



# Category-Level Adversarial Network for Cross-Domain Sentiment Classification

Shaokang Zhang<sup>1,2</sup>, Huailiang Peng<sup>1,2</sup>(✉), Yanan Cao<sup>1,2</sup>, Lei Jiang<sup>1,2</sup>,  
Qiong Dai<sup>1,2</sup>, and Jianlong Tan<sup>1,2</sup>

<sup>1</sup> Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China  
{zhangshaokang, penghuailiang, caoyanan, jianglei, daiqiong,  
tanjianlong}@iie.ac.cn

<sup>2</sup> School of Cyber Security, University of Chinese Academy of Sciences,  
Beijing, China

**Abstract.** Cross-domain sentiment classification utilizes useful information in the source domain to improve the sentiment classification accuracy in the target domain which has few or no labeled data. Most existing methods based on single domain classifier only consider the global alignment without taking category-level alignment into consideration, which can lead to the mismatch of category-level features and reduce classification accuracy. To solve the above problem, we propose the Category-level Adversarial Network (CAN). On the basis of single domain classifier, CAN adds  $K$  category-wise domain classifiers which can achieve fine-grained alignment of different data distributions by combining the label information and document representations. Specifically, we obtain document representations by introducing transferable attention network which mirrors the hierarchical structure of documents and transfers attentions across domains. Experiments results demonstrate that CAN model outperforms state-of-the-art methods on the Amazon and Airline datasets.

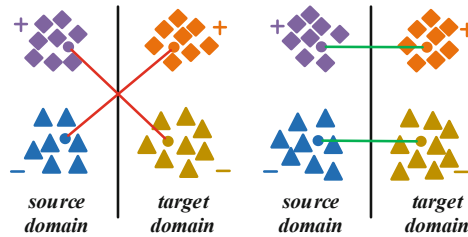
**Keywords:** Category-level Adversarial Network · Cross-domain sentiment classification · Domain adaptation · Hierarchical Attention

## 1 Introduction

Sentiment classification, which identifies the sentiment polarity of the review or a sentence, has attracted more and more research attention over the past decades. Traditional sentiment classification methods generally have good performance for a specific domain with abundant labeled data [1–3]. However, because labeling data is expensive and time-consuming, many domains lack of sufficient labeled data, which make traditional methods don't work well.

To address the problem, cross-domain sentiment classification has been proposed. It uses the knowledge from source domain with sufficient labeled data to enhance the prediction accuracy of target domain with few or no labeled

data. Researchers have proposed many methods to solve the cross-domain sentiment classification problem. Learning domain-shared and domain-specific features have been presented, which utilize the words with high co-occurrence in different domains and domain-independent words [4, 5]. While these methods require to manually extract domain-independent words. Recently, some methods can learn better sample features by deep neural network [6–9]. Domain-Adversarial training of Neural Networks (DANN) [8] which adds adversarial mechanism into the training of deep neural network. It introduces a domain classifier which can minimize the discrepancy between the source and target domain by gradient reversal. Most of the previous efforts only focus on aligning the global marginal distribution, while ignoring the category-level alignment. As shown in Fig. 1 (left), the positive/negative data aligns the negative/positive data from different domains. This mismatch promotes negative transfer and reduces classification accuracy.



**Fig. 1.** “+” and “-” denote positive and negative samples respectively. **Left:** domain adaptation without category-level alignment. **Right:** domain adaptation with category-level alignment

To overcome the sample mismatch issue, we propose the Category-level Adversarial Network (CAN) for cross-domain sentiment classification. CAN achieves the category-level alignment by introducing the category-wise domain classifiers, as shown in Fig. 1 (right). CAN constructs category-level adversarial network by combining the label information and document representations. This method can decide how much each document should be sent to the category-wise domain classifiers by utilizing the probability distribution over the label space. Besides, the word with sentiment polarity usually has higher contribution for document representation. CAN utilizes the hierarchical attention transfer mechanism, which automatically transfers word-level and sentence-level attentions across domains. In summary, the main contributions of our work are summarized as follows:

- We introduce the category-level information to achieve fine-grained alignment of different data distributions.
- We propose a CAN method which achieves category-level alignment. It adds the category-wise domain classifiers which joint the label information and

document representations. Besides, the hierarchical attention transfer mechanism can transfer attentions by assigning different weights to words and sentences.

