BOSS: A Bilateral Occupational-Suitability-Aware Recommender System for Online Recruitment

Xiao Hu*
Career Science Lab, BOSS Zhipin
Beijing, China
huxiao@kanzhun.com

Yuan Cheng*
Career Science Lab, BOSS Zhipin
Beijing, China
chengyuan02@kanzhun.com

Zhi Zheng
School of Data Science, University of Science and Technology of China
Hefei, China
Career Science Lab, BOSS Zhipin
Beijing, China
zhengzhi97@mail.ustc.edu.cn

Yue Wang
Career Science Lab, BOSS Zhipin
Beijing, China
wangyue05@kanzhun.com

Xinxin Chi
BOSS Zhipin
Beijing, China
chixinxin@kanzhun.com

Hengshu Zhu†
Career Science Lab, BOSS Zhipin
Beijing, China
zhuhengshu@gmail.com

ABSTRACT

With the rapid development of online recruitment platforms, a variety of emerging recommendation services have been witnessed for benefiting both job seekers and recruiters. While many researchers have studied the problem of reciprocal recommendation in two-sided markets (e.g., marriage market and real estate market), there is still a lack of in-depth understanding of the bilateral occupational preferences of different participants in the online recruitment market. To this end, in this paper, we propose a Bilateral Occupational-Suitability-aware recommender System (BOSS) for online recruitment, in consideration of the reciprocal, bilateral, and sequential properties of realistic recruitment scenarios simultaneously. To be specific, in BOSS, we first propose a multi-group-based mixture-of-experts (MoE) module to independently learn the preference representations of job seekers and recruiters. Then, with a specially-designed multi-task learning module, BOSS can progressively model the action sequence of recruitment process through a bilateral probabilistic manner. As a result, the reciprocal recommendations can be efficiently implemented by leveraging the product of different action probabilities of job seekers and recruiters. Finally, we have conducted extensive experiments on 5 real-world large-scale datasets as well as the online environment. Both online A/B test and offline experimental results clearly validate that our recommender system BOSS can outperform other state-of-the-art baselines with a significant margin.

*Both authors contributed equally to this research.
†Corresponding author.

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08…$15.00
https://doi.org/10.1145/3580305.3599783

CCS CONCEPTS

• **Information systems** → **Recommender systems**; **Social recommendation**.

KEYWORDS

Online recruitment, Reciprocal recommender systems, Multi-task learning

ACM Reference Format:
Xiao Hu, Yuan Cheng, Zhi Zheng, Yue Wang, Xinxin Chi, and Hengshu Zhu. 2023. BOSS: A Bilateral Occupational-Suitability-Aware Recommender System for Online Recruitment. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3580305.3599783

1 INTRODUCTION

Recent years have witnessed the rapid development and prevalence of online recruitment platforms. According to the latest report from MarketWatch [1], the global online recruitment market size was valued at USD 30.6 billion in 2021 and is expected to reach USD 48.7 billion by 2027. Along this line, to facilitate the user experience and improve the talent-job matching efficiency in the two-sided market, a variety of emerging recommendation services have been exploited. Different from traditional one-way recommendation scenarios, in the context of online recruitment, recommender systems are largely expected to satisfy the demands and preferences of both job seekers and recruiters at the same time.

In the literature, considerable research efforts have been made for reciprocal recommendation in different two-sided markets [17], such as marriage market [40] and real estate market [6]. Nevertheless, there is still a lack of in-depth understanding on the diverse occupational preferences of different participants in the online recruitment market. Indeed, there exist some unique characteristics of realistic recruitment scenario that should be taken into account. To be specific, the first is the bilateral property, which refers to the fact that job seekers and recruiters usually provide different information (e.g., job seekers provide demographic and social information, while recruiters provide company and job information) and

