

Domain Adaptation for Person-Job Fit with Transferable Deep Global Match Network

Shuqing Bian^{1,2}, Wayne Xin Zhao^{*1,2}, Yang Song³, Tao Zhang³ and Ji-Rong Wen^{1,2}

¹School of Information, Renmin University of China

²Beijing Key Laboratory of Big Data Management and Analysis Methods

³BOSS Zhipin

{bianshuqing, jrwen}@ruc.edu.cn batmanfly@gmail.com

{songyang, kyleen.zhang}@kanzhun.com

Abstract

Person-job fit has been an important task which aims to automatically match job positions with suitable candidates. Previous methods mainly focus on solving the match task in single-domain setting, which may not work well when labeled data is limited. We study the domain adaptation problem for person-job fit. We first propose a deep global match network for capturing the global semantic interactions between two sentences from a job posting and a candidate resume respectively. Furthermore, we extend the match network and implement domain adaptation in three levels, *i.e.*, sentence-level representation, sentence-level match, and global match. Extensive experiment results on a large real-world dataset consisting of six domains have demonstrated the effectiveness of the proposed model, especially when there is not sufficient labeled data.

1 Introduction

Recent years have witnessed the rapid growth of online recruitment platforms. According to (Iqbal, 2019), there were 590 million users and 11 million job listings in LinkedIn¹ from about over 200 countries and territories all over the world. With the increasing amount of online recruitment data, it has become an essential task that is able to automatically match jobs with suitable candidates, called *person-job fit* (Qin et al., 2018; Shen et al., 2018).

Due to the importance of person-job fit, many efforts have been devoted to improving the match algorithms (Xu et al., 2018; Malinowski et al., 2006; Paparrizos et al., 2011; Lee and Brusilovsky, 2007; Zhang et al., 2014). Among these studies, a typical approach is to casting the

task as a supervised text match problem. Given a set of labeled data (*i.e.*, person-job match records), it aims to predict the match label based on the text content of job postings and candidate resumes. More recently, deep learning has shed light on person-job fit methods by learning more effective text representations or text match models (Qin et al., 2018; Zhu et al., 2018; Jiang et al., 2019).

Although these methods have achieved significant progress in person-job fit, they rely on a considerable amount of labeled data to learn the match models. When labeled data is insufficient, existing methods may not work well as expected. Therefore, the deficiency in labeled data has raised a significant challenge to the person-job fit algorithms. The availability of labeled data for different job categories is highly varying. A job position from a major category usually attracts hundreds of job applicants, while one from a minor category receives very few applications. Besides, due to the emergence of new positions, data accumulation under these new job categories is limited. For this purpose, we investigate domain adaptation for person-job fit, focusing on the text match between job postings and candidate resumes. We aim to utilize the acquired knowledge and information from a source domain with sufficient labeled data to improve the prediction performance in a target domain with limited or little labeled data.

Text-based domain adaptation has been extensively studied in the literature, such as text classification (Li et al., 2018; Glorot et al., 2011; Ziser and Reichart, 2017). However, these works mainly focus on how to model an individual document instead of the match for a document pair, which are not directly applicable to our task. As for existing person-job fit methods (Qin et al., 2018), they mainly learn an overall representation for both a job posting and a candidate resume. Then,

^{*}Corresponding author

¹<https://www.linkedin.com>

the match score is measured via the similarity between the two overall representations, which is limited to extend for domain adaptation. A major reason is that job postings and resumes are written on top of basic semantic units (*e.g.*, sentences) related to skills, abilities or experiences, we have to model the global interaction between fine-grained semantic units for an accurate match. Such global match information is important to consider for effective domain adaptation.

To address these difficulties, we first design a deep global match network for modeling person-job fit in a single domain. This model can capture the comprehensive interaction information based on all pairwise sentence interactions given a job posting and a resume. Furthermore, we extend the match network for domain adaptation in three levels. First, to overcome the semantic gap or language variation across different domains, the sentence representations have been enhanced via structural correspondence learning. The derived sentence representations are more transferable across domains. Second, the sentence-level match function has been set to share parameters across domains for improving the match accuracy in target domains. We construct a match matrix to contain all the match results in sentence level. Third, we design a convolution based component to learn transferable match patterns and components across domains. Although different domains have varying semantic representations or match functions, the global match information reflected in the match matrix should share similar patterns.

To our knowledge, it is the first time that domain adaptation has been studied for the person-job fit task. To evaluate our proposed model, we construct a large real-world dataset containing two large and four small domains from an online recruitment platform. Extensive experiment results have demonstrated the effectiveness of the proposed model, especially when the labeled data is limited.

2 Related Work

As an important task in recruitment data mining (Li et al., 2017a; Xu et al., 2018; Oentaryo et al., 2018), person-job fit has been extensively studied in the literature. Early methods include treating person-job fit as a job/candidate recommendation problem (Lu et al., 2013; Diaby et al., 2013) and extending users' historical behavior by

job application records for better match (Hong et al., 2013). Recently, deep learning methods have been utilized to design a hierarchical representation structure based on historical recruitment records (Qin et al., 2018; Zhu et al., 2018; Ramanath et al., 2018).

Domain adaptation is a hot topic in natural language processing, which has received much attention over the decades. Among the early works, Structural Correspondence Learning (SCL) (Blitzer et al., 2007, 2006) is a classic algorithm which learns transferable feature representations across domains. With the revival of deep learning, several studies focus on extending the neural network models for domain adaptation (Yu and Jiang, 2016; Ziser and Reichart, 2017). To further eliminate the distribution differences between source and target domains, the adversarial mechanism has been incorporated into domain adaptation (Li et al., 2017b, 2018; Ganin et al., 2016).