- The experimental results clearly demonstrate that our method outperforms other state-of-the-art methods.

## 2 Related Work

**Domain Adaptation:** Domain adaptation has a large number of works in natural language processing over the past decades. Among them, Blitzer et al. [4] proposed the Structural Correspondence Learning (SCL) which produces correspondences among the features from different domains. Pan et al. [5] proposed the Spectral Feature Alignment (SFA) which solves mismatch of data distribution by aligning domain-specific words. Unfortunately, these methods mentioned above highly rely on manually selecting domain-shared features.

Recently, deep learning methods have obtained better feature representations for cross-domain sentiment classification. Glorot et al. [6] proposed the Stacked Denoising Auto-encoders (SDA) which successfully learns feature representations of a document from different domains [6]. [7] Chen et al. proposed Marginalized Stacked Denoising Autoencoder (mSDA) which reduces computing cost and improves the scalability to high-dimensional features. Kim, Yu et al. [10, 11] used two auxiliary tasks to produce the sentence embedding based on convolutional neural network. DANN leverages the adversarial training method to produce feature representations [8]. Li et al. [12] proposed the Adversarial Memory Network (AMN) which automatically obtains the pivots by using attention mechanism and adversarial training. Li et al. [9] proposed Hierarchical Attention Transfer Network (HATN) which transfers word-level and sentence-level attentions. Zhang et al. [13] proposed Interactive Attention Transfer Network (ITAN) which provides an interactive attention transfer mechanism by combining the information of sentence and aspect. Peng et al. [14] proposed the CoCMD which simultaneously extracts domain specific and invariant representations. Sharma et al. [15] proposed a method which can identify transferable information by searching significant consistent polarity (SCP) words. But these methods only align the global marginal distribution by fooling domain classifiers, which bring the category-level mismatch. To solve this problem, we align the category-level distribution by adding the label information.

**Attention Mechanism:** The contribution of each word in a document is different. To address this problem, attention mechanism is also used in many other tasks, such as machine translation [16], sentiment analysis [3], document classification [17], question prediction [18]. Besides, the hierarchical structure has superior performance than word-level attention which captures better feature representations since it expresses the hierarchical structure of the document.

### 3 Category-Level Adversarial Network

In this section, we first introduce the problem definition of cross-domain sentiment classification, followed by an summary of the model. Finally we present the details of CAN model.

#### 3.1 Problem Definition

We assume that there are two domains  $D_s$  and  $D_t$  which denote a source domain and a target domain respectively. We further suppose that we give a set of labeled training data  $\mathbf{X}_s^l = \{x_s^i, y_s^i\}_{i=1}^{N_s^l}$  and unlabeled training data  $\mathbf{X}_s^u = \{x_s^j\}_{j=N_s^l+1}^{N_s}$  from the source domain, where  $N_s^l$  is the number of labeled data and  $N_s$  is the all data of source domain. Besides, we give a set of unlabeled training data  $\mathbf{X}_t = \{x_t^j\}_{j=1}^{N_t}$  in the target domain, where  $N_t$  is the number of unlabeled data. The goal of cross-domain sentiment classification is training a robust model on labeled data and adapts it to predict the sentiment label on unlabeled data.

#### 3.2 An Overview of CAN

We introduce the overview of the CAN as shown in Fig. 2. Firstly, we obtain document representations by Transferable Attention Network (TAN). Then we utilize the Category-level Adversarial (CA) classifiers which combine the label information and document representations on adversarial process. Finally, we use the data in source domain and unlabeled data in target domain to train CA classifiers. Meanwhile, we train sentiment classifier using the labeled data. Finally, TAN and sentiment classifier predict the sentiment label.