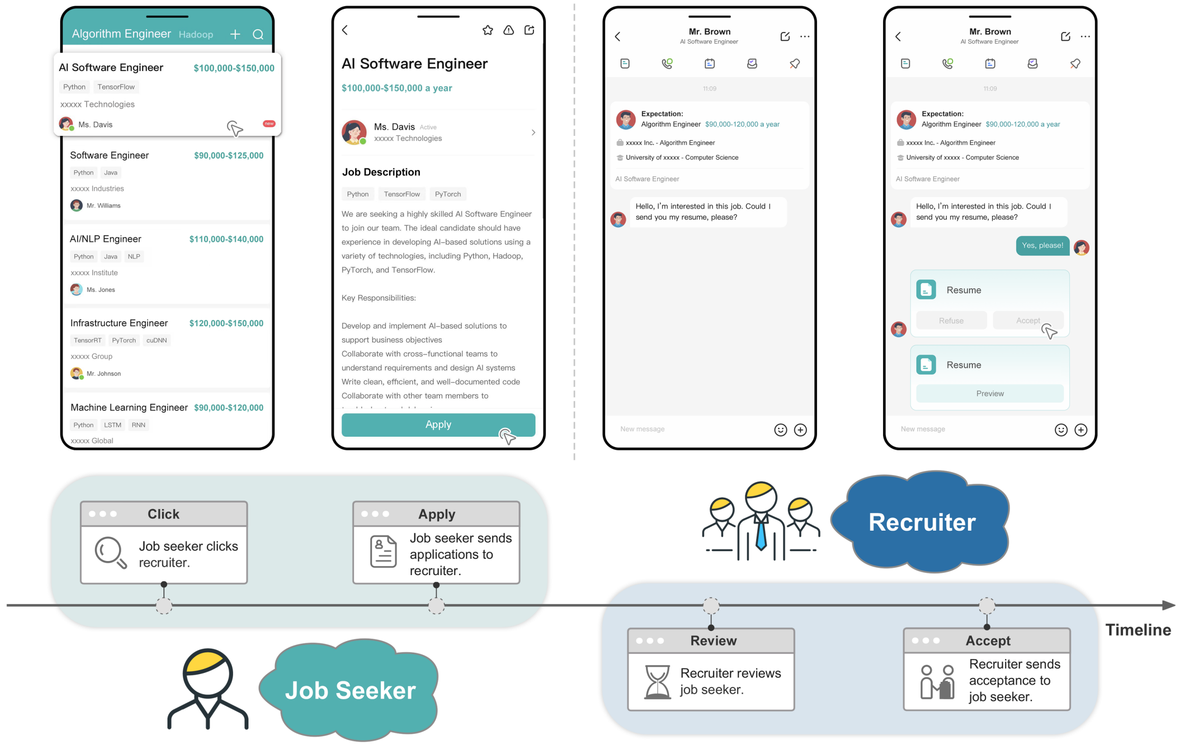

Figure 1: An illustration of matching process in online recruitment, where four distinct actions are respectively conducted by job seeks and recruiters in a sequential manner: (1) The job seeker can browse the recommended job list and *click* the job posting that she/he is interested in; (2) The job seeker can further *apply* for a chat with the recruiter of the job; (3) The recruiter can choose to *review* the application of job seeker for justifying the job suitability; (4) The recruiter will *accept* the application for the suitable candidate and send message for further arrangements, e.g., interview.

have different perspectives on the same features (e.g., job seekers prefer higher salary, while recruiters usually demand a lower price of candidates). The second is the reciprocal property, which refers to the need of satisfying the preferences of participants from both sides of the market (i.e., job seekers and recruiters), in order to support the practice of mutually beneficial exchanges. The last is the sequential property, which refers to the multi-action matching process in online recruitment, which includes at least two actions for proactive and reactive users to expose their decisions. For example, Figure 1 shows a typical multi-stage matching process of online recruitment.

To this end, in this paper, we propose to study the problem of reciprocal recommendation for online recruitment by introducing a Bilateral Occupational-Suitability-Aware recommender System (BOSS). In particular, BOSS is designed to simultaneously adapt the above-mentioned properties of realistic recruitment scenario, including reciprocal, bilateral, and sequential properties. Specifically, we first propose a multi-group-based mixture-of-experts (MoE) module to independently learn the representations of job seekers and recruiters. In this module, several expert models are arranged into two groups for modeling job seekers and recruiters respectively. Then, an auto-adaptive cross-group gate model is introduced with a specially-designed multi-task learning module, which can progressively model the action sequence of recruitment process through a bilateral probabilistic manner. Therefore, the reciprocal recommendations can be efficiently implemented by leveraging the product of different action probabilities of job seekers and recruiters. Finally, the major contributions of this paper can be summarized as follows.

- We study the problem of reciprocal recommendation for online recruitment, with an in-depth consideration of the bilateral occupational preferences of different participates in the two-sided market.
- We design a Bilateral Occupational-Suitability-aware recommender System (BOSS) for online recruitment, in consideration of the reciprocal, bilateral, and sequential properties of realistic recruitment scenarios simultaneously.
- We have conducted extensive experiments on five real-world large-scale datasets and online A/B test, which clearly validate the effectiveness of our recommender system BOSS.

Overview. The remainder of this paper is organized as follows. Section 2 provides a brief review of related works. In Section 3, we introduce the problem statement of this paper. In Section 4, we introduce the technical details of our recommender system BOSS. In Section 5, we report the extensive experimental results. Finally, in Section 6, we conclude the paper.

2 RELATED WORK

In this section, we summarize related work in the following two categories, namely multi-task recommender systems and recommendation for online recruitment.