Besides, our work is also related to semantic text match (Mueller and Thyagarajan, 2016; Yin et al., 2016; Wang et al., 2017), specially the work which casts text match as image recognition (Pang et al., 2016) or captures both global and local interactions (Dai et al., 2018; Rao et al., 2018; Yang et al., 2018).

Our work is highly based on these related studies. To our knowledge, domain adaptation for person-job fit has seldom been studied before. We focus on how to transfer the match information between job postings and resumes across domains.

3 Problem Formulation

We assume text content of a job posting and a candidate resume are available as input. A job posting p is described as n_p sentences $\{\mathbf{w}_i^{(p)}\}_{i=1}^{n_p}$, and the i -th sentence $\mathbf{w}_i^{(p)}$ is a sequence of $n_{p,i}$ word tokens denoted by $\{w_{i,k}^{(p)}\}_{k=1}^{n_{p,i}}$, where $w_{i,k}^{(p)}$ denotes the k -th word of the i -th sentence in p . Each job posting sentence typically describes the skill or ability that is required for the position. Similarly, a resume r consists of n_r sentences $\{\mathbf{w}_i^{(r)}\}_{i=1}^{n_r}$, where $\mathbf{w}_i^{(r)}$ is a sequence of $n_{r,i}$ word tokens denoted by $\{w_{i,k}^{(r)}\}_{k=1}^{n_{r,i}}$. Each resume sentence typically describes the skills or experiences that a candidate has acquired. Let a binary label y denote the match result for a person-job pair based on the text of job posting p and

Symbols	Descriptions
p, r	the job posting and resume
n_p, n_r	the number of sentences in the p and r
$w_i^{(p)}, w_i^{(r)}$	the i -th sentence in the p and r
$\{w_{i,k}^{(p)}\}, \{w_{i,k}^{(r)}\}$	the k -th word of the i -th sentence in p and r
y	the match result for a person-job pair based on the p and r
$\mathcal{D}^t, \mathcal{D}^s$	the training set for the target domain t and source domain s
n_t, n_s	the number of target labeled instances and source labeled instances
$\mathbf{h}_{i,k}^{(p)}, \mathbf{h}_{i,k}^{(r)}$	the hidden states learned from BiGRU network for the k -th word of the i -th sentence in the p and r
L_P, L_R	the dimension size of hidden states for the p and r
\mathbf{M}	the match matrix ($\in \mathbb{R}^{n_p \times n_r}$) for the sentence similarity between the p and r
\mathbf{W}	the weight matrix ($\in \mathbb{R}^{K \times K}$), K is the dimension size
$\mathbf{m}^{p,r}$	the match representation derived by stacked convolutional layers with max-pooling layers
Θ	all the involved parameters
$\mathbf{h}^p, \mathbf{h}^r$	the representations of the p and r
\hat{y}	the output score calculated by the Multi-Layer Perceptron
$\mathbf{x}_{i,\text{SCL}}^p, \mathbf{x}_{i,\text{SCL}}^r$	the transferable representation transformed by the original feature in the p and r
$\widetilde{\mathbf{h}}_i^p, \widetilde{\mathbf{h}}_i^r$	the enhanced job posting and resume sentence representation
\mathbf{A}	the shared weight matrix by all the domains
\mathbf{B}^d	the specific weight matrix to the domain d
$\hat{\Theta}^s$	the learned parameters for the auxiliary domain s
$\hat{\Theta}^t$	the domain-specific parameters for the target domain t
$\widetilde{\mathbf{m}}^{p,r}$	the new representation by the concatenation of the output from both components with the parameters $\hat{\Theta}^s$ and $\hat{\Theta}^t$

Table 1. The notations used in this paper.

candidate resume r .

Following (Qin et al., 2018; Zhu et al., 2018), we cast the person-job fit problem into a text match task based on the text information of job postings and candidate resumes. The *Person-Job Fit* task aims to learn a match function that is able to predict the actual label for a new job-person pair from the target domain.

Furthermore, we consider a domain adaptation setting for such a match task. Given a target domain t , a training set $\mathcal{D}^t = \{\langle p_j^t, r_j^t, y_j \rangle\}_{j=1}^{n_t}$ of n_t labeled instances is given. Considering the target domain to be new or minor, there is very little labeled data. We assume an auxiliary training set $\mathcal{D}^s = \{\langle p_j^s, r_j^s, y_j \rangle\}_{j=1}^{n_s}$ from another source domain s is given, consisting of n_s labeled instances. The source domain (or auxiliary domain) is selected from an existing domain with sufficient match records. In this case, we have $n_t \ll n_s$. For simplicity, we will drop the domain index unless needed, which is the target domain by default. The notations used in this paper are summarized in Table 1.

4 The Proposed Model

In this section, we first present a deep global match network for person-job fit using only target domain data. Then, we extend the match network by leveraging the labeled data of an auxiliary domain.

4.1 A Deep Global Match Network for Single-Domain Person-Job Fit

Previous studies (Shen et al., 2018; Zhu et al., 2018) mainly focus on modeling the overall interaction between the representations of a job and a candidate. They cannot explicitly capture global match information in terms of fine-grained semantic units. In this section, we propose a deep global match network which is able to characterize global sentence-level interaction for semantic match between a job posting and a candidate resume.

Hierarchical Attention-based RNN Encoder.