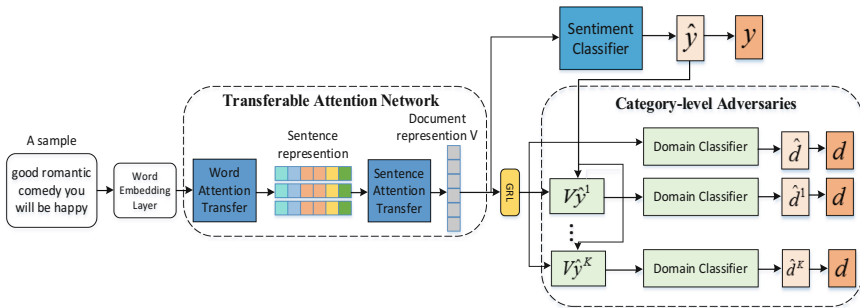


Fig. 2. The architecture of the CAN model, where  $\hat{y}$  is the predicted sentiment label and  $\hat{d}$  is the predicted domain label;  $y$  and  $d$  are the ground truth.

### 3.3 Components of CAN

CAN mainly includes three parts, i.e., the TAN transfers the word-level and sentence-level attentions across domains, sentiment classifier predicts the sentiment label and CA classifiers align the category from source to target domain. TAN includes the word attention transfer and sentence attention transfer. The document representations and labels are obtained by TAN and sentiment classifier respectively. On the basis of single domain classifier, CA classifiers add  $K$  category-wise domain classifiers, which combine the document representations and pseudo labels to avoid the mismatch. We describe the components of CAN successively.

**Transferable Attention Network:** To transfer the important words across domains, we use the hierarchical attention network that assign different weights to words [17]. Assuming that a document has  $L$  sentences and each sentence  $s_p$  contains  $Q$  words, where  $w_{pq}$  is the  $q$ -th word in the  $p$ -th sentence,  $q \in [1, Q]$ , We map words into dense vectors for representing sentences through an embedding matrix  $M$ ,  $x_{pq} = Mw_{pq}$ . The sentence vector  $s_p$  summarizes all words' representation by word attention transfer and the document vector  $v$  which summarizes the all sentences' information by sentence attention transfer.

**Category-Level Adversarial:** In domain adaptation problems, the data distribution is usually very complicated and it is difficult to achieve complete alignment. Incorrect alignment may be prone to under transfer or negative transfer. To enhance the positive transfer and combat negative transfer, we urgently need a technology which can align the data distribution for improving the transfer effect.

We propose the CA classifiers. On the basis of single domain classifier  $G_d$ , we add  $K$  category-wise domain classifiers  $G_d^k (k = 1, 2 \dots K)$ , where  $K$  is the number of category. Since target domain data is unlabeled, we use the output of sentiment classifier  $\hat{y} = G_c(v)$  as the probability of target domain. Similarly, it is also utilized on source domain. It is similar to attention mechanism which indicates the probability of the data from source or target domain. In other words, this method take category-level alignment into account during the adversarial process. The document vector  $v$  is modeled by the weight  $\hat{y}$  as the input of domain classifier  $G_d^k$ . We use the data  $X_s^l, X_s^u$  and  $X_t$  to train domain classifiers  $G_d$  and  $G_d^k$ , which predict domain labels. The goal of domain classifiers is to distinguish the two domains as accurately as possible. However, we want to learn the common features which can not be distinguished by domain classifiers. To address this problem, we introduce the Gradient Reversal Layer (GRL) [8] to reverse the gradient in the training process. The feedward and backpropagation process are as follows:

$$G(x) = x, \frac{\partial G(x)}{\partial x} = -\lambda I \quad (1)$$

The  $v$  and  $\widehat{vy}^k$  are the input of domain classifiers  $G_d$  and  $G_d^k$  respectively. Through the GRL as  $G(v) = \widetilde{v}_d$  and  $G(\widehat{vy}^k) = \widetilde{v}_d^k$ . Then we feed it to the corresponding domain classifiers.

$$\tilde{y}_d = \text{softmax}(fc(fc(\tilde{v}_d))) \quad (2)$$

$$\tilde{y}_d^k = \text{softmax}(fc(fc(\tilde{v}_d^k))) \quad (3)$$

where  $fc$  is the fully connected layer.