2.1 Multi-task Recommender System

Recent years have witnessed the prosperity of recommender systems in a variety of applications [4, 20, 26, 30, 37, 41–44], and modern recommender systems have been further advanced by utilizing deep architectures for inferring the representation of user preferences and item characteristics [7, 8, 10, 12, 21, 29, 35]. In order to improve the performance of general recommender systems, some studies focus on utilizing multi-task learning methods for facilitating the user-item matching [3, 14, 15, 25]. For example, the research of auxiliary tasks in [3] claimed that multi-task learning models have potential to achieve better outcomes, since they have more adequate utilization of information by applying sharing bottom and isolated towers corresponding to subtasks. Based on the multi-task learning strategy, Entire Space Multi-task Model [15] made a good usage of the sequential property of action flows including click and conversion by auxiliary tasks following a conditional probability form, which achieves significant improvement against regular multi-task learning model. Ma *et al.* [14] introduced the idea of mixture-of-expert, points out that combinations of outputs from multiple expert models may result in a better outcomes in multi-task learning, especially when the subtasks have high correlations to each other. Based on the above, in this work, we follow the paradigm of multi-task learning to handle the matching process with multiple highly correlated labels.

2.2 Recommendation for Online Recruitment

In the field of online recruitment, various studies have been proposed to match job seekers with recruiters [11, 18, 19, 24, 28, 32, 45]. According to [24], traditional recommender systems can predict job seekers' preferences well, but it is important to consider recruiters' preferences if we want to improve the overall effectiveness of the system. Benjamin *et al.* [11] studied the interactive signals from both job seekers and recruiters, and found that the response from recruiters plays a more important role compared with the clicks of job seekers. Some context-aware recommendation algorithms have also been attempted [2, 9, 13, 34, 38] to design reciprocal recommender systems in online recruitment. Moreover, some researchers have attempted to implement hybrid recommender systems for online recruitment [9, 34] with both content-based algorithms and collaborative filtering to facilitate the matching between job seekers and recruiters. For example, Zheng *et al.* [27] introduced a system that merges the results from a CV-Recommender and a Job-Recommender to achieve the matching purpose between job seekers and recruiters. In [16], Voigt *et al.* used an approach named Bloom Filter that is a hybrid system based on calculating vector-like information to predict the matching rate. Ozcan and Guduru presented a system in [46], which can classify job seekers and recruiters based on their profile and preference along with their interactive information. Nielsen *et al.* [5] applied a GBDT-based model on the different feature sets for modeling users' preference. Moreover, there are also some deep learning-based reciprocal recommender systems [36, 39]. For example, in [39], researchers proposed a multi-task learning framework called biDeepFM. Yang *et al.* [36] proposed a novel loss to formulate the problem, which discovers the relationships between proactive/reactive job seekers and recruiters. Different from the above methods, in this paper, we propose a novel reciprocal recommendation framework with the consideration of the reciprocal, bilateral, and sequential properties of realistic recruitment scenarios simultaneously.

Table 1: Mathematical Notations.

Symbol	Description
$\mathcal{P}^j$	The profile information of job seeker j.
$\mathcal{P}^r$	The profile information of recruiter r.
$\mathcal{H}^j$	The historical behavior information of job seeker j.
$\mathcal{H}^r$	The historical behavior information of recruiter r.
C	The context and interactive information.
$\mathbf{V}$	The embedding matrix from input layer.
$\mathbf{E}$	The embedding matrix from MOE layer.
$\mathbf{s}_i$	The output for the i-th stage.
p_i^j	The i-th profile feature of job seeker j.
p_i^r	The i-th profile feature of recruiter r.
h_i^j	The i-th historical behavior of job seeker j.
h_i^r	The i-th historical behavior of recruiter r.
c_i	The i-th context and interactive feature in C.

3 PROBLEM STATEMENT

In this paper, we studied the reciprocal recommendation problem derived from a popular online recruitment platform in China, namely BOSS Zhipin [1]. As shown in Figure 1, the jobseeker-recruiter matching process in realistic recruitment scenario mainly consists of four stages. First, a list of recommended job descriptions are shown to the job seekers. Each job seeker can browse the recommended job lists and *click* the job posting he/she is interested in. After the clicking, the job seeker can further *apply* for a chat with the recruiter of the job. Then, if a recruiter receives the application of job seekers, he/she can choose to *review* the detailed information for justifying the job suitability. Finally, if the recruiter is satisfied with the job seeker, he/she can *accept* the application and send message for further arrangements, e.g., interview. With this scenario, in this paper, the goal of reciprocal recommendation for online recruitment is to recommend jobs for job seekers that are most likely to reach the stage of *accept*.