First, we employ the bi-directional recurrent neural networks with gated recurrent unit (BiGRU) to model both sentences and documents (*i.e.*, a job posting or a resume).

Formally, let $\overleftarrow{\mathbf{h}}_{i,k}^{(p)}$ and $\overrightarrow{\mathbf{h}}_{i,k}^{(p)}$ denote the forward and backward states learned from the BiGRU network for the k -th word of the i -th sentence in job posting p . We concatenate the two directional representations as the representation of a word, *i.e.*, $\mathbf{h}_{i,k}^{(p)} = [\overleftarrow{\mathbf{h}}_{i,k}^{(p)}, \overrightarrow{\mathbf{h}}_{i,k}^{(p)}]$. Since some words in a sentence are important than others, we apply the attention mechanism (Qin et al., 2018) to derive the sentence representation $\mathbf{h}_i^{(p)} \in \mathbb{R}^{L_P}$ in a job posting:

$$\mathbf{h}_i^{(p)} = \sum_{k=1}^{n_{p,i}} \alpha_k \cdot \mathbf{h}_{i,k}^{(p)}, \quad (1)$$

where the attention weight α_k is calculated as

$$\begin{aligned} \alpha_k &= \text{softmax}(a_k), \\ a_k &= \mathbf{v}_1^\top \cdot \tanh(\mathbf{W}_1 \mathbf{h}_{i,k}^{(p)} + \mathbf{b}_1). \end{aligned} \quad (2)$$

Similarly, we can derive the sentence representations for a resume, denoted by $\mathbf{h}_i^{(r)} \in \mathbb{R}^{L_R}$. Based on the learned sentence representations, we apply the similar attentional BiGRU network to derive the overall representations for a job posting and a

resume, denoted by \mathbf{h}^p and \mathbf{h}^r :

$$\mathbf{h}^{(p)} = \sum_{i=1}^{n_p} \beta_i^{(p)} \cdot \mathbf{h}_i^{(p)}, \quad (3)$$

$$\mathbf{h}^{(r)} = \sum_{i=1}^{n_r} \beta_i^{(r)} \cdot \mathbf{h}_i^{(r)}, \quad (4)$$

where $\beta_i^{(p)}$ and $\beta_i^{(r)}$ are the attention weights defined in a similar way as Eq. 2.

Global Match Representation. After encoding the sentences, we further model global semantic interactions between sentence representations of a job and a resume. In specific, we compute the similarities between each sentence in a job posting and each sentence in a candidate resume. Assume that a job posting p has n_p sentences and a resume r has n_r sentences. We can derive a match matrix $\mathbf{M} \in \mathbb{R}^{n_p \times n_r}$ for modeling the global semantic interactions. Formally, to calculate the sentence similarity in \mathbf{M} , we apply a linear form as

$$M_{i,i'} = \mathbf{h}_i^{(p)\top} \cdot \mathbf{W} \cdot \mathbf{h}_{i'}^{(r)}, \quad (5)$$

where $\mathbf{W} \in \mathbb{R}^{K \times K}$ is the match matrix, $\mathbf{h}_i^{(p)}$ is the representation of the i -th sentence for a job posting p , and $\mathbf{h}_{i'}^{(r)}$ is the representation of the i' -th sentence for resume r . Inspired by the recent work on image-based text match models (Pang et al., 2016), we propose to use the convolution based method to model the match information. Formally, we stack two 2-D convolutional layers with two 2-D max-pooling layers in an interleaving way, and derive a match representation $\mathbf{m}^{p,r}$ as

$$\mathbf{m}^{p,r} = \text{ConvNet}(\mathbf{M}; \Theta), \quad (6)$$

where $\mathbf{m}^{p,r}$ summarizes the match information from global sentence interactions, and “ Θ ” denote all the involved parameters.

Predicting the Match Label. With the learned match representation $\mathbf{m}^{p,r}$, we concatenate it with the representations of the job posting and the resume as the input for the predictor,

$$\hat{y} = \text{MLP}([\mathbf{m}^{p,r}; \mathbf{h}^p; \mathbf{h}^r]), \quad (7)$$

where we \mathbf{h}^p and \mathbf{h}^r are the representations of job posting p and resume r defined in Eq. 3 and Eq. 4 respectively, and $\text{MLP}(\cdot)$ is the Multi-Layer Perceptron with a nonlinear layer and a sigmoid layer. Compare with previous work (Qin

et al., 2018; Zhu et al., 2018), a major novelty of this match model lies in the fact it has explicitly modeled the match information between a job posting and a resume. It is able to model sentence-level interactions between two text documents.

4.2 Domain Adaptation by Transferring Local and Global Match Information

In this part, we study how to extend the proposed deep global match network for domain adaptation. Instead of directly enriching the overall representation of a training instance, we seek a way to transfer local and global match information.

