





Multi-domain Transfer Learning for Text Classification

Xuefeng Su^{1,2} , Ru Li^{1,3} , and Xiaoli Li^{1,4}

¹ School of Computer and Information Technology,
Shanxi University, Taiyuan, China
liru@sxu.edu.cn

² School of Electronic Business and Logistics,
Business College of Shanxi University, Taiyuan, China
suxuefeng@bcsxu.edu.cn

³ Key Laboratory of Ministry of Education for Computation Intelligence and Chinese
Information Processing, Shanxi University, Taiyuan, China

⁴ Institute for Infocomm Research, A-star, Singapore, Singapore
xlli@i2r.a-star.edu.sg

Abstract. Leveraging data from multiple related domains to enhance the model generalization performance is critical for transfer learning in text classification. However, most existing approaches try to separate the features into shared and private spaces regardless of correlations between domains, resulting in the inadequate features sharing among certain most related domains. In this paper, we propose a generic dual-channels multi-task learning framework for multi-domain text classification, which can capture global-shared, local-shared, and private features simultaneously. Our novel framework incorporates Adversarial network and Mixture of experts into a neural network for multi-domain text classification, which is very useful for sharing more features among domains. The extensive experiments on the real-world text classification data-sets across 16 domains demonstrate our proposed approach achieves better results than five state-of-the-art techniques.

Keywords: Multi-domain classification · Multi-task learning · Transfer learning · Adversarial network · Mixture of experts

1 Introduction

The current generation of neural network-based natural language processing models perform extremely well when large amounts of labelled data are available. However, they are prone to overfitting when faced with insufficient training data in a target domain for a classification task. It is thus an intuitive idea to

Supported by the National Natural Science Foundation of China (No. 61936012, No. 61772324) and Open Project Foundation of Intelligent Information Processing Key Laboratory of Shanxi Province (No. CICIP2018007).

leverage data from some related domains to enhance the models’ generalization performance for target domain.

A straightforward approach to utilizing data from multiple related domains is to combine them into a single domain. This strategy, however, does not account for distinct relations among examples from multiple domains. Transfer learning [1, 2], as an effective approach to transfer knowledge across domains, can be used to share knowledge for multiple domains and multiple languages [3, 4]. Moreover, multi-task learning, as a branch of transfer learning, has become a widely used approach for multi-domain text classification [5–8].

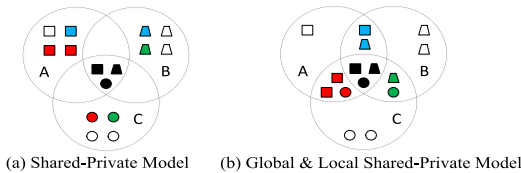


Fig. 1. Two sharing schemes for domain A, B and C. The overlap between three domains is global shared space and the overlap between any two domains is local shared space. The black and color solid icons denote the global shared features shared across all domains and local shared features shared only between certain domains respectively. Red, green and blue solid icons represent local shared features between domain A&C, B&C, and A&B. The hollow icons represent private features. (Color figure online)

Nevertheless, most existing work on multi-task learning attempts to divide the features of different domains into two spaces [5–8], namely, private and shared spaces (See Fig. 1(a)). In particular, one is used to store domain-specific features, while the other one is used to capture domain-invariant features. However, there are two limitations in this framework. First, there is no explicit modeling of the local correlations between the domains, and each domain is treated equally. For example, given three domains: book, video and movie, video domain can share more information with movie domain than with book domain besides common features across three domains, because video and movie domain are more similar. As shown in Fig. 1(a), suppose domain A is more similar to domain C, domain A can share more features with domain C than domain B. We can regard domain A is more important than domain B for domain C instead of equally important. In shared-private framework, every domain is treated equally without considering the correlations between domains. Second, separating feature space into shared and private spaces causes inadequate use of inter-domain information. As shown in Fig. 1(a), the color solid icons represent the local shared features. In shared-private framework, these features are treated as private features, resulting in the inadequate use of these local shared features.