In order to solve the above problem, we propose to exploit the profile and historical behavior information of both job seekers and

[1] https://www.zhipin.com

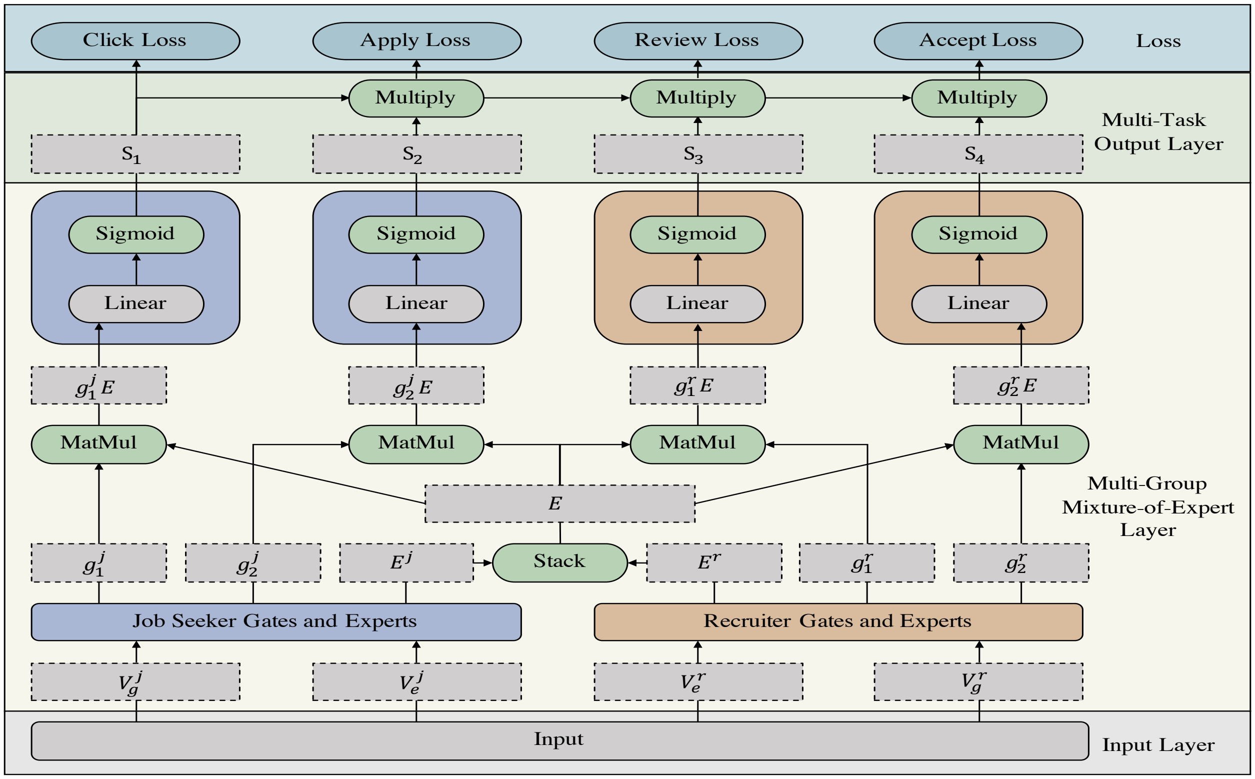

Figure 2: The model architecture overview of BOSS.

recruiters. Specifically, given a job seeker j, we format the profile and historical behavior information of j as $\mathcal{P}^j = [p_1^j, \ldots, p_m^j]$ and $\mathcal{H}^j = [h_1^j, \ldots, h_n^j]$, where p_i^j is the i-th profile feature of j and h_i^j is the hashed ID of the i-th recruiter with *accept* stage in the history. m and n are the lengths of the profile and historical behavior list of j. Similarly, for recruiter r, we denote the profile and historical behavior information of r as $\mathcal{P}^r = [p_1^r, \ldots, p_s^r]$ and $\mathcal{H}^r = [h_1^r, \ldots, h_t^r]$, where p_i^r is the i-th profile feature of r and h_i^r is the id of the i-th job seeker with *accept* stage in the history. s and t are the lengths of the profile and historical behavior list of r. Furthermore, we can format each impression log in the training data $\mathcal{D}$ as $[\mathcal{P}^j, \mathcal{H}^j, \mathcal{P}^r, \mathcal{H}^r, C, label]$, where $C = [c_1, \ldots, c_k]$ is the context feature vector, and *label* is the stage indicator, i.e., 1 for *click*, 2 for *apply*, 3 for *review*, 4 for *accept*, and 0 for unclicked, respectively. Based on the above, we can formally define the reciprocal recommendation problem as follows.

