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

Shuqing Bian$^{1,2}$, Wayne Xin Zhao$^{1,2}$, Yang Song$^3$, Tao Zhang$^3$ and Ji-Rong Wen$^{1,2}$

$^1$School of Information, Renmin University of China
$^2$Beijing Key Laboratory of Big Data Management and Analysis Methods
$^3$BOSS Zhipin

{bianshuqing,jrwen}@ruc.edu.cn  batmanfly@gmail.com
{songyang,kylen.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 LinkedIn$^1$ 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,
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 $\{w_{i}^{(p)}\}_{i=1}^{n_p}$, and the $i$-th sentence $w_{i}^{(p)}$ 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 $\{w_{i}^{(r)}\}_{i=1}^{n_r}$, where $w_{i}^{(r)}$ 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
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 an match task. Given a target domain $t$, a training set $\mathcal{D}^t = \{(p_j^t, r_j^t, y_j)\}_{j=1}^{n_t}$ 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 $\mathcal{D}^s = \{(p_j^s, r_j^s, y_j)\}_{j=1}^{n_s}$ 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 matching 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 $\tilde{h}_{i,k}^{(p)}$ and $\tilde{h}_{i,k}^{(r)}$ 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., $h_{i,k}^{(p)} = [\tilde{h}_{i,k}^{(p)}, \tilde{h}_{i,k}^{(r)}]$. Since some words in a sentence are important than others, we apply the attention mechanism (Qin et al., 2018) to derive the sentence representation $h_i^{(p)} \in \mathbb{R}^{L_{p}}$ in a job posting:

$$h_i^{(p)} = \sum_{k=1}^{n_p} \alpha_k \cdot h_{i,k}^{(p)}, \quad (1)$$

where the attention weight $\alpha_k$ is calculated as

$$\alpha_k = \text{softmax}(a_k), \quad (2)$$

$$a_k = v_i^T \cdot \tanh(W_i h_{i,k}^{(p)} + b_i).$$

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

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p, r$</td>
<td>the job posting and resume</td>
</tr>
<tr>
<td>$n_p, n_r$</td>
<td>the number of sentences in the $p$ and $r$</td>
</tr>
<tr>
<td>$w_{i,p}^{(r)}$, $w_{i,r}^{(p)}$</td>
<td>the $i$-th word in the $p$ and $r$</td>
</tr>
<tr>
<td>${w_{i,p}^{(r)}, {w_{i,r}^{(p)}}$</td>
<td>the match result for a person-job pair based on the $p$ and $r$</td>
</tr>
<tr>
<td>$D^t$, $D^s$</td>
<td>the training set for the target domain $t$ and source domain $s$</td>
</tr>
<tr>
<td>$n_t, n_s$</td>
<td>the number of target labeled instances and source labeled instances</td>
</tr>
</tbody>
</table>

$h_{i,k}^{(p)}$, $h_{i,k}^{(r)}$ | 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 $r$ |

M | the match matrix ($\in \mathbb{R}^{K \times K}$) for the sentence similarity between the $p$ and $r$ |
| W | the weight matrix ($\in \mathbb{R}^{K \times K}$) |
| $m_{p,r}$ | the dimension size |
| @ | all the involved parameters |
| $h_{p}^{(p)}$, $h_{r}^{(r)}$ | the representations of the $p$ and $r$ |
| $y$ | the output score calculated by the Multi-Layer Perceptron |

$x_{i,scl}^{j,pcl}$, $x_{i,scl}^{j,pcl}$ | the transferable representation transformed by the original feature in the $p$ and $r$ |

$h_{i,p}^{(scl)}$, $h_{i,r}^{(scl)}$ | the enhanced job posting and resume sentence representation |

A | the shared weight matrix by all the domains |
| $B^p$, $B^r$ | the specific weight matrix to the domain $d$ |
| $\Theta^a$, $\Theta^r$ | the learned parameters for the auxiliary domain |
| $\hat{\Theta}^t$ | the domain-specific parameters for the target domain $t$ |
| $m_{p,r}$ | the new representation by the concatenation of the output from both components with the parameters $\Theta^a$ and $\Theta^r$ |