**Sentiment Classifiers:** The sentiment classifier utilizes the unlabeled data from source domain. According to the document representation  $v$ , we calculate the output of sentiment classifier as follows:

$$\tilde{y}_s = \text{softmax}(fc(fc(fc(v)))) \quad (4)$$

**Training Strategy:** The cross-domain sentiment classifier needs to use the domain-shared features to predict sentiment labels. In order to achieve this goal, CAN has two tasks i.e., domain classification and sentiment classification. We introduce cross-entropy loss functions for training CA classifiers and sentiment classifier respectively:

$$L_{dom_{con}} = -\frac{1}{N_s + N_t} \sum_{i=1}^{N_s+N_t} \tilde{y}_d \ln y_d + (1 - \tilde{y}_d) \ln(1 - y_d) \quad (5)$$

$$L_{dom_{class}} = -\frac{1}{N_s + N_t} \sum_{k=1}^K \sum_{i=1}^{N_s+N_t} \tilde{y}_d^k \ln y_d + (1 - \tilde{y}_d^k) \ln(1 - y_d) \quad (6)$$

$$L_{CA} = L_{dom_{con}} + L_{dom_{class}} \quad (7)$$

$$L_{sen} = -\frac{1}{N_s^l} \sum_{i=1}^{N_s^l} \tilde{y}_s \ln y_s + (1 - \tilde{y}_s) \ln(1 - y_s) \quad (8)$$

where  $y_d$  and  $y_s$  are the ground truth. Besides, we add the squared  $l_2$  regularization for sentiment classifier and CA classifiers. Finally, the objective function is as follows:

$$L = L_{CA} + L_{sen} + \rho L_{reg} \quad (9)$$

where  $L_{reg}$  is the regularization and prevents the overfitting,  $\rho$  is the regularization parameter. CAN model aims to minimize  $L$  except the GRL part which is maximized. Besides, we optimize the parameters by SGD.

## 4 Experiments

### 4.1 Dataset Preparation

In this section, we utilize two datasets to evaluate the ability of CAN method, one is the Amazon reviews dataset, the other is Airline reviews dataset. Table 1 summarizes the all dataset. We select the data from four domains: Books (B), DVD (D), Electronics (E) and Kitchen (K). Each domain contains 6000 labeled

reviews with 3000 positive samples (higher than 3 stars) and 3000 negative samples (lower than 3 stars). Additionally, the dataset also contains lots of unlabeled data. Here we randomly extract 8,000 unlabeled reviews as training data. These unlabeled data are only used to train the domain classifier which distinguishes the data comes from different domains. We choose 1000 reviews from the target domain as the testing data. We conduct the cross-domain experiments between every two domains and get 12 cross-domain sentiment classification tasks:  $B \rightarrow D$ ,  $B \rightarrow E$ ,  $B \rightarrow K$ ,  $D \rightarrow B$ ,  $D \rightarrow E$ ,  $D \rightarrow K$ ,  $E \rightarrow B$ ,  $E \rightarrow D$ ,  $E \rightarrow K$ ,  $K \rightarrow B$ ,  $K \rightarrow D$ ,  $K \rightarrow E$ . For example, the  $B \rightarrow D$  is the task which transfers from the source domain B to the target domain D.