Enhanced Sentence Representation with Structural Correspondence. Different domains may have significant language variation or semantic gap, which makes it difficult to derive transferable sentence representations across domains. Inspired by the classic SCL algorithm (Blitzer et al., 2007), we propose to model the structural correspondence of sentences in different domains by using pivot keywords. Consider an example of two snippets from two different job domains:

S_1 : *grasp* C programming skills

S_2 : *grasp* picture editing skills

Although the semantics of computer programming and picture editing are very different, they are aligned skill requirements for the two domains via the pivot word *grasp*. By pre-selecting a number of high-quality pivot words, SCL algorithm is able to learn such semantic alignment via large-scale co-occurrence data. In specific, the SCL algorithm is able to learn a mapping function $f_{\text{SCL}}(\cdot)$ that transforms an original into a more transferable representation:

$$\mathbf{x}_{i,\text{SCL}}^p = f_{\text{SCL}}(\mathbf{h}_i^p). \quad (8)$$

For the i -th sentence of a job posting, we derive its representation by concatenating the original representation in Eq. 1 and the transformed representation using SCL algorithm:

$$\widetilde{\mathbf{h}}_i^p = \mathbf{h}_i^p \oplus \mathbf{x}_{i,\text{SCL}}^p, \quad (9)$$

where “ \oplus ” is the vector concatenation operation. Similarly, we can obtain the enhanced resume sentence representation, denoted by $\widetilde{\mathbf{h}}_i^r$. With the $\widetilde{\mathbf{h}}_i^p$ and $\widetilde{\mathbf{h}}_i^r$, we can update the overall representations of \mathbf{h}^p and \mathbf{h}^r in Eq. 3 and 4.

Transferring Sentence-level Match Information. After obtaining the new sentence representations, we study how to transfer the parameters

for modeling sentence-pair match. As shown in Eq. 5, we incorporate a transformation matrix \mathbf{W} to compute sentence similarities. A simple transfer strategy is to share the whole matrix in both target and source domains. However, such a method will make it less flexible to capture domain-specific match information, since the local match information is likely to be varied across domains. Hence, we propose to factorize the matrix \mathbf{W} into a product of two smaller matrices $\mathbf{A} \in \mathbb{R}^{l \times K}$ and $\mathbf{B}^d \in \mathbb{R}^{l \times K}$:

$$\mathbf{W}^d = \mathbf{A}^\top \cdot \mathbf{B}^d, \quad (10)$$

where \mathbf{A} is shared by all the domains, while \mathbf{B}^d is specific to domain d . In this factorization, we share the parameters across domains and meanwhile set domain-specific parameters to better capture domain-specific information.

Transferring Global Match Information. As shown in Eq. 5, we have constructed a match matrix consisting of pairwise sentence similarities between a job posting and a resume. Our idea is that although sentence-level representation or match is different across domains, global semantic interactions are likely to show the similar patterns. For example, the majority of required skills or abilities in the job posting should be well covered for a good candidate, corresponding to large entries in the match matrix \mathbf{M} . Based on this idea, we propose to transfer the parameters of the convolution module across domains. In specific, we first use the rich training data of the auxiliary domain s to train the parameters in Eq. 6, and obtain the learned parameters $\hat{\Theta}^s$. We assume the mapping process from the match matrix to the final match representation (*i.e.*, $\mathbf{M} \rightarrow \mathbf{m}^{p,r}$) is transferable. Hence, we directly reuse $\hat{\Theta}^s$ trained from the source domain. For enhancing the modeling capacity, we also train a domain-specific convolution component with the parameters $\hat{\Theta}^t$. For the target domain, we derive the new match representation by using the concatenation of the output from both components with the parameters $\hat{\Theta}^s$ and $\hat{\Theta}^t$:

$$\widetilde{\mathbf{m}}^{p,r} = \text{ConvNet}(\mathbf{M}; \hat{\Theta}^s) \oplus \text{ConvNet}(\mathbf{M}; \hat{\Theta}^t). \quad (11)$$

4.3 The Final Model and Training

To predict the match label for domain adaptation, we replace \mathbf{h}^p , \mathbf{h}^r and $\mathbf{m}^{p,r}$ in Eq. 7 with $\widetilde{\mathbf{h}}^p$, $\widetilde{\mathbf{h}}^r$,

Modules	Settings
HA-RNN	$L_R = L_P = 200$, #GRU-layer=2, batch-size=512, init.-learning-rate=0.001
SCL	#pivots = 200, init.-learning-rate=0.01
$\mathbf{W}, \mathbf{A}, \mathbf{B}^d$	$K = 100, l = 50$
Global Matching	kernel sizes = $\{4 \times 4, 6 \times 6\}$, init.-learning-rate=0.001, batch-size=512

Table 2. Parameter settings of our model.

Domain	#instances	#ASPJ	#ASPR
Technology	60,000	14.0	12.3
Sales	75,000	12.0	8.2
Design	3,000	14.2	10.5
Product	3,000	15.6	9.1
Car	3,000	12.7	7.9
Journey	3,000	9.5	7.0

Table 3. Statistics of the datasets. #ASPJ means the average number of sentences per job posting, and #ASPR means the average number of sentences per resume.

and $\widetilde{\mathbf{m}}^{p,r}$ respectively. We present an overview sketch of the proposed model in Fig. 1. We have highlighted the three points for transferring information across domain. Different from previous text-based domain adaption methods, we focus on the transfer of match information, including both local and global semantic interactions.

To optimize our model, we adopt the the binary cross-entropy loss over the entire training data as the total loss. To learn the model parameters, we adopt the Adam optimizer (Kingma and Ba, 2014). In order to avoid overfitting, we adopt the dropout strategy with a rate of 0.1. More parameter settings can be found in Table 2.

5 Experiments

In this section, we first set up the experiments, and then report the results and analysis.