To address these problems, in this paper we divide the feature space into global shared, local shared and private space (See Fig. 1(b)), and propose an generic framework for multi-domain text classification, in which the global

shared, local shared and private features are modeled explicitly with a *dual-channels neural network*. Specifically, one channel adopts a structure similar to the adversarial network which is widely used to model the common and domain-invariant *global shared features* in computer vision and natural language processing [9, 10, 18]. The other channel is based on mixture of experts structure, which explicitly models the domain relationships and allows parameters to be automatically allocated to capture either *local shared features* or *private features* [10–12]. Finally, the features from two channels are effectively combined into a feature vector as an integrated representation. The contribution of this paper is threefold:

- We extend the multi-task learning to mitigate data insufficient problem in given domain by utilizing data from other related domains.
- We propose a novel generic framework for multi-domain text classification which explicitly models global-shared, local-shared and private features.
- Our extensive experimental results on real benchmark data demonstrate the efficiency and effectiveness of our proposed method.

2 Related Work

In recent years, multi-source domain adaptation has attracted the attention of many researchers in NLP. Kim *et al.* use attention based on the base models' representation to compute interpolation weights [13]. Sebastian *et al.* propose a method to weight source domain models with the similarity between source domain and the corresponding target domain [14]. Himanshu *et al.* utilize unlabeled data of the target domain to find a distribution weighted combination of the source domains [15]. Recent adversarial methods on multi-source domain adaptation align source domains to the target domains globally [16, 17]. Jiang *et al.* express the target model as a mixture of source domain experts [10].

Multi-source domain adaptation is similar to multi-domain classification in some ways, but the research goal and applied scenario are different. We can not directly use the methods of multi-source domain adaptation to multi-domain classification tasks, although the theories on knowledge sharing are common.

With the development of deep learning, the neural-based model for multi-task learning has been widely applied as a common technique in NLP. Liu *et al.* first utilize different LSTM layers to construct multi-task learning framework for text classification [6], and they subsequently propose a generic multi-task framework [7]. Liu *et al.* propose a shared-private multi-task model, which uses multiple LSTM to encode sentences from different domains [8]. Liu *et al.* adopt self-attention to learn domain-specific descriptor vectors and Bi-LSTM to learn general sentence-level vectors [5].

Different from these models, our model represents the text from multiple domain in a more refined way that the features are divided into global-shared, local-shared and private features.

3 Preliminary

Multi-domain Text Classification. Suppose there are m domains $\{D_k\}_{k=1}^m$, and $\{D_k\}$ contains $|D_k|$ data points (s_j^k, d_j^k, y_j^k) , where $j \in \{1, 2, \dots, |D_k|\}$, s_j^k is a sequence of words $\{w_1, w_1, \dots, w_{|s_j^k|}\}$, d^k is a domain indicator (since we use 1 to m to indicate each domain, $d_j^k = k$) and y_j^k is class label (e.g. $y_j^k \in \{-1, +1\}$ for binary sentiment classification). The task is to learn a function F which maps each input (s_j^k, d_j^k) to its corresponding class label y_j^k .

Text Representation. This paper uses self-attention mechanism [23] to weight the output of encoding network (such as LSTM) to form a text representation. Suppose $H = \{h_1, h_2, \dots, h_n\}$ is the output of encoding network whose input is word embedding sequence $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$. The encoding function can be implemented with RNN or one of its variants, which will be discussed in Sect. 4. Then text representation function can be expressed as follows:

$$h = \text{Rep}(H) = \alpha^T H \quad (1)$$

where $\alpha = \text{softmax}(\tanh(H^T)v)$ is attention vector over H , v is a parameter vector; h is the vector representation of input sequence \mathbf{x} .

4 The Proposed Method

In this paper, we propose a generic dual-channel multi-task learning framework for multi-domain text classification based on global and local shared representation (GLR-MTL), which consists of four parts: embedding layer, global-shared representation network, local-shared representation network and text classification layer. Two representation networks, called dual-channel network, encode the input from embedding layer into global-shared representation and local-shared representation respectively in a parallel manner. Note that private representation is regarded as a special case of local-shared representation, so we don't mention it above for simplicity. Then, the outputs of two channels are concatenated into an integrated representation, and the text classification layer maps it into a label distribution. The structure of GLR-MTL is illustrated in Fig. 2.