Table 1. The notations used in this paper.
resume, denoted by $h^p$ and $h^r$:

$$h^{(p)}_i = \sum_{i=1}^{n_p} \beta^{(p)}_i \cdot h^{(p)}_i,$$
$$h^{(r)}_i = \sum_{i=1}^{n_r} \beta^{(r)}_i \cdot h^{(r)}_i,$$

(3) (4)

where $\beta^{(p)}_i$ and $\beta^{(r)}_i$ 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 $M \in \mathbb{R}^{n_p \times n_r}$ for modeling the global semantic interactions. Formally, to calculate the sentence similarity in $M$, we apply a linear form as

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

(5)

where $W \in \mathbb{R}^{K \times K}$ is the match matrix, $h^{(p)}_i$ is the representation of the i-th sentence for a job posting $p$, and $h^{(r)}_{i'}$ 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 $m^{p,r}$ as

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

(6)

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

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

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

(7)

where $h^p$ and $h^r$ 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:

$$x^{r}_{i,\text{SCL}} = f_{\text{SCL}}(h^{r}_{i}).$$

(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:

$$\tilde{h}^p_i = h^p_i \oplus x^{r}_{i,\text{SCL}},$$

(9)

where “$\oplus$” is the vector concatenation operation. Similarly, we can obtain the enhanced resume sentence representation, denoted by $\tilde{h}^r_i$. With the $\tilde{h}^p_i$ and $\tilde{h}^r_i$, we can update the overall representations of $h^p$ and $h^r$ 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 $W$ 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 $W$ into a product of two smaller matrices $A \in \mathbb{R}^{1 \times K}$ and $B^d \in \mathbb{R}^{d \times K}$:

$$W^d = A^\top \cdot B^d,$$  \hspace{1cm} (10)

where $A$ is shared by all the domains, while $B^d$ 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 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 $M$. 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., $M \rightarrow m^{p,r}$) 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 $\hat{\Theta}^t$:

$$\tilde{m}^{p,r} = \text{ConvNet}(M; \hat{\Theta}^s) \oplus \text{ConvNet}(M; \hat{\Theta}^t).$$  \hspace{1cm} (11)

4.3 The Final Model and Training

To predict the match label for domain adaptation, we replace $h^p$, $h^r$ and $m^{p,r}$ in Eq. 7 with $h^p$, $h^r$, and $\tilde{m}^{p,r}$ 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 adaptation 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” (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
In our dataset, the ratio between matched and unmatched dates, provided by the online recruiting platform, is according to the acceptance status of the candidate-job pairs. The match label information is obtained from the interaction between minor domains, which have a limited number of person-job pairs. The match label information is obtained from the interaction between minor domains, which have a limited number of person-job pairs. 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 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.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Technology ACC</th>
<th>Technology F1</th>
<th>Sales ACC</th>
<th>Sales F1</th>
<th>Design ACC</th>
<th>Design F1</th>
<th>Cars ACC</th>
<th>Cars F1</th>
<th>Product ACC</th>
<th>Product F1</th>
<th>Journey ACC</th>
<th>Journey F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSSM</td>
<td>0.691</td>
<td>0.625</td>
<td>0.686</td>
<td>0.619</td>
<td>0.638</td>
<td>0.577</td>
<td>0.621</td>
<td>0.558</td>
<td>0.625</td>
<td>0.566</td>
<td>0.616</td>
<td>0.557</td>
</tr>
<tr>
<td>PJFNN</td>
<td>0.685</td>
<td>0.623</td>
<td>0.678</td>
<td>0.617</td>
<td>0.633</td>
<td>0.566</td>
<td>0.618</td>
<td>0.555</td>
<td>0.619</td>
<td>0.561</td>
<td>0.608</td>
<td>0.546</td>
</tr>
<tr>
<td>BPJFNN</td>
<td>0.681</td>
<td>0.619</td>
<td>0.669</td>
<td>0.611</td>
<td>0.627</td>
<td>0.565</td>
<td>0.611</td>
<td>0.554</td>
<td>0.610</td>
<td>0.552</td>
<td>0.602</td>
<td>0.545</td>
</tr>
<tr>
<td>APJFNN</td>
<td>0.694</td>
<td>0.630</td>
<td>0.688</td>
<td>0.626</td>
<td>0.641</td>
<td>0.580</td>
<td>0.627</td>
<td>0.563</td>
<td>0.630</td>
<td>0.571</td>
<td>0.619</td>
<td>0.559</td>
</tr>
<tr>
<td>Our model</td>
<td>0.713</td>
<td>0.642</td>
<td>0.709</td>
<td>0.638</td>
<td>0.659</td>
<td>0.600</td>
<td>0.644</td>
<td>0.582</td>
<td>0.648</td>
<td>0.591</td>
<td>0.637</td>
<td>0.575</td>
</tr>
</tbody>
</table>