5.1 Experimental Setup

We evaluate our model on a large real-world dataset provided by the largest online recruiting platform named ‘‘BOSS Zhipin’’ (the BOSS Recruiting)² in China. To protect the privacy of candidates, all the records have been anonymized by deleting identity information. Since the original amount of recruitment data is huge, we randomly sample a fraction of the entire data, containing job postings and resumes within a time period of six months. Specifically, the dataset has covered six job domains: two major domains have a large number person-job pairs and four

²<https://www.zhipin.com>

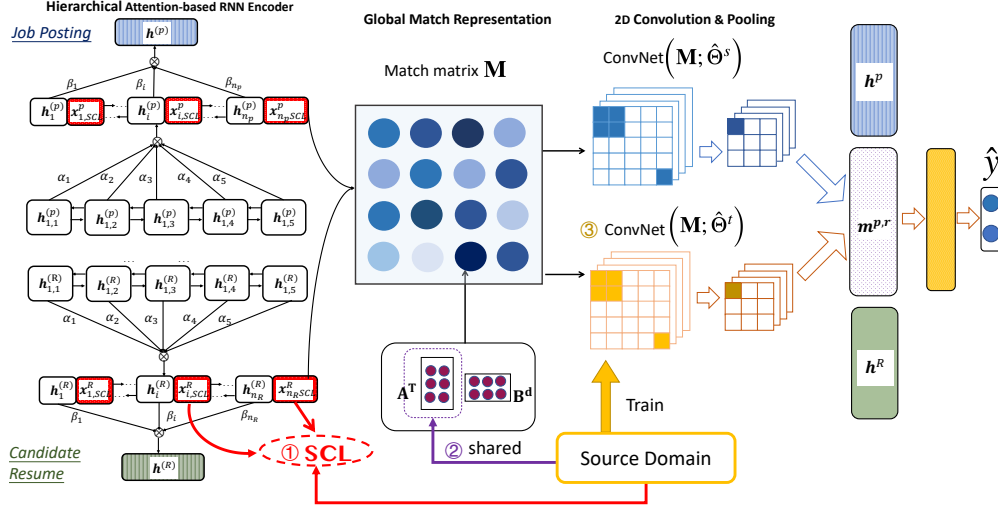


Figure 1. The overview of the transferable deep global match network.

Methods	Technology		Sales		Design		Cars		Product		Journey	
	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
DSSM	0.690	0.625	0.686	0.619	0.638	0.577	0.621	0.558	0.625	0.566	0.616	0.557
PJFNN	0.685	0.623	0.678	0.617	0.633	0.566	0.618	0.555	0.619	0.561	0.608	0.546
BPJFNN	0.681	0.619	0.669	0.611	0.627	0.565	0.611	0.554	0.610	0.552	0.602	0.545
APJFNN	0.694	0.630	0.688	0.626	0.641	0.580	0.627	0.563	0.630	0.571	0.619	0.559
Our model	0.713	0.642	0.709	0.638	0.659	0.600	0.644	0.582	0.648	0.591	0.637	0.575

Table 4. Performance comparisons on single domain evaluation for person-job fit.

minor domains have a limited number of person-job pairs. The match label information is obtained according to the acceptance status of the candidates, provided by the online recruiting platform. In our dataset, the ratio between matched and unmatched instances is 1:1³. We perform the basic preprocessing steps on the text data, including tokenization and stopword removal. The statistics of the dataset are summarized in Table 3. Our code and data are available at <https://github.com/RUCAIBox/Person-Job-Fit>.

We consider two evaluation settings, namely single domain evaluation and domain adaptation evaluation. For all domains, we first split the entire data set with a ratio of 1:1 into a training set and a test set. Single domain evaluation examines the performance of a model trained with domain-specific training data respectively for each of all six domains, while domain adaptation evaluation examines the performance of a model trained with the data from both the source and target domains. For domain adaptation, we consider the four minor domains as target

³There were more negative instances in original data. we randomly sample an equal number of negative instances to reduce the data bias.

domains, and the major domains as source domains.

Since our task is casted as a classification task, we adopt four commonly used evaluation metrics, including Accuracy, Precision, Recall and F1.

5.2 Comparison Methods

Single Domain. For single domain evaluation, we consider three latest methods as baselines:

- DSSM (Huang et al., 2013): it utilizes a deep neural network (DNN) to map high-dimensional sparse features into low-dimensional dense features, and calculates the semantic similarity of the text pair.
- BPJFNN (Qin et al., 2018): It applies BiLSTMs to obtain the semantic representation of each word in job postings and resumes, and considers the text content as a single sequence.
- PJFNN (Zhu et al., 2018): It proposes a bipartite Convolutional Neural Networks that can effectively learn the joint representation of Person-Job fitness from historical job applications.
- APJFNN (Qin et al., 2018): It learns a word-level semantic representation for both job requirements and resumes based on RNN and four

hierarchical ability-aware attention strategies.

Domain Adaptation. For domain adaptation evaluation, we consider three baseline methods:

- SCL-MI (Blitzer et al., 2007): It proposes the SCL algorithm where pivot features are selected based on mutual information in the unlabeled data of both the source and target domains.
- AE-SCL-SR (Ziser and Reichart, 2017): It proposes to find a shared low dimensional representation in order to overcome the domain adaptation problem.
- ASP-MTL (Liu et al., 2017): It proposes an adversarial multi-task learning framework, alleviating the shared and private latent feature spaces from interfering with each other.

For our deep match network, we also prepare three simple single-domain variants, include (1) *Tgt-Only* trains the model with only target domain data, (2) *Src-Only* trains the model with only source domain data, and (3) *Mixed* trains the model with a simple mixture of source and target domain data.

5.3 Results and Analysis

In this part, we construct a series of experiments on the effectiveness of the proposed model for the person-job fit task.