Airline reviews dataset is scraped from Skytraxs Web portal which has one of the most popular review sites within the air travel industry. It is the labeled data that includes 41396 reviews for Airline(AL), 17721 Reviews for Airport(AP), 1258 reviews for Seat(S), 2264 reviews for Lounge(L). We select the 3000 positive reviews (recommended value is 1) and 3000 negative reviews (recommended value is 0) from the Airline and Airport to be consistent with the Amazon reviews dataset. Besides, we randomly extract 8000 labeled reviews to train the domain classifier. We construct 8 cross-domain sentiment classification tasks:  $B \rightarrow AL$ ,  $D \rightarrow AL$ ,  $K \rightarrow AL$ ,  $E \rightarrow AL$ ,  $B \rightarrow AP$ ,  $D \rightarrow AP$ ,  $K \rightarrow AP$ ,  $E \rightarrow AP$ .

**Table 1.** Statistics of Amazon and Airline datasets

Domain	B	D	K	E	AL	AP
#Train	5000	5000	5000	5000	5000	5000
#Test	1000	1000	1000	1000	1000	1000
#Unlabel	8000	8000	8000	8000	8000	8000

## 4.2 Implementation Details

We adopt the 300-dimensional word2vec vectors with the skip-gram model to initialize the embedding matrix  $M$  [19]. The maximum sentence length  $L$  and maximum word length  $Q$  are 20 and 28 respectively. All weight matrices are randomly initialized by a uniform distribution  $U[-0.01,0.01]$ . The dimensional of GRU hidden states is set to 70. The regularization weight  $\rho$  and dropout rate are set to 0.005 and 0.6 respectively. We utilize the stochastic gradient descent with momentum rate 0.9 to optimize model during the training process. Because the different training sizes for different classifiers we set batch size  $b_s = 50$  for the sentiment classifier and batch size  $b_d = 260$  for the domain classifier. The adaptation rate is  $\lambda = \frac{2}{1+exp(\frac{-10p}{N})-1}$ , where  $p = \frac{n}{N}$ . The  $n$  and  $N$  are current epoch and the maximum epoch respectively. The  $N$  is set to 100. The learning rate is  $\eta = max(0.003 * 0.1^{\lfloor \frac{n}{10} \rfloor}, 0.0005)$ .

### 4.3 Benchmark Methods

**Naive** (Hochreiter et al. 1997): it only uses source domain data based on LSTM.

**SCL** (Blitzer et al. 2006): it aims to identify the correlation between pivot and non-pivot by using multiple pivot prediction tasks.

**SFA** (Pan et al. 2010): it aims to solve mismatch of data distribution by aligning domain-specific words.

**DANN** (Ganin et al. 2015): it uses the domain adaptation with the 5000-dimension feature representations by adversarial training.

**AMN** (Li et al. 2017): it uses memory networks and adversarial training to get domain-shared representations.

**HATN & HATN<sup>n</sup>** (Li et al. 2018): it extracts pivots and non-pivots by the hierarchical attention network across domains. HATN does not contain the hierarchical positional encoding and HATN<sup>h</sup> does.

**ITAN & ITAN<sup>n</sup>** (Zhang et al. 2019): it uses interactive attention which combines aspects and sentences information. ITAN<sup>n</sup> does not contain the aspects information and ITAN does.

**CAN<sup>s</sup> & CAN<sup>c</sup> & CAN**: it is our methods that include the single domain classifier, category-wise classifiers and the CA classifiers respectively.