4.1 Global-Shared Representation

Global-shared representation should be domain-invariant, that is, the common feature representation of all domains. Many researchers integrate the adversarial network into the deep neural network to learn the domain-invariant representation. Inspired by these studies, this paper designs a novel global-shared representation network as a module for GLR-MTL, which consists of G-Encoder layer, gradient reverse layer (GRL) and domain classifier layer, as shown in Fig. 2.

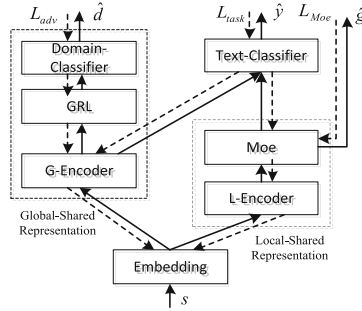


Fig. 2. Overall framework for GLR-MTL. The solid arrows represent the direction of data flow in forward propagation, and the dotted arrows represent the flow direction of gradient in back-propagation.

G-Encoder. G-Encoder is used to model the input sequence and output the global-shared representation with the help of adversarial training. Theoretically G-Encoder can adopt any kind of recurrent neural network. Here we adopt recurrent neural network with long short-term memory (LSTM) or bidirectional LSTM (BiLSTM) due to their superior performance in various NLP tasks. Given input \mathbf{x} , G-Encoder can be expressed as follows:

$$H_g = G - Encoder(\mathbf{x}, \theta_g) \tag{2}$$

where \mathbf{x} is the input, and θ_g is the parameters.

Domain Discriminator. Domain Discriminator, as a part of adversarial network, is used to predict domain label distribution of the input text. The neural network architecture of domain discriminator consists of one fully connected layer and one softmax layer. It can be defined as follows:

$$\hat{d} = softmax(W_D Rep(H_g) + b_D) \tag{3}$$

where \hat{d} is the distribution of domain label, $Rep(H_g)$ is the text representation of H_g , W_D and b_D are the parameter matrix and bias respectively. For the simplicity of illustration, we use a function to represent the domain discriminator as follows:

$$\hat{d} = Dc(H_g, \theta_g) \tag{4}$$

where H_g denotes the input of the function, and θ_g represents all parameters.

Incorporating Adversarial Training. Inspired by Adversarial networks [8, 19], we design a new network similar to adversarial network for global-shared representation, in which G-Encoder is working adversarially towards domain discriminator, preventing it from making an accurate prediction about the labels of domains. We assume that a shareable feature is one for which the domain

discriminator cannot learn to identify the origin domain of the input observation based on domain adaptation theory [9]. Therefore, during the training phase, when G-Encoder and domain discriminator reach a point at which both cannot improve and the domain discriminator is unable to differentiate among all the domains, the output of G-Encoder is the global-shared representation for all domains. To reach this goal, the adversarial training loss is incorporated into our learning goal, and it is expressed as follows:

$$L_{adv} = \min_{\theta_G} (\max_{\theta_D} \sum_{k=1}^m \sum_{j=1}^{|D_k|} d_j^k) \log[Dc(H_{g_j}^k, \theta_D)] \quad (5)$$

where θ_D and θ_G are the parameters of the domain discriminator and G-Encoder respectively.

Gradient Reverse Layer (GRL). When the gradient descent method is used to solve the above min-max loss, the parameters usually need to be solved by alternating training. Yaroslav [20] proposed a gradient inversion method to transform the problem into a single minimum objective problem without alternating training. Thus, we insert a GRL between G-Encoder and the domain discriminator to simplify the training process.

4.2 Local-Shared Representation.

Global-shared presentation focuses on capturing the common information of all domains, and pay attention equally to every domain. However, it is obvious that the similarity or relatedness is different between any two domains, which means two similar domains can share more features than less similar ones. For example, given three domains: book, video and music, music domain can share more information with video domain than with book-domain, besides common features across three domains. As shown in Fig. 1-(b), the color icons denote shareable features between any two domains. We call the features shared between any two domains local-shared features that are not considered in global-shared representation module. Next, we will discuss local-shared representation module of GLR-MTL consisting of L-Encoder layer and Mixture of experts (Moe).

L-Encoder. L-Encoder is first used to encode the input sequence into intermediate representation and subsequently feed it into Moe. Here we adopt network architecture as same as G-Encoder. Given input \mathbf{x} , L-Encoder can be expressed as follows:

$$H_L = L - Encoder(\mathbf{x}, \theta_L) \quad (6)$$

where \mathbf{x} is the input, and θ_L is the parameters.