Single Domain Results. We first report the results on single domain evaluation, where we only use the training data from the target domain. In Table 4, we compare our model with three recently proposed methods for person-job fit. Among the three baselines, the APJFNN method performs better than the other two methods. It adopts fine-grained attention mechanism based on skills and abilities. Our (single domain) model outperforms all the baselines. The proposed deep global match network extends the APJFNN by explicitly modeling the global match information for the sentence representations of job postings and resumes. These results indicate the global match information is useful to improve the performance of the person-job fit task.

Domain Adaptation Results. We select the domains of *technology* and *sales* as the source domains, since both contain much more training data, while the rest domains are selected as the target domains. Table 5 presents the performance of different comparison methods for domain adaptation. First, for the three variants for our single-

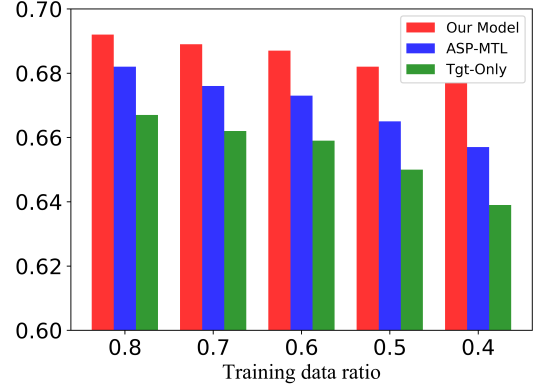


Figure 2. Domain adaptation with different train/test splitting ratios in the target domain for “Technology→Car”.

domain model, the performance order is *Src-Only* < *Mixed* < *Tgt-Only*. The result indicates that directly incorporating the source domain data as training data may hurt the performance for the target domain. Second, ASP-MTL performs best among the three baselines. It sets up both shared and private feature representations, and apply the adversarial mechanism to alleviate the shared and private latent feature spaces from interfering with each other. Finally, our model outperforms all the baselines with a large margin. Previous methods (e.g., ASP-MTL) learns an overall representation for a document, which cannot characterize global semantic interaction in fine-grained semantic units. As a comparison, our deep global match network is able to transfer information in three levels, namely sentence representation, sentence-level match and global match. Different from existing text-based domain adaption methods, our model is tailored to transfer the match information between a document pair.

Ablation Analysis. Recall we proposed three transfer techniques to improve the performance for domain adaptation in Section 4.2, including sentence representation (denoted by *SR*), sentence-level match (denoted by *SM*) and global match (denoted by *GM*). Now we examine the effect of each factor on the prediction performance. At each time, we remove an individual factor while keep the rest. Due to space limit, we only report the results with the cases “Technology→Cars” and “Sales→Product”. As shown in Table 6, all the three factors are useful to improve the performance of our model. Especially, the transferred information in sentence representation

source domain	methods	DESIGN		CARS		PRODUCT		JOURNEY	
		ACC	F1	ACC	F1	ACC	F1	ACC	F1
TECHNOLOGY	Src-Only	0.566	0.503	0.542	0.484	0.550	0.491	0.528	0.469
	Tgt-Only	0.659	0.600	0.644	0.582	0.648	0.591	0.637	0.575
	Mixed	0.653	0.599	0.636	0.580	0.641	0.582	0.629	0.574
	SCL-MI	0.657	0.598	0.645	0.586	0.649	0.591	0.640	0.577
	AE-SCL	0.672	0.610	0.659	0.602	0.662	0.603	0.650	0.594
	ASP-MTL	0.675	0.616	0.661	0.600	0.666	0.608	0.654	0.595
	Our Model	0.696	0.638	0.680	0.625	0.687	0.627	0.669	0.610
SALES	Src-Only	0.547	0.485	0.535	0.474	0.562	0.603	0.539	0.484
	Tgt-Only	0.659	0.600	0.644	0.582	0.648	0.591	0.637	0.575
	Mixed	0.646	0.587	0.631	0.568	0.643	0.586	0.632	0.574
	SCL-MI	0.655	0.596	0.642	0.582	0.652	0.594	0.645	0.583
	AE-SCL	0.665	0.606	0.654	0.597	0.666	0.608	0.657	0.598
	ASP-MTL	0.671	0.612	0.658	0.599	0.672	0.613	0.660	0.601
	Our Model	0.689	0.630	0.675	0.618	0.692	0.634	0.676	0.618

Table 5. Performance comparisons on domain adaptation evaluation for person-job fit.

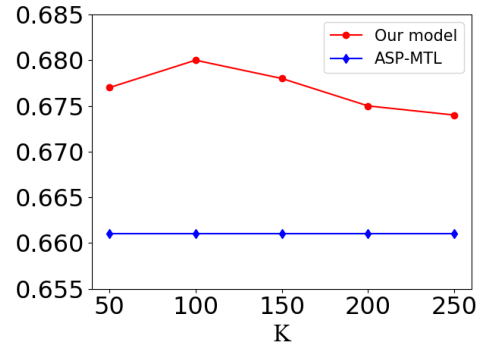
Variants	T→C		S→P	
	ACC	F1	ACC	F1
Our model	0.680	0.625	0.692	0.634
w/o SR	0.659	0.601	0.671	0.614
w/o SM	0.676	0.623	0.685	0.631
w/o GM	0.654	0.598	0.667	0.612

Table 6. Ablation study on the three factors of our model.

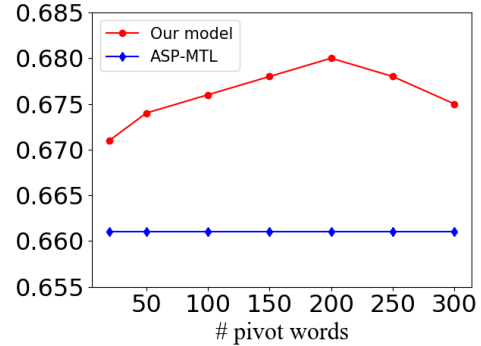
and global match bring more improvement to our model.