**Table 2.** Classification results on the Amazon dataset.

S T	Naive	SCL	SFA	DANN	AMN	HATN	HATN <sup>h</sup>	IATN	IATN <sup>n</sup>	CAN <sup>s</sup>	CAN <sup>c</sup>	CAN
B D	0.786	0.807	0.813	0.832	0.855	0.858	0.861	0.868	0.854	0.869	0.871	<b>0.874</b>
B E	0.752	0.763	0.776	0.764	0.824	0.853	0.857	0.865	0.849	0.861	0.859	<b>0.867</b>
B K	0.737	0.771	0.785	0.790	0.811	0.849	0.852	0.859	0.838	0.866	0.865	<b>0.871</b>
D B	0.756	0.782	0.788	0.805	0.846	0.858	0.863	0.870	0.848	0.871	0.871	<b>0.876</b>
D E	0.734	0.754	0.758	0.796	0.812	0.849	0.856	<b>0.869</b>	0.855	0.858	0.863	0.867
D K	0.767	0.779	0.786	0.814	0.827	0.853	0.862	0.858	0.839	0.858	0.861	<b>0.865</b>
E B	0.696	0.716	0.724	0.735	0.766	0.808	0.810	0.818	0.768	0.832	0.835	<b>0.841</b>
E D	0.722	0.745	0.754	0.786	0.827	0.838	0.840	0.841	0.825	0.844	0.845	<b>0.850</b>
E K	0.787	0.817	0.825	0.841	0.857	0.868	0.879	0.887	0.859	0.888	0.893	<b>0.894</b>
K B	0.686	0.713	0.724	0.752	0.805	0.824	0.833	0.847	0.828	0.841	0.843	<b>0.851</b>
K D	0.723	0.752	0.758	0.776	0.812	0.841	0.845	0.844	0.835	0.838	0.841	<b>0.848</b>
K E	0.807	0.818	0.825	0.843	0.867	0.868	0.870	0.876	0.864	0.878	0.881	<b>0.882</b>
Avg	0.746	0.768	0.776	0.794	0.825	0.847	0.851	0.859	0.837	0.858	0.861	<b>0.866</b>

The experimental results are shown in Table 2. Comparing with others methods, CAN model has achieved the best performances on most tasks. The Naive model performance is badly at every task because it only uses the source domain data. SFA model achieves 77.6% on average since the features of the review are manually extracted and the linear classifier is not sufficient to express the model. HATN<sup>h</sup> model achieves 85.1% on average because it automatically learns the domain-shared and domain-specific features. IATN model achieves 85.9%



on average because it combines the sentences and aspects information. However, HATN<sup>h</sup> and IATN model does not take the category-level alignment into account. Compared with CAN<sup>s</sup> model, CAN<sup>c</sup> achieves 86.1% on average, which is 0.3% higher than CAN<sup>s</sup> because the categories from different domains are aligned. Comparing with CAN<sup>c</sup> model, CAN model has better performance. The reason is that the pseudo-labels may lead to the incorrect alignment of data distributions. For the hard transfer task E  $\rightarrow$  B, E  $\rightarrow$  D, the performance of CAN<sup>c</sup> is very close to the CAN<sup>s</sup>. But for the simple transfer task E  $\rightarrow$  K, K  $\rightarrow$  E, the promotion is very obvious since the classification accuracy of target domain is higher. Finally, CAN model achieves 86.6% on average since we combine the single domain and category-wise classifiers.

The cross-domain sentiment classification tasks in Table 2 come from different domains with the same origin, such as E  $\rightarrow$  K and B  $\rightarrow$  D task. They are highly relevant. To show the performance of CAN on hard transfer tasks, we construct 8 new tasks which come from different origins. The experimental results are reported in Table 3. The Naive method without target domain samples performs the worst and the classification accuracy is 8.1% lower than CAN. The CAN and CAN<sup>c</sup> improves the classification accuracy by 0.9% and 0.3% than CAN<sup>s</sup>, respectively. The results show that our method is effective on difficult transfer tasks.