Definition 1 (Problem Statement). *Given a job seeker j with profile $\mathcal{P}^j$ and historical behavior $\mathcal{H}^j$, a recruiter r with profile $\mathcal{P}^r$ and historical behavior $\mathcal{H}^r$, and the context information C, the goal of our reciprocal recommendation task is to estimate the probability that j and r can reach the accept stage by learning from the training data $\mathcal{D}$, i.e., $f : j, r \rightarrow \mathbb{R}$.*

4 TECHNICAL DETAILS

In this section, we will introduce the technical details of our recommender system BOSS in detail. As shown in Figure 2, the model framework of BOSS mainly consists of three components: *embedding layer*, *bilateral isolated hidden layer*, and *multi-task output layer*. We will introduce the details of each component in the following subsections. The mathematical notations used throughout this paper are listed in Table 1.

4.1 Input Layer

Given a job seeker j with profile features $\mathcal{P}^j = [p_1^j, \ldots, p_m^j]$ and a recruiter r with profile features $\mathcal{P}^r = [p_1^r, \ldots, p_s^r]$, we first convert $\mathcal{P}^j$ and $\mathcal{P}^r$ to embedding matrices $\mathbf{P}^j = [\mathbf{p}_1^j, \ldots, \mathbf{p}_m^j]$ and $\mathbf{P}^r = [\mathbf{p}_1^r, \ldots, \mathbf{p}_s^r]$, where $\mathbf{p}_i^j$ is the corresponding embedding vector of p_i^j, and $\mathbf{p}_i^r$ is the corresponding embedding vector of p_i^r. For the historical behaviors $\mathcal{H}^j = [h_1^j, \ldots, h_n^j]$ and $\mathcal{H}^r = [h_1^r, \ldots, h_t^r]$, we first get the profile embedding matrix corresponding to each item in $\mathcal{H}^j$ and $\mathcal{H}^r$ based on the above profile embedding process. Then, we get the embedding matrix corresponding to $\mathcal{H}^j$ and $\mathcal{H}^r$ by mean pooling as:

$$\begin{aligned} \mathbf{H}^j &= \mathrm{Mean}([\mathbf{H}_1^j, \ldots, \mathbf{H}_n^j]), \\ \mathbf{H}^r &= \mathrm{Mean}([\mathbf{H}_1^r, \ldots, \mathbf{H}_t^r]), \end{aligned} \tag{1}$$

where $\mathbf{H}_i^j$ and $\mathbf{H}_i^r$ are the corresponding embedding matrix of h_i^j and h_i^r. We also convert the context information C to embedding matrix $\mathbf{C} = [\mathbf{c}_1, \ldots, \mathbf{c}_k]$, where $\mathbf{c}_i$ is the corresponding embedding vector of c_i. After getting the above embedding matrix, we concatenate the embedding matrices as the final input embedding matrix as:

$$\mathbf{V} = \mathrm{Concat}([\mathbf{P}^j, \mathbf{P}^r, \mathbf{H}^j, \mathbf{H}^r, \mathbf{C}]). \tag{2}$$

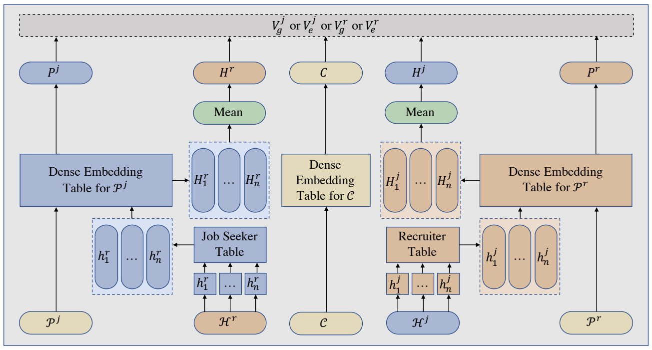

Figure 3: The architecture of input layer of BOSS.

where Concat denotes the function concatenates multiple matrices or vectors along the first dimension.

Figure 3 provides a detailed illustration of the input layer.

4.2 Multi-Group Mixture-of-Expert Layer

Different from traditional Click-Through-Rate (CTR) prediction problem which only needs to model the click-through behaviors of users, in our online recruitment scenario, there are two major participants (i.e., job seeker and recruiter) and four types of actions, (i.e., click, apply, review, and accept). Therefore, for the fine-grained modeling of both job seekers and recruiters, here we propose a multi-group MoE (i.e., Mixture-of-Expert) method in BOSS. Specifically, we first utilize two different groups of MoE models to capture the different characteristics of job seekers and recruiters. Furthermore, in each MoE model, we utilize a feature interaction (FI) unit for capturing the interaction of different features better. After that, in order to fuse the output from these MoE models, each group also contains two different gate layers. The two different outputs from the job seek group are utilized for click and apply prediction, while the outputs from the recruiter group are utilized for review and accept prediction. We will address each component in detail in the following two subsections.