Varying the Amount of Training Data. An important consideration for domain adaptation is the amount of training data in the target domain. Here, we vary the ratio of training data in the target domain with the case “Technology→Car”. We re-split the dataset of the *Car* domain according to different training data ratios. We select the single domain variant of our model and the best baseline ASP-MTL as comparisons. Fig. 2 presents the varying results. With the decreasing of training data, the performance of all the methods drops accordingly. While, domain adaption techniques are useful to alleviate the deficiency of training data, since ASP-MTL and our model yield less performance drop. Especially, the performance of our model is relatively stable, which indicates its robustness for domain adaption.

Parameters Sensitivity Analysis. In our model, we have two important parameters, namely the dimension number K of the matrix \mathbf{W} in Eq. 10 and the number of pivot words in the SCL algorithm. We vary the two parameters and present the tuning results in Fig. 3. We select the best baseline ASP-MTL as a comparison. As we can see, the performance of our model is relatively stable *w.r.t.* the varying of the two parameters. A



(a) Varying the dimension number K .



(b) Varying the number of pivot words.

Figure 3. Parameter tuning for our model with the case “Technology→Car”.

dimension number of 100 and a number of 200 pivot words lead to the best performance.

5.4 Qualitative Analysis

A key point of our model is that it can transfer the global match information derived by computing the pairwise similarities between two sentence respectively from a job posting and a resume. Recall that we reuse the convolution-based component

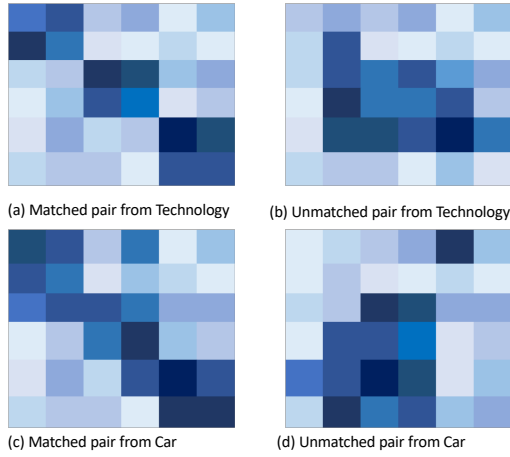


Figure 4. An illustrative example for the effect of the global match in our model. Here, “matched” and “unmatched” indicates the candidate person has been or not been accepted by the job position, respectively.

trained with the source domain data. Now, we study how such a mechanism work and why it is useful to improve the performance.

In Fig. 4, we present four person-job pairs from our dataset containing two matched pairs and two unmatched pairs. In a subfigure, each row corresponds to a sentence in a job posting, while a column corresponds to a sentence in a resume. Due to space limit, we only select six sentences for each document. We follow the method in Eq. 5 to construct the match matrix M . We use the darkness degree to indicate the similarity between two sentences. For ease of understanding, we further perform the row and column permutation to make the large entries distributed in blocks.

From Fig. 4, we can make the following observations. First, global match matrices have a significant difference between a matched and an unmatched case. Although the match matrix for an unmatched pair also contains many large values, they are usually located in off-diagonal entries. Comparing Fig. 4(a) and Fig. 4(c), it is interesting to see that they share a similar distribution pattern, *i.e.*, large values are mainly distributed along the diagonal line and these values are also aggregated in small blocks. Such a phenomenon indicates that the majority of the required skills have been well covered by a candidate in a matched pair. Without considering global semantic interactions, it is difficult to correctly predict the label for the unmatched pairs in Fig. 4(b) and Fig. 4(d), since they are likely to have a large similarity measured by the overall representations.

6 Conclusion

This paper studied the domain adaptation for person-job fit. We first proposed a deep global match network for the single-domain match setting. Then, we extended the proposed model for domain adaptation in three aspects. We constructed extensive experiments on a large real-world recruitment dataset, containing six job domains. The results have demonstrated the effectiveness of our model in terms of the prediction accuracy for person-job fit, especially when the training data is limited. Our current characterization with global match information provides a good form to develop interpretable domain adaptation models, *i.e.*, what kind of information has been transferred across domains. As future work, we will work along this line and investigate the design interpretable solutions to domain adaptation for person-job fit. We will also consider how to model the domain relationship for effectively transferring information across domains.

Acknowledgments

The authors would like to thank the anonymous reviewers for their helpful and constructive comments. This work was partially supported by the National Natural Science Foundation of China under Grant No. 61872369 and 61832017, the Fundamental Research Funds for the Central Universities, the Research Funds of Renmin University of China under Grant No. 18XNLG22 and 19XNQ047.