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

There were more negative instances in original data. We randomly sample an equal number of negative instances to reduce the data bias.
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-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.
<table>
<thead>
<tr>
<th>source domain</th>
<th>methods</th>
<th>DESIGN</th>
<th>ACC</th>
<th>F1</th>
<th>CARS</th>
<th>ACC</th>
<th>F1</th>
<th>PRODUCT</th>
<th>ACC</th>
<th>F1</th>
<th>JOURNEY</th>
<th>ACC</th>
<th>F1</th>
</tr>
</thead>
<tbody>
<tr>
<td>TECHNOLOGY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Src-Only</td>
<td>0.556</td>
<td>0.503</td>
<td>0.542</td>
<td>0.484</td>
<td>0.550</td>
<td>0.491</td>
<td>0.528</td>
<td>0.469</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tgt-Only</td>
<td>0.659</td>
<td>0.600</td>
<td>0.644</td>
<td>0.582</td>
<td>0.648</td>
<td>0.591</td>
<td>0.637</td>
<td>0.575</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mixed</td>
<td>0.653</td>
<td>0.599</td>
<td>0.636</td>
<td>0.580</td>
<td>0.641</td>
<td>0.582</td>
<td>0.629</td>
<td>0.574</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCL-MI</td>
<td>0.657</td>
<td>0.598</td>
<td>0.645</td>
<td>0.586</td>
<td>0.649</td>
<td>0.591</td>
<td>0.640</td>
<td>0.577</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AE-SCL</td>
<td>0.672</td>
<td>0.610</td>
<td>0.659</td>
<td>0.602</td>
<td>0.662</td>
<td>0.603</td>
<td>0.650</td>
<td>0.594</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASP-MTL</td>
<td>0.675</td>
<td>0.616</td>
<td>0.661</td>
<td>0.600</td>
<td>0.666</td>
<td>0.608</td>
<td>0.654</td>
<td>0.595</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Our Model</td>
<td>0.696</td>
<td>0.638</td>
<td>0.680</td>
<td>0.625</td>
<td>0.687</td>
<td>0.627</td>
<td>0.669</td>
<td>0.610</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SALES</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Src-Only</td>
<td>0.547</td>
<td>0.485</td>
<td>0.535</td>
<td>0.474</td>
<td>0.562</td>
<td>0.603</td>
<td>0.539</td>
<td>0.484</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tgt-Only</td>
<td>0.659</td>
<td>0.600</td>
<td>0.644</td>
<td>0.582</td>
<td>0.648</td>
<td>0.591</td>
<td>0.637</td>
<td>0.575</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mixed</td>
<td>0.646</td>
<td>0.587</td>
<td>0.631</td>
<td>0.568</td>
<td>0.643</td>
<td>0.586</td>
<td>0.632</td>
<td>0.574</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCL-MI</td>
<td>0.635</td>
<td>0.596</td>
<td>0.642</td>
<td>0.582</td>
<td>0.632</td>
<td>0.594</td>
<td>0.645</td>
<td>0.583</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AE-SCL</td>
<td>0.665</td>
<td>0.606</td>
<td>0.654</td>
<td>0.597</td>
<td>0.666</td>
<td>0.608</td>
<td>0.657</td>
<td>0.598</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ASP-MTL</td>
<td>0.671</td>
<td>0.612</td>
<td>0.658</td>
<td>0.599</td>
<td>0.672</td>
<td>0.613</td>
<td>0.660</td>
<td>0.601</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Our Model</td>
<td>0.689</td>
<td>0.630</td>
<td>0.675</td>
<td>0.618</td>
<td>0.692</td>
<td>0.634</td>
<td>0.676</td>
<td>0.618</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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

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 $W$ 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 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.

![Diagram](image-url)
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


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.


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.