4.2.1 Interaction-aware MoE layers. Based on Section 4.1, given the feature embedding matrix $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_l]$, we utilize inner product to capture the feature interaction as:

$$v_{i,j} = \mathbf{v}_i^\top \mathbf{v}_j. \tag{3}$$

By this way, we can get the feature interaction vector corresponding to $\mathbf{V}$ as $\mathbf{v} = [v_{1,1}, \ldots, v_{l,l}]$, and we combine $\mathbf{v}$ and the vectors in $\mathbf{V}$ as the input for the expert models as:

$$\hat{\mathbf{v}} = \text{Concat}([\text{Flatten}(\mathbf{V}), \mathbf{v}]). \tag{4}$$

where Flatten denotes the function transforms a matrix into a vector along the first dimension.

In this paper, we utilize MLP layers as expert models, and denotes the function of expert model as "MLP". Then the output of MoE models can be format as:

$$\mathbf{E} = \text{Stack}([\text{MLP}_1(\hat{\mathbf{v}}), \ldots, \text{MLP}_z(\hat{\mathbf{v}})]), \tag{5}$$

where Stack denotes the function combines multiple vectors into a matrix and z is the number of experts.

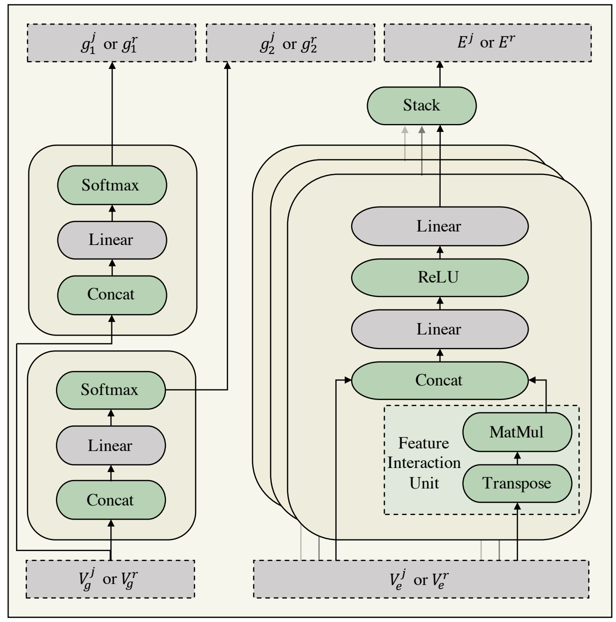

Figure 4: The architecture of expert models of BOSS.

4.2.2 Multi-Gate Output Layer. In our recruitment scenario, the click and apply stage are dominated by the preference of job seekers, while the review and accept stage are dominated by the preference of recruiters. In this paper, we design a multi-gate output layer for both of the two expert groups to get the output for different stages adaptively. Specifically, each group has two gate layer to fuse the outputs of the two expert models, and the input of the two gate layers are shared. Given the input embedding matrix $\mathbf{V}$, the output of each gate layer can be obtained as:

$$\mathbf{g} = \text{Softmax}(\text{MLP}(\text{Flatten}(\mathbf{V}))). \tag{6}$$

Note that there is no FI unit in gate layers. Based on the Eq 5, we can get the outputs from MoE models corresponding to job seeker j and recruiter r as $\mathbf{E}^j$ and $\mathbf{E}^r$, and the outputs from multi-gate layers as $\mathbf{g}_1^j$, $\mathbf{g}_2^j$, $\mathbf{g}_1^r$, and $\mathbf{g}_2^r$. We concatenate $\mathbf{E}^j$ and $\mathbf{E}^r$ as:

$$\mathbf{E} = \text{Concat}([\mathbf{E}^j, \mathbf{E}^r]), \tag{7}$$

and the final output for the four stages can be calculated as:

$$\begin{aligned} \mathbf{s}_1 &= \text{MLP}(\mathbf{g}_1^j\mathbf{E}), \quad \mathbf{s}_2 = \text{MLP}(\mathbf{g}_2^j\mathbf{E}), \\ \mathbf{s}_3 &= \text{MLP}(\mathbf{g}_1^r\mathbf{E}), \quad \mathbf{s}_4 = \text{MLP}(\mathbf{g}_2^r\mathbf{E}). \end{aligned} \tag{8}$$

Figure 4 provides a detailed demonstration of the multi-group MoE layer in BOSS.

4.3 Multi-task Training

As we mentioned before, there are four progressive stages in our online recruitment platform, respectively click, apply, review, and accept. We can find that the achievement of each stage depends on the previous stage. Therefore, in this paper, we propose to utilize conditional probability to describe the whole process. For example, we can format the probability of the achievement of accept stage

Table 2: Statistics of offline experimental datasets.