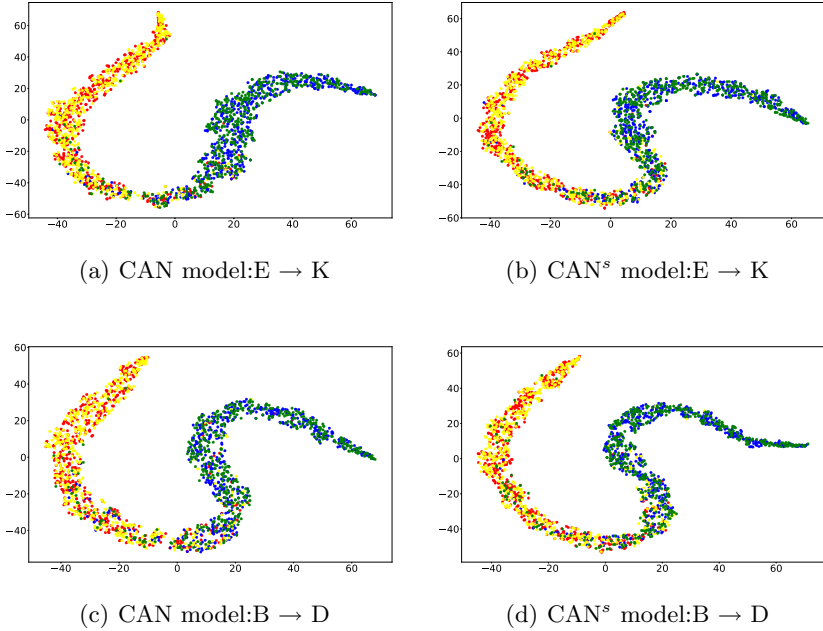
**Table 3.** Classification results on the Airline dataset.

S T	B AL	D AL	E AL	K AL	B AP	D AP	E AP	K AP	Avg
Naive	0.702	0.694	0.718	0.704	0.584	0.582	0.603	0.579	0.646
CAN <sup>s</sup>	0.768	0.787	0.792	0.794	0.651	0.659	0.644	0.649	0.718
CAN <sup>c</sup>	0.773	0.792	0.789	0.801	0.646	0.661	0.646	0.658	0.721
CAN	<b>0.776</b>	<b>0.795</b>	<b>0.794</b>	<b>0.811</b>	<b>0.658</b>	<b>0.669</b>	<b>0.652</b>	<b>0.659</b>	<b>0.727</b>

#### 4.4 Visualization of Features

To better illustrate the effectiveness of CAN, we visualize the feature of penultimate layer as shown in Fig. 3. We choose the task E  $\rightarrow$  K and B  $\rightarrow$  D. The visualization results show that CAN model has better distinguishable features. The features of CAN<sup>s</sup> model are not well discriminated clearly for lacking the category-wise alignment.

CAN model considers complex structures of the data distributions. The different domains are more indistinguishable and different categories are more discriminated under the CAN model. The experimental result is superior for cross-domain sentiment classification because we add the category-wise domain classifiers on training process.



**Fig. 3.** The t-SNE visualization of features extracted by CAN and CAN<sup>s</sup> model for E → K and B → D task. The red, blue, yellow and green points denote the source positive, source negative, target positive and target negative examples correspondingly (Color figure online)

<p><b>DVD-negative Example:</b>  lightweight. this film has been mentioned in the same breath as so called hood films such as clockers and menace society. it may be important, and it may have been hardcore gritty convincing</p> <p><b>Electronics-positive Example:</b>  lightweight and compact. very good speakers. except i use them with powerpoint with small groups. the volume is ok but i bought a second unit and found the computer would only run one speaker at a time dan poynter author publisher speaker author.</p> <p style="text-align: center;">D → E Task</p>
<p><b>DVD-negative Example:</b>  lightweight. this film has been mentioned in the same breath as so called hood films such as clockers and menace society. it may be important, and it may have been hardcore gritty convincing</p> <p><b>Kitchen-positive Example:</b>  comforter is a delight we have the heat in our house turned way down at night and we are toasty warm under our new comforter it is lightweight and delight</p> <p style="text-align: center;">D → K Task</p>

**Fig. 4.** Visualization of the CAN on D → E and D → K tasks

## 4.5 Visualization of Attention

We also visualize the word attention transfer in Fig. 4. We choose the  $D \rightarrow E$  and  $D \rightarrow K$  and highlight the words according to the attentions. Deeper red words mean that it has the heavier weight than others. Figure 4 shows that the words with sentiment polarity have higher attentions. Generally, the word “lightweight” intuitively indicates a negative sentiment in DVD domain. In contrast, the word “lightweight” usually indicates a positive sentiment in the electric or kitchen domain. Since there are some words (good, delight) which have opposite sentiment polarity in the document, CAN model still correctly predicts labels on target domain. The document representation takes these words into account by hierarchical attention mechanism.