References

- John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447.
- John Blitzer, Ryan McDonald, and Fernando Pereira. 2006. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 conference on empirical methods in natural language processing*, pages 120–128. Association for Computational Linguistics.
- Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. 2018. Convolutional neural networks for soft-matching n-grams in ad-hoc search. In *Proceedings of the eleventh ACM international*

- conference on web search and data mining, pages 126–134. ACM.
- Mamadou Diaby, Emmanuel Viennet, and Tristan Launay. 2013. Toward the next generation of recruitment tools: an online social network-based job recommender system. In *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013)*, pages 821–828. IEEE.
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, Francois Laviolette, Mario Marchand, and Victor S Lempitsky. 2016. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):189–209.
- 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.
- Wenxing Hong, Siting Zheng, and Huan Wang. 2013. Dynamic user profile-based job recommender system. In *2013 8th International Conference on Computer Science & Education*, pages 1499–1503. IEEE.
- Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Conference on information and knowledge management*, pages 2333–2338. ACM.
- Mansoor Iqbal. 2019. LinkedIn usage and revenue statistics 2018. <http://www.businessofapps.com/data/linkedin-statistics/>. Accessed February 27, 2019.
- Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Mike Bendersky, Nadav Golbandi, and Marc Najork. 2019. Semantic text matching for long-form documents. In *International World Wide Web Conference, WWW*.
- Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Danielle H Lee and Peter Brusilovsky. 2007. Fighting information overflow with personalized comprehensive information access: A proactive job recommender. In *In Proceedings of the 3rd International Conference on Autonomic and Autonomous Systems*, pages 21–21. IEEE.
- Huayu Li, Yong Ge, Hengshu Zhu, Hui Xiong, and Hongke Zhao. 2017a. Prospecting the career development of talents: A survival analysis perspective. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 917–925. ACM.
- Zheng Li, Ying Wei, Yu Zhang, and Qiang Yang. 2018. Hierarchical attention transfer network for cross-domain sentiment classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Zheng Li, Yun Zhang, Ying Wei, Yuxiang Wu, and Qiang Yang. 2017b. End-to-end adversarial memory network for cross-domain sentiment classification. In *IJCAI*, pages 2237–2243.
- Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10.
- Yao Lu, Sandy El Helou, and Denis Gillet. 2013. A recommender system for job seeking and recruiting website. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 963–966. ACM.
- Jochen Malinowski, Tobias Keim, Oliver Wendt, and Tim Weitzel. 2006. Matching people and jobs: A bilateral recommendation approach. In *In Proceedings of the 39th Annual Hawaii International Conference on System Sciences, Vol. 6. IEEE*.
- Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016)*, pages 2786–2792. AAAI Press.
- Richard J Oentaryo, Ee-Peng Lim, Xavier Jayaraj Siddarth Ashok, Philips Kokoh Prasetyo, Koon Han Ong, and Zi Quan Lau. 2018. Talent flow analytics in online professional network. *Data Science and Engineering*, 3(3):199–220.
- Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Thirtieth AAAI Conference on Artificial Intelligence*.
- Ioannis Paparrizos, B Barla Cambazoglu, and Aristides Gionis. 2011. Machine learned job recommendation. In *In Proceedings of the 5th ACM International Conference on Recommender Systems*, pages 325–328. ACM.
- Chuan Qin, Hengshu Zhu, Tong Xu, Chen Zhu, Liang Jiang, Enhong Chen, and Hui Xiong. 2018. Enhancing person-job fit for talent recruitment: An ability-aware neural network approach. In *In Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-2018)*, Ann Arbor, Michigan, USA.
- Rohan Ramanath, Hakan Inan, Gungor Polatkan, Bo Hu, Qi Guo, Cagri Ozcaglar, Xianren Wu, Krishnaram Kenthapadi, and Sahin Cem Geyik. 2018. Towards deep and representation learning for talent search at linkedin. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 2253–2261. ACM.

- Jinfeng Rao, Wei Yang, Yuhao Zhang, Ferhan Ture, and Jimmy Lin. 2018. Multi-perspective relevance matching with hierarchical convnets for social media search. In *The 32nd AAAI Conference on Artificial Intelligence (AAAI-2018)*, New Orleans, LA, USA, 2018, pages 7–11.
- Dazhong Shen, Hengshu Zhu, Chen Zhu, Tong Xu, Chao Ma, and Hui Xiong. 2018. A joint learning approach to intelligent job interview assessment. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI-2018)*, Stockholm, Sweden, 2018.
- Chenglong Wang, Feijun Jiang, and Hongxia Yang. 2017. A hybrid framework for text modeling with convolutional rnn. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 2061–2069. ACM.
- Tong Xu, Hengshu Zhu, Chen Zhu, Pan Li, and Hui Xiong. 2018. Measuring the popularity of job skills in recruitment market: A multi-criteria approach. In *The 32nd AAAI Conference on Artificial Intelligence (AAAI-2018)*, New Orleans, LA, USA, 2018.
- Zhilin Yang, Jake Zhao, Bhuwan Dhingra, Kaiming He, William W. Cohen, Ruslan R. Salakhutdinov, and Yann LeCun. 2018. Glomo: Unsupervisedly learned relational graphs as transferable representations. In *Proceedings of the 32nd Annual Conference on Neural Information Processing Systems (NIPS)*, pages 8950–8961.
- Wenpeng Yin, Hinrich Schutze, Bing Xiang, and Bowen Zhou. 2016. Abcnn: Attention-based convolutional neural network for modeling sentence pairs. *Transactions of the Association for Computational Linguistics*, 4(1):259–272.
- 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.
- Yingya Zhang, Cheng Yang, and Zhixiang Niu. 2014. A research of job recommendation system based on collaborative filtering. In *In Proceedings of the 7th International Symposium on Computational Intelligence and Design*, pages 533–538. IEEE.
- Chen Zhu, Hengshu Zhu, Hui Xiong, Chao Ma, Fang Xie, Pengliang Ding, and Pan Li. 2018. Person-job fit: Adapting the right talent for the right job with joint representation learning. *ACM Transactions on Management Information Systems ACM TMIS*.
- Yftah Ziser and Roi Reichart. 2017. Neural structural correspondence learning for domain adaptation. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 400–410.