Mixture of Experts (Moe). Inspired by previous studies on Moe [10–12], GLR-MTL integrates Moe into the local shared representation module. As shown in Fig. 3, our proposed Moe consists of one gate network and multiple expert networks. The gate network is used to generate the probability distribution of the domain to which the input sequence belongs to. On the other hand, each expert network acts as a domain-specific encoder, and multiple experts encode the input at each time step in a parallel manner. Finally, the weighted sum, the results of the outputs of these experts multiplied by the outputs of the gate networks as the weights, is used to represent the output of Moe at current time step. In other words, given sequence $H_L = \{h_1, h_2, \dots, h_n\}$ from L-Encoder, at time step t , Moe can be precisely expressed as follows:

$$\hat{g} = \text{softmax}(W_g \text{Rep}(H_L) + b_g) \quad (7)$$

$$E_k(h_t) = \text{ReLU}(W_k h_t + b_k) \quad (8)$$

$$h_t^L = \sum_{k=1}^m \hat{g}[k] E_k(h_t) \quad (9)$$

where \hat{g} is the predicted probability distribution of domains and $\sum_{k=1}^m \hat{g}[k] = 1$, $W_g \in$ and b_g are parameter matrix and bias of the gate network; $E_k(\cdot)$ is the function of k -th domain-specific encoder, W_k and b_k are parameter matrix and bias of $E_k(\cdot)$; h_t^L is the output of Moe at time step t , and $H_L = \{h_1^L, h_2^L, \dots, h_n^L\}$ is the local-shared representation of the input.

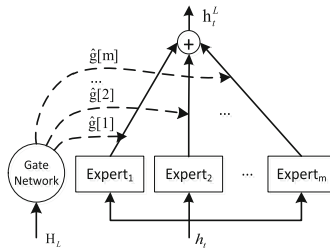


Fig. 3. Structure of Mixture of Experts

At time step t , h_t^L integrates the information of multiple domain experts into one representation, and the amount of fused information from each expert depends on the probability distribution of gate output. For an input sample belongs to domain k , we assume $k = \text{arg max}_{k \in \{1, 2, \dots, m\}}(\hat{g})$. Then $\hat{g}[i]$ should be close to $\hat{g}[k]$ if domain i is similar to domain k . The sample can share more information from domain i by Function (9). In an extreme case, $\hat{g}[k]$ is almost equal to 1 if there is no similar domain to domain k , which means the presentation $H_L = \{h_1^L, h_2^L, \dots, h_n^L\}$ of the sample is totally private because Moe can only use

the information from k -th expert. Therefore, it is essential to learn a precise \hat{g} , and the learning goal can be expressed as follows:

$$L_{Moe} = \min\left(-\sum_{k=1}^m \sum_{j=1}^{|D_k|} d_j^k \log(\hat{g}_j^k)\right) \quad (10)$$

4.3 Multi-domain Text Classification

Given an input text, we first concatenate its global-shared representation H_g and local-shared representation H_l into a completed representation. Then we pass the representation into text classification module, which has a fully connected layer followed by a softmax non-linear layer that predicts the probability distribution over classes. In particular, the classification module can be expressed as follows:

$$\hat{y} = \text{softmax}(W_C \text{Rep}([H_g; H_l]) + b_C) \quad (11)$$

where \hat{y} is prediction probabilities of text classes, and W_C and b_C are parameter matrix and bias respectively. For multi-domain text classification, the learning goal can be expressed as follows:

$$L_{task} = \min\left(-\sum_{k=1}^m \sum_{j=1}^{|D_k|} y_j^k \log(\hat{y}_j^k)\right) \quad (12)$$

4.4 GLR-MTL Objective

GLR-MTL incorporates adversarial network and Moe into classification model to learn global-shared and local-shared representation. Therefore, we need to jointly learn these supervised objectives, resulting in the following learning objective:

$$L = L_{task} + \delta L_{Moe} + \eta L_{adv} \quad (13)$$

where δ and η are hyper-parameters.