## 5 Conclusion

In this paper, we propose the CAN model. CA classifiers further align the category-level data distribution by combining the label information and document representations on adversarial process. Besides, we transfer word-level and sentence-level attentions under TAN. The experimental results show that the CAN model effectively improves classification accuracy on Amazon reviews dataset and Airline reviews dataset.

**Acknowledgments.** This paper is Supported by National Key Research and Development Program of China under Grant No.2017YFB0803003 and National Science Foundation for Young Scientists of China (Grant No. 61702507).

## References

1. Wang, G., Sun, J., Ma, J., Xu, K., Gu, J.: Sentiment classification: the contribution of ensemble learning. *Decis. Support Syst.* **57**, 77–93 (2014)
2. Wang, S., Manning, C.D.: Baselines and bigrams: simple, good sentiment and topic classification. In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers*, vol. 2, pp. 90–94. Association for Computational Linguistics (2012)
3. Tang, D., Qin, B., Feng, X., Liu, T.: Target-dependent sentiment classification with long short term memory. *arXiv preprint arXiv:1512.01100* (2015)
4. Blitzer, J., McDonald, R., Pereira, F.: Domain adaptation with structural correspondence learning. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pp. 120–128. Association for Computational Linguistics (2006)
5. Pan, S.J., Ni, X., Sun, J.T., Yang, Q., Chen, Z.: Cross-domain sentiment classification via spectral feature alignment. In: *Proceedings of the 19th International Conference on World Wide Web*, pp. 751–760. ACM (2010)
6. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: a deep learning approach. In: *Proceedings of the 28th International Conference on Machine Learning (ICML 2011)*, pp. 513–520 (2011)

7. Chen, M., Xu, Z., Weinberger, K., Sha, F.: Marginalized denoising autoencoders for domain adaptation. arXiv preprint [arXiv:1206.4683](https://arxiv.org/abs/1206.4683) (2012)
8. Ganin, Y., Lempitsky, V.: Unsupervised domain adaptation by backpropagation. arXiv preprint [arXiv:1409.7495](https://arxiv.org/abs/1409.7495) (2014)
9. Li, Z., Wei, Y., Zhang, Y., Yang, Q.: Hierarchical attention transfer network for cross-domain sentiment classification. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
10. Yu, J., Jiang, J.: Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 236–246 (2016)
11. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint [arXiv:1408.5882](https://arxiv.org/abs/1408.5882) (2014)
12. Li, Z., Zhang, Y., Wei, Y., Wu, Y., Yang, Q.: End-to-end adversarial memory network for cross-domain sentiment classification. In: IJCAI, pp. 2237–2243 (2017)
13. Zhang, K., Zhang, H., Liu, Q., Zhao, H., Zhu, H., Chen, E.: Interactive attention transfer network for cross-domain sentiment classification. In: Thirty-Third AAAI Conference on Artificial Intelligence (2019)
14. Peng, M., Zhang, Q., Jiang, Y.G., Huang, X.J.: Cross-domain sentiment classification with target domain specific information. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 2505–2513 (2018)
15. Sharma, R., Bhattacharyya, P., Dandapat, S., Bhatt, H.S.: Identifying transferable information across domains for cross-domain sentiment classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 968–978 (2018)
16. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint [arXiv:1409.0473](https://arxiv.org/abs/1409.0473) (2014)
17. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., Hovy, E.: 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, pp. 1480–1489 (2016)
18. Huang, Z., et al.: Question difficulty prediction for reading problems in standard tests. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)