Data	#Sample		
	Train	Valid	Test
Technique	7,274,559	304,844	305,097
Manufacturing	5,575,962	218,121	218,480
Service	9,577,657	378,208	377,764
Marketing	7,249,080	295,622	295,945
Arts	3,057,923	126,221	125,665

as the product of four conditional probabilities:

$$p(accept) = p(click) \times p(apply|click) \times p(review|click, apply) \times p(accept|click, apply, review). \tag{9}$$

In order to model the above process, based on the output of the multi-gate output layer, we calculate four probabilities as:

$$\begin{aligned} p_1 &= \sigma(\text{Linear}(\mathbf{s}_1)),\ p_2 = \sigma(\text{Linear}(\mathbf{s}_2)), \\ p_3 &= \sigma(\text{Linear}(\mathbf{s}_3)),\ p_4 = \sigma(\text{Linear}(\mathbf{s}_4)), \end{aligned} \tag{10}$$

where $\sigma(x) = 1/(1 + \exp(-x))$. Based on the above, we can obtain the probability of the four stages as follows:

$$\begin{aligned} &p(click) = p_1,\ p(apply) = p_1 \times p_2,\ p(review) = p_1 \times p_2 \times p_3, \\ &p(accept) = p_1 \times p_2 \times p_3 \times p_4. \end{aligned} \tag{11}$$

In order to train the parameters in our model, we adapt binary cross-entropy loss for the k-th stage as:

$$\mathcal{L}_k = -\frac{1}{N_k} \sum_{i=1}^{N_k} (y_i^k \log(\prod_{j=1}^{k} p_j) + (1 - y_i^k) \log(\prod_{j=1}^{k} p_j)), \tag{12}$$

where N_k is the number of training data for k-th stage and y_i^k is the ground truth label for the i-th data. Finally, we sum up the loss function for each stage to get the final loss function as:

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4. \tag{13}$$

5 EXPERIMENTS

In this section, we first present the experiment setup including the introduction of the real-world datasets and baselines we used for comparison. Then, we evaluate the proposed model and the baselines comprehensively. Meanwhile, we provide the information of ablation studies and discussions. Finally, we present the online experimental results in detail.

5.1 Experiment Setups

5.1.1 Data Description and Preparation. The real-world datasets used for offline experiments were collected from BOSS Zhipin, a popular online recruitment platform in China. We used the impression logs on this platform from Aug 14, 2022 to Sep 10, 2022. To protect the privacy of users and platform operators, all the identification information and all sensitive commercial information in these impression log were hashed or removed. Since the size of entire log file is tremendous, we randomly picked partial job seekers by mod calculation on their hashed IDs to ensure that every selected job seeker's historical behaviors are completed. As mentioned in Section 3, each impression log contains the profile features of job seekers (e.g., salary expectation), the profile features of recruiters (e.g., skills requirement of the job posting), and context information (e.g., the impression time). Furthermore, in order to construct a completed picture of a specific job seeker or recruiter, we also took the historical behaviors of job seekers and recruiters into consideration. For each feature field in the impression log is either categorical or numerical, we bucketized each numerical field into buckets by frequency, and built a vocabulary dictionary for every categorical field. After remapping, only the index numbers were presented in our datasets. Finally, we also collected the action stage of each impression log as mentioned before. For more comprehensive study on the performance of different models, we split the datasets into several subsets by taxonomy of career clusters, i.e., information technology, manufacturing, services, marketing, and arts. For each dataset, after applying ascending sort based on the impression time, we selected the first 90% samples as the training set, and equally divided the rest of samples in to validation set and test set. The detailed statistical information of our datasets is summarized in Table 2.

5.1.2 Baseline Methods. To evaluate the performance of BOSS, we select a number of representative frameworks for multi-task and multi-label learning as baselines.

- **Single Model (Binary-class)**: This denotes a single model without framework to learn the *accept* label.
- **Single Model (Multi-class)**: This denotes a single model without framework to learn the *impression*, *click*, *apply*, *review* and *accept* labels.
- **Shared Bottom**: This denotes a framework that contains four independent three-layers feed-forward networks supported by a shared bottom to learn the *click*, *apply*, *review* and *accept* labels respectively [3].
- **Multi-gate Mixture-of-Experts** (MMoE): This denotes a MoE framework with multiple gates. According to the four stages in our scenario, four gates independently distribute the weights for mixing the output of expert models for three-layers feed-forward networks to learn the four labels respectively [14].
- **Customized Gate Control** (CGC): This represents a MoE-liked framework. However, in contrast to the conventional MoE structure that utilizes a commonly shared pool of experts for all subtasks, this framework introduces the concept of dedicated experts for each subtask, in addition to the shared ones, enables the framework to possess an expanded capacity for learning each subtask individually [30].
- **Progressive Layered Extraction** (PLE): Echoing the idea of CGC framework, this represents a MoE-liked framework that incorporates the notion of progressive layers. These layers essentially embody the CGC structure replicated across multiple stages, facilitating a sequential and in-depth processing of information. [30].
- **Entire Space Multi-task Model** (ESMM): This denotes a multi-task learning model similar to the shared bottom framework. In our scenario, this framework contains four dependent three-layers feed-forward networks supported by a shared bottom, and output four probabilities corresponding

to *click*, *apply*, *review* and *accept* labels and the total loss is the sum of four binary cross entropy losses [15].

Moreover, we select some CTR prediction models as the FI unit inside of the aforementioned baseline frameworks and our BOSS model for feature interactive information extraction:

- **Not Applicable** (NA): This denotes that no FI units are deployed in the model.
- **Cross Network** (CN): This denotes a structure that applies feature crossing calculation by adding learn-able weights and bias to add multiple features for achieving the feature interactive information [33].
- **Factorization Machines** (FM): This denotes a structure based on matrix factorization for user-item recommender system. It expresses interactive information by second-order feature interaction [23].
- **Inner Product Layer** (IP): This denotes a layer previously described in Section 3. Several models applied inner product layer and obtained outstanding performances on various datasets [22].

We use the Area Under the ROC Curve (AUC) score as our evaluation metric, which represents the probability that a model ranks a positive case higher than a negative case.

5.1.3 Implementation Details. We used Adam as the optimizer with its learning rate set to be 0.001, β_1 set to be 0.9, and β_2 set to be 0.999. The batch size of training set is 1,024 and it is shuffled during every training epoch. For model setup, the unified embedding dimension for the embedding layer is 16. For BOSS, the two-hidden-layers feed-forward network in expert model consists of 200 neurons and 80 neurons respectively, and the dimension of one-hidden-layers feed-forward network consists of 40 neurons. For other baseline frameworks, the three-layers feed-forward networks consists of 200 neurons, 80 neurons and 40 neurons respectively. When initializing a model at the beginning of training process, all parameters are initialized by the truncated normal distribution with a standard deviation of 0.01. For all baseline frameworks and BOSS, neither dropout nor regularization are activated.

We conducted each of all the experiments 20 times and used the average value as the final result. The t-test was used to identify the statistically significant differences between the performances of BOSS and the best baseline. Experimental results validate that most of the results of BOSS have passed the t-test with a P-value of 0.01.

5.2 Overall Performance

To demonstrate the effectiveness of BOSS, we compared it with the baseline methods, and corresponding results are shown in Table 3. From the results, we can get the following observations: (1) We can find that the multi-task learning based models perform better than Single Model, which demonstrates the effectiveness of utilizing auxiliary tasks to improve the performance on accept prediction. (2) MoE models can achieve better performance than the shared bottom model, which indicates the necessity of utilizing MoE models as the feature extractor in our model. (3) Our BOSS model can achieve better performance than MMoE and ESMM models, which clearly demonstrates the effectiveness of our multi-group MoE architecture. (4) Models with IPNN layer as the FI unit can achieve

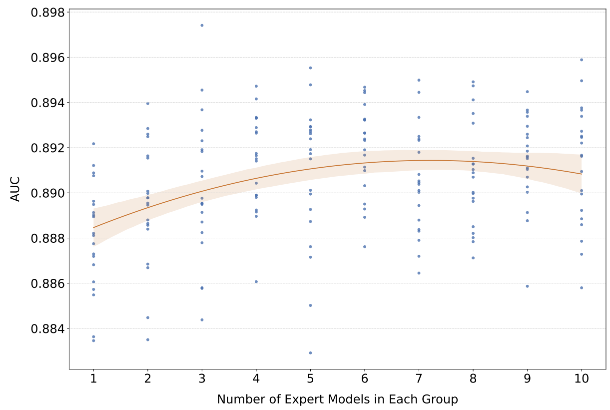

Figure 5: The performance of BOSS with different numbers of expert models.

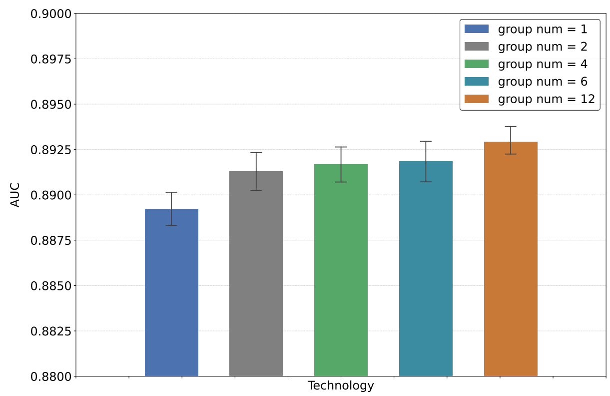