5 Experiment

5.1 Datasets and Experimental Settings

To make an extensive evaluation, we use FuDan [8] datasets consisting of 16 different domains from several popular review corpora. The data is labelled with either positive or negative. All the datasets in each domain are partitioned randomly into training set, development set and testing set with the proportion of 70%, 20% and 10% respectively. The average data size of training set, development set and testing test are 1386, 200 and 400 respectively. The average length of sentences across domains ranges from 21 to 269.

For fair comparison, our GLR-MTL model and competing models use the same pre-training word embedding (Glove 200-dimension embedding [24]) and

encoder module. LSTM and BiLSTM, as two different encoder settings, are adopted to act as L-Encoder and G-Encoder.

The hidden state units of the encoder is set to 100. The dropout rate and mini-batch size are set to 0.5 and 8 respectively. We employ Adam optimizer with the learning rate of 0.002. We take the hyper-parameters which achieve the best performance on the development set via an small grid search over combinations of $\delta \in [0.01, 0.1]$ and $\eta \in [0.01, 0.1]$. Finally, we choose δ as 0.05 and η as 0.01.

5.2 Baselines

Multi-domain text classification can be solved in a single task manner or multiple task manner. We choose two single task models and three recently proposed multi-task models related to multi-domain text classification as baselines.

- **SD-ST**: Single-domain single-task model, which means we separately train the model for each domain. These models can not share features among domains. SD-ST uses vanilla LSTM or BiLSTM to encode the input text, and uses the last hidden outputs as the text representation [22].
- **CNN-ST**: This model uses vanilla LSTM or BiLSTM as encoder, and use CNN to represent the features. Different from SD-ST, multiple domains are combined into a single domain before the model training [21].
- **SP-MTL**: This model is a multi-task model which uses one shared LSTM and multiple private-LSTMs to represent the shared and private features respectively. Then, these kinds of features are concatenated into a vector [8].
- **ASP-MTL**: This model is a variant of SP-MTL, which incorporates adversarial network into the model [8].
- **DSAM**: This model adopts self-attention to learn a domain-specific descriptor vector and uses BiLSTM to learn the general sentence-level vector. Then, the general and domain-specific are concatenated into one vector [5].

5.3 Results and Analysis

As shown in Table 1, we can see that the overall performance of our models achieve the highest accuracy on 16 domains, no matter whether LSTM or BiLSTM encoder is used. More concretely, compared with single-task models, the performance of GLR-MTL has achieved much better results, indicating that multi-task models which utilize multi-domain data simultaneously are helpful to improve the model performance on each domain. It is noteworthy that, compared to multi-task models, CNN-ST is a strong single task model in which all domains are combined into one domain and it achieves a comparable results with certain multi-task models (such as SP-MTL and ASP-MTL). These results confirm our observation that separating feature space into shared-private spaces causes inadequate use of inter-domain information.

Compared with multi-task models, GLR-MTL achieves 1.5% average improvement, which indicates the importance of explicitly modeling the local

Table 1. Accuracy (averages across five random seeds) of our models on 16 domains against existing baselines. The title in each parentheses below the model name represents the encoder type (LSTM or BiLSTM) used by the model.

Models	Single-Task				Multi-Task			Ours	
	SD-ST (LSTM)	SD-ST (BiLSTM)	CNN-ST (LSTM)	CNN-ST (BiLSTM)	SP-MTL (LSTM)	ASP-MTL (LSTM)	DSAM (BiLSTM)	GLR-MTL (LSTM)	GLR-MTL (BiLSTM)
Apparel	83.2	86.0	84.5	87.7	86.5	87.0	85.3	88.5	88.2
Baby	84.7	83.5	88.0	89.0	86.7	88.2	89.6	90.3	92.3
Books	79.5	81.0	84.8	86.0	81.2	84.0	84.4	89.5	88.3
Camera	85.2	86.0	86.8	89.5	88.0	89.2	89.3	90.5	89.5
Electronics	81.7	80.5	86.0	87.0	84.0	85.5	86.3	87.8	90.3
DVD	80.5	78.5	85.0	86.5	84.7	86.8	87.0	86.0	87.3
Health	84.5	78.7	87.8	90.3	87.2	88.2	89.3	90.0	90.5
IMDB	81.7	85.0	83.8	85.3	84.7	85.5	86.8	83.0	87.5
Kitchen	78.0	81.2	87.0	86.8	85.2	86.2	90.0	90.3	89.8
Magazines	89.2	91.5	92.3	93.3	92.0	92.2	93.0	91.5	92.3
MR	72.7	74.7	68.4	69.2	76.0	76.7	76.8	72.7	72.7
Music	76.7	77.2	82.8	82.3	83.0	82.5	82.3	86.8	87.5
Software	84.7	85.7	89.3	88.8	87.0	87.2	86.8	89.5	91.8
Sports	81.7	84.0	89.0	86.5	87.2	85.7	88.5	89.1	87.8
Toys	83.2	84.7	87.3	87.5	85.2	88.0	89.3	91.0	89.8
Video	81.5	83.7	87.3	88.0	83.2	84.5	87.0	87.3	90.8
Avg. acc	81.8	82.6	85.6	86.5	85.1	86.1	87.0	87.7	88.5

correlations between the domains and capturing global-shared, local-shared and private features simultaneously. Note for GLR-MTL, the performances on certain domains are degraded, since this model puts all features into a unified space and optimizes the overall goal for all domains as a whole.

5.4 Ablation Analysis

To analyze the influence of Adversarial network and Moe module on the model performance, we design the ablation experiments based on GLR-MTL(BiLSTM). From Table 2, we can see that both Adversarial network and Moe are helpful for GLR-MTL to improve its performance, since the model performance is degraded without each of them. We also observe that both global-shared and local-shared features are important to improve the performance of the model, as the model performance is degraded with only using one kind of features.

5.5 Transferability Analysis

GLR-MTL can transfer knowledge from related domains to the target domain through feature sharing, which can enhance the generalization performance on target domain with limited training data. To test the transferability of GLR-MTL, we take turns using different percentage (See Fig. 4) of training data of target domain combined with all data of source domains to train the model.

As shown in Fig. 4, we can see that the accuracy of each domain rises with the amount of training data increasing, and almost reach its upper-bound with

Table 2. Average accuracy of GLR-MTL with different settings on 16 domains. GLR-MTL(w/o Moe) represents the model in which $\delta = 0$ and the parameters of Moe module is fixed, and GLR-MTL(w/o adv) represents the model in which $\eta = 0$ and the parameters of Domain-classifier module is fixed. GR-MTL and LR-MTL represent the model use only global-shared and local-shared representation respectively.

Models	Avg. acc
GLR-MTL	88.5
GLR-MTL(w/o Adv)	87.8
GLR-MTL(w/o Moe)	87.9
GR-MTL	87.7
LR-MTL	87.3

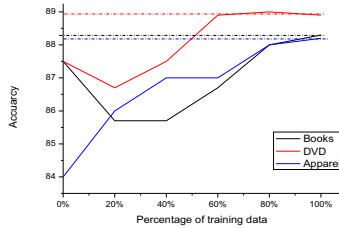


Fig. 4. Transferability on three randomly choosed domains: Books, DVD and Apparel. Solid lines denote the accuracy of each domain with different percentage of training data. Dotted lines represent the accuracy upper bound which is the accuracy of the model trained with all data of 16 domains.

using only 60%–80% of training data, which means the GLR-MTL framework can use less data to achieve nearly same accuracy through the features sharing. Moreover, the accuracy of target domain Books and DVD both achieves 87.5% without using target domain data that is higher than the single-domain model as shown in Table 1, while Apparel achieves 84% nearly the same as the single-domain model. This indicates Books and DVD domains can share more features from other domains than Apparel domain does, because the relatedness of Apparel with other domains is relative weak while DVD domain is much similar to Video and IMDB domains. It is noteworthy that the performance on DVD and Books domains declines with very limited training data, such as 20% and 40% of training data, which indicates very limited training data may causes negative transfer and certain amount of training data is necessary.

6 Summary

This paper proposed a framework GLR-MTL for multi-domain text classification, which can model global-shared features and local-shared features respectively with the help of adversarial network and Moe. Experiments on datasets

of 16 domains show that the overall performance of GLR-MTL is significantly better than five baseline models. Moreover, this framework can transfer features from multi-domains to one target domain, which makes the model achieve comparative performance on target domain with very limited training data.

