouns with less representativeness, such as “receiver”, “product” and etc. Compared to them, ACAN is able to extract the domain-specific words like “flawlessly” and “rechargeable”. These results are consistent with the accuracy of each model’s predictions. We also conduct experiments on the tasks B $\rightarrow$ K and K $\rightarrow$ D. Due to the space limitations, the results are presented in Appendix B.

4.6 Visualization of features

For more intuitive understanding of the differences between the global marginal alignment and category alignment, we further perform a visualization of the feature representations of the ACAN-KL and ACAN model for the training data in the source domain and the testing data in the target domain for the K $\rightarrow$ E task. As can be seen in Figure 3, global marginal alignment causes ambiguous
| Method   | Negative Sentiment                                                                 | Positive Sentiment                                                                 |
|----------|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| Baseline | audio-was-distorted, is-absolutely-pointless, *very-disappointing, waste-of-money, was-point-most, an-unsupported-config, an-extremely-disappointed, author-album-etc, cure-overnight-headphones, aa-rechargeable-batteries | wep-encryption-detailed, totally-wireless-headset, best-i, love-it-i, again-period-i, beautifully-great-price, awesome-accurate-sound, beautifully-designed-futuristic, wonderful-product-i, glad-i-purchased |
| ACAN-KL  | totally-useless-method, audio-was-distorted, *very-weak, *very-disappointing, extra-ridiculous-buttons, hopeless-mess-no, now-as-useless, waste-of-cash, is-absolutely-pointless, manual-is-useless | gift-i-love, uniden-cordless-telephone, a-journey-to, totally-wireless-headset, your-own-frequencies, a-gift-excellent, exceptional-being-rechargeable, gorgeous-picture-excellent, with-wireless-security, beautifully-designed-futuristic |
| DAS      | receiver-was-faulty, defective-product-i, is-useless-i, do-not-waste, did-not-work, very-poor-quality, the-crippy-keyboard, just-too-weak, is-absolutely-pointless, very-stupid-design | is-an-excellent, excellent-monitor-with, is-very-nice, truly-excellent-headphones, an-incredible-sound, advanced-technology-incredible, this-is-an, !-highly-recommended show-very-easy, picture-is-fabulous |
| ACAN     | very-poorly-designed, garbage-im-sorry, handed-was-defective, receiver-was-faulty, *very-disappointing, audio-was-distorted, dirty-and-scratched, extra-ridiculous-buttons, cartridges-are-incompatible, awful-absolutely-horrible | performs-flawlessly-hours, a-gift-excellent, beautifully-great-price, encryption-detailed-monitoring, fit-excellent-sound, !-very-happy, exceptional-being-rechargeable, beautifully-designed-futuristic, smooth-accurate-tracking, digital-camera-during |

Table 2: Comparison of the top trigrams chosen from 10 most related CNN filters learned on the task $B \rightarrow E$. The entire table contains the results achieved by the variants of our method. * denotes a padding. The domain-specific words are in bold.

**Figure 4:** The influence of the number of labeled target data on the task $E \rightarrow D$ and $B \rightarrow E$.

**Figure 5:** The training process of four ACAN model variants on the task $K \rightarrow E$.

features locating between two clusters while category alignment effectively projects these points into clusters, thus leading a more robust classification result. We also conduct experiments on the tasks $B \rightarrow E$ and $B \rightarrow K$. Due to the space limitations, the results are presented in Appendix C.

### 4.7 Model Analysis

In this part, we provide analysis to our proposed ACAN variants. In Figure 4, we show the comparison between Baseline and ACAN under a setting that some labeled target data are randomly selected and mixed with training data. Here, we present results on two transferring tasks while a similar tendency can be observed in other pairs. With an increase in the number of randomly selected labeled target data, the difference between the two models gradually decreases and ACAN also progressively obtains better results. These trends indicate that our ACAN is more effective with no or little-labeled target data and can further benefit from more labeled target data. In Figure 5, we can easily observe that ACAN continuously shows better results during the whole training process among four settings. After some epochs, ACAN-KL starts presenting lower testing accuracy than Baseline. One possible reason is that those categories which are initially well aligned between the source and target may be incorrectly mapped because of ignoring category-level feature distribution. This observation can prove our motivation in some degree.

### 5 Conclusion

In this paper, we propose a novel approach, which utilizes diverse view classifiers to achieve category-level alignment for sentiment analysis. Unlike previous works, we take the decision boundary into consideration, thus classifying the
target samples correctly into the corresponding category. Experiments show the proposed ACAN significantly outperforms state-of-the-art methods on the Amazon benchmark. In future we would like to adapt our method to other domain adaptation tasks and consider more effective alternatives for the generator regularizer.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61602197, and in part by the Fundamental Research Funds for the Central Universities, HUST: 2016YXMS085.

References

Mikhail Belkin and Partha Niyogi. 2003. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput., 15(6):1373–1396.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. Machine Learning, 79(1):151–175.

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 for Computational Linguistics, pages 440–447. Association for Computational Linguistics.

Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2009. Semi-supervised learning (chapelle, o. et al., eds.: 2006)[book reviews]. IEEE Transactions on Neural Networks, 20(3):542–542.

Minmin Chen, Zhixiang Xu, Kilian Weinberger, and Fei Sha. 2012. Marginalized denoising autoencoders for domain adaptation. arXiv preprint arXiv:1206.4683.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. Journal of machine learning research, 9(Aug):1871–1874.

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 machine learning (ICML-11), pages 513–520.

Stephan Gouws, GJ Van Rooyen, MIH Medialab, and Yoshua Bengio. 2012. Learning structural correspondences across different linguistic domains with synchronous neural language models. In Proc. of the xLite Workshop on Cross-Lingual Technologies, NIPS.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahnmeier. 2018. Adaptive semi-supervised learning for cross-domain sentiment classification. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3467–3476. Association for Computational Linguistics.

Yulan He, Chenghua Lin, and Harith Alani. 2011. Automatically extracting polarity-bearing topics for cross-domain sentiment classification. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 123–131. Association for Computational Linguistics.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1681–1691. Association for Computational Linguistics.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 655–665. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751. Association for Computational Linguistics.

Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018. Hierarchical attention transfer network for cross-domain sentiment classification. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI 2018, New Orleans, Louisiana, USA, February 2–7, 2018.

Zheng Li, Yu Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017. End-to-end adversarial memory network for cross-domain sentiment classification. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI 2017).
Yucen Luo, Jun Zhu, Mengxi Li, Yong Ren, and Bo Zhang. 2017. Smooth neighbors on teacher graphs for semi-supervised learning. *arXiv preprint arXiv:1711.00238*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. *Glove: Global vectors for word representation*. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics.

Kuniaki Saito, Yoshiataka Ushiku, Tatsuya Harada, and Kate Saenko. 2017. Adversarial dropout regularization. *arXiv preprint arXiv:1711.01575*.

Kuniaki Saito, Kohei Watanabe, Yoshiataka Ushiku, and Tatsuya Harada. 2018. Maximum classifier discrepancy for unsupervised domain adaptation. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3723–3732.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. *Recursive deep models for semantic compositionality over a sentiment treebank*. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642. Association for Computational Linguistics.

Baochen Sun, Jiashi Feng, and Kate Saenko. 2016. Return of frustratingly easy domain adaptation. In *AAAI*.

T. Tieleman and G. Hinton. 2012. *Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude*. COURSERA: Neural Networks for Machine Learning.

Jing Wang, Yu Cheng, and Rogério Schmidt Feris. 2016. Walk and learn: Facial attribute representation learning from egocentric video and contextual data. In *CVPR*, pages 2295–2304. IEEE Computer Society.

Yi Yang and Jacob Eisenstein. 2014. *Fast easy unsupervised domain adaptation with marginalized structured dropout*. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 538–544. Association for Computational Linguistics.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. *Hierarchical attention networks for document classification*. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1480–1489. Association for Computational Linguistics.

Jianfei Yu and Jing Jiang. 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*, pages 236–246. Association for Computational Linguistics.

Werner Zellinger, Thomas Grubinger, Edwin Lughofer, Thomas Natschläger, and Susanne Saminger-Platz. 2017. Central moment discrepancy (CMD) for domain-invariant representation learning. *arXiv preprint arXiv:1702.08811*.

Kunpeng Zhang, Yu Cheng, Yusheng Xie, Daniel Honbo, Ankit Agrawal, Diana Palsetia, Kathy Lee, Wei-keng Liao, and Alok N. Choudhary. 2011. SES: sentiment elicitation system for social media data. In *ICDM*, pages 129–136. IEEE Computer Society.

Guangyou Zhou, Zhiwen Xie, Jimmy Xiangji Huang, and Tingting He. 2016a. *Bi-transferring deep neural networks for domain adaptation*. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 322–332. Association for Computational Linguistics.

Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. 2016b. *Text classification improved by integrating bidirectional lstm with two-dimensional max pooling*. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3485–3495. The COLING 2016 Organizing Committee.

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*.

Yftah Ziser and Roi Reichart. 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)*, pages 1241–1251. Association for Computational Linguistics.
### A URLs of Data and Code

Here, we provide a list of URLs about the dataset and the code of the previous methods we compare.

- The Amazon product review dataset gathered by Blitzer et al (2007): [http://jmcauley.ucsd.edu/data/amazon/](http://jmcauley.ucsd.edu/data/amazon/)

- Code for AuxNN (Yu and Jiang, 2016): [https://github.com/jefferyYu/Learning-Sentence-Embeddings-for-cross-domain-sentiment-classification](https://github.com/jefferyYu/Learning-Sentence-Embeddings-for-cross-domain-sentiment-classification)

- Code for DANN (Ganin et al., 2016): [https://github.com/pumpikano/tf-dann](https://github.com/pumpikano/tf-dann)

- Code for PBLM (Ziser and Reichart, 2018): [https://github.com/yftah89/PBLM-Domain-Adaptation](https://github.com/yftah89/PBLM-Domain-Adaptation)

- Code for DAS (He et al., 2018): [https://github.com/ruidan/DAS](https://github.com/ruidan/DAS)

### B Trigram Full Results

In the paper, Table 2 shows the top trigrams chosen from 10 most related CNN filters learned on the task $B \rightarrow E$ by the DAS the and variants of the ACAN. For a more comprehensive presentation, we also conduct experiments on the task $B \rightarrow K$ and $K \rightarrow D$, and the results are listed in Table 3 and Table 4 respectively. It is obvious that the proposed ACAN is better to capture domain-specific words, compared to its variants and DAS.

### C Visualization full results

In this paper, Figure 3 visualizes the feature representations of the ACAN-KL and ACAN model for the training data in the source domain and the testing data in the target domain for the $K \rightarrow E$ task. For a more comprehensive presentation, we also conduct experiments on the task $B \rightarrow E$ and $B \rightarrow K$, and the results are listed in Figure 7 and Figure 8 respectively. It is obvious that global marginal alignment causes ambiguous features locating between two clusters while category alignment effectively projects these points into clusters.

### D Detailed Illustration of Training Phase

The overview of the propose ACAN is shown in Figure 2. For a better understanding, we present the changes of decision boundaries and data distribution during the network training process, shown in Figure 6. First, we train $F_1$ and $F_2$ to locate the points near the decision boundary by maximizing their discrepancy. Then, we train $G$ to minimize the discrepancy to achieve category-level alignment. At the same time, the generator $G$ is regularized with data from target domain.
Table 4: Comparison of the top trigrams chosen from 10 most related CNN filters learned on the task $B \rightarrow K$. The entire table contains the results achieved by the variants of our method. * denotes a padding. The domain-specific words are in bold.

| Method   | Negative Sentiment                                                                 | Positive Sentiment                                                                 |
|----------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| Baseline | rice also disappointing, be such shoddy, basically worthless, does n’t toast, waste your time, is totally useless, waste of time, was sorely disappointed, safe stainless versus, were very dull | sophisticated gorgeous retro, lodge properly packaged, delonghi cooked pretty, this stunning slice, beautifully i highly, perfection i, i, an excellent performer, beautiful shape, i, your cooking equipment, * highly recommend |
| ACAN-KL  | totally useless and, rice also disappointing, be such shoddy, flatware is unavailable, misleading advertising, i, waste of time, poorly made expensive, were very dull, makes weak coffee, was sorely disappointed | dishwaher nonstick, * highly get a, this stunning slice, * highly recommend, grilled meats and |
| DAS      | was very disappointing, shoddy junk garbage, totally useless and, thermometer very disappointing, disappointing coffee maker, by dimly brittle, do not waste, waste of time, very disappointing and, be lukewarm disgusting | makes wonderful tasting, this beautiful pan, is an excellent, sophisticated gorgeous retro, is highly recommend, awesome i, *, makes great coffee, and versatile pan, also highly recommend, it is great |
| ACAN     | kettle was leaking, totally useless and, rice also disappointing, flatware is unavailable, was sorely disappointed, waste of money, flat crooked ugly, is no metal, now basically worthless, makes weak coffee | sophisticated gorgeous retro, great hot drinks, it a learning, * highly recommend, great grilled sandwiches, * highly recommend, it toasts beautifully, look wonderful and, excellent addition to, nonstick i, you |

Table 3: Comparison of the top trigrams chosen from 10 most related CNN filters learned on the task $B \rightarrow K$. The entire table contains the results achieved by the variants of our method. * denotes a padding. The domain-specific words are in bold.

| Method   | Negative Sentiment                                                                 | Positive Sentiment                                                                 |
|----------|------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| Baseline | beyond is badly, such gross audio, returning for the, poorly executed poorly, director john ford, is so disappointing, does not work, is a disappointment, lighting poor directing, pathetic remake from | hip hop dvd, combines multiple genres, the most amazing, very good price, stylish photography, best performances since, amazing i, the, lasting and unique, accomplished and dedicated, loves being able |
| ACAN-KL  | return an even, such gross audio, star hollywood material, poorly executed poorly, does not work, a complete failure, is a disappointment, pathetic remake from, waste of money, failed miserably alson | beautifully classic comedy, adult who enjoys, i bought loves, the most acclaimed, combines multiple genres, very good price, ’s memorable entrance, great and splendid, acclaimed romantic comedies, stylish photography and |
| DAS      | release the movie, a total waste, dodging bullets and, incompetent direction by, is absolutely horrible, very disappointing once, is pretty pathetic, was a waste, awful the ending, pathetic remake from | entertaining and inspirational, truly enjoy it, an amazing artist, superb production and, very good price, fantastic films !, best performance since, perfect love on, amazing film from, and fascinating documentaries |
| ACAN     | unfunny overrated movie, of gross caricature, was a waste, pathetic remake from, poorly executed poorly, was awful from, directing poor writing, is a disappointment, disgusting badly written, tasteless unoriginal drivel | combines multiple genres, very good price, are great featuring, accomplished and dedicated, great performance tongue, ’s stylish photography, is my favourite, family classics action, the most amazing, fantastic action picture |

Table 4: Comparison of the top trigrams chosen from 10 most related CNN filters learned on the task $K \rightarrow D$. The entire table contains the results achieved by the variants of our method. * denotes a padding. The domain-specific words are in bold.

Figure 6: The detail of changes in decision boundaries and data distribution during the network training process.
Figure 7: Visualization by applying principal component analysis to the representation of source training data and target testing data produced by ACAN-KL (left) and ACAN (right) for B $\rightarrow$ E task. The red, blue, green, and black points denote the source positive, source negative, target positive, and target negative examples correspondingly.

Figure 8: Visualization by applying principal component analysis to the representation of source training data and target testing data produced by ACAN-KL (left) and ACAN (right) for B $\rightarrow$ K task. The red, blue, green, and black points denote the source positive, source negative, target positive, and target negative examples correspondingly.