References

1. Sebastian, R.: Neural Transfer Learning for Natural Language. National University of Ireland, Ireland, Galway (2019)
2. Jindong W., Yiqiang C., Han Y.: Easy transfer learning by exploiting intra-domain structures. arXiv preprint [arXiv:1904.01376v2](https://arxiv.org/abs/1904.01376v2) (2019)
3. Yuhsiang L., Chianyu C., Jean L., et al.: Choosing transfer languages for cross-lingual learning. In: 57th Proceedings of ACL, pp. 3125–3135. ACL, Florence (2019)
4. Xilun C., Ahmed H. A., Hany H.: Multi-source cross-lingual model transfer: learning what to share. In: 57th Proceedings of ACL, pp. 3098–3112. ACL, Florence (2019)
5. Liu, Q., Zhang, Y., Liu, J.: Learning domain representation for multi-domain sentiment classification, In: Proceedings of the 2018 Conference on NAACL-HLT, pp. 541–550. ACL, New Orleans (2018)
6. Pengfei L., Xipeng Q., Xuanjing H.: Recurrent neural network for text classification with multi-task learning. In: 25th Proceedings of IJCAI, pp. 2873–2879. AAAI, California (2016)
7. Pengfei L., Xipeng Q., Xuanjing H.: Deep multi-task learning with shared memory. In: Proceedings of the 2016 Conference on EMNLP, pp. 118–127. ACL, Austin (2016)
8. Pengfei L., Xipeng Q., Xuanjing H.: Adversarial multi-task learning for text classification. In: 55th Proceedings of ACL, pp. 1–10. ACL, Vancouver (2017)
9. Han Z., Shanghang Z., Guanhang W., et al.: Adversarial multiple source domain adaptation. In: 32nd Conference on NIPS, pp. 1–12. Montréal (2018)
10. Jiang G., Darsh S., Regina B.: Multi-source domain adaptation with mixture of experts. In: Proceedings of the 2018 Conference on EMNLP, pp. 4694–4703. ACL, Brussels (2018)
11. Noam S., Azalia M., Krzysztof M., et al.: outrageously large neural networks: the sparsely-gated mixture-of-experts layer. In: 5th Proceedings of ICLR, pp. 1–19. Toulon (2017)
12. Jiaqi M., Zhe Z., Xinyang Y., et al.: Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In: 24th Proceedings of The International Conference on KDD, pp. 1930–1938. SIGKDD, London (2018)
13. Kim Y., Stratos K.: Domain attention with an ensemble of experts. In: 55th Proceedings of ACL, pp. 643–653. ACL, Vancouver (2017)
14. Sebastian, R., Parsa, G., John, G.: Knowledge adaptation: teaching to adapt. arXiv preprint [arXiv:1702.02052](https://arxiv.org/abs/1702.02052) (2017)
15. Himanshu S., Manjira S., Shourya R.: Cross-domain text classification with multiple domains and disparate label sets. In: 54th Proceedings of ACL, pp. 641–1650. ACL, Berlin (2016)
16. Himanshu S., Arun R., Shourya R.: Multi-source iterative adaptation for cross-domain classification. 25th Proceedings of IJCAI, California, pp. 3691–3697 (2016)
17. Han, Z., Shanghang, Z., Guanhang, W., et al.: Multiple source domain Adaptation with adversarial learning. In: 6th Proceedings of ICLR, Vancouver, pp. 1–10 (2018)

18. Xilun, C., Claire, C.: Multinomial adversarial networks for multi-domain text classification. In: Proceedings of the 2018 Conference on NAACL-HLT, pp. 1226–1240. ACL, New Orleans (2018)
19. Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, pp. 2672–2680 (2014)
20. Yaroslav, G., Evgeniya, U.: Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**, 1–35 (2016)
21. Zhou, C., Sun, C., Liu, Z., et al.: A C-LSTM neural network for text classification. *Comput. Sci.* **1**(4), 39–44 (2015)
22. Rafal J., Wojciech Z., Ilya, S.: An empirical exploration of recurrent network architectures. In: 32nd Proceedings of ICML, pp. 2342–2350. ACM, Lille (2015)
23. Zhouhan, L., Minwei, F., Cicero, N., et al.: A structured self-attentive sentence embedding. In: 5th Proceedings of ICLR, Toulon, pp. 1–15 (2017)
24. GloVe, 17 April 2020. <https://nlp.stanford.edu/projects/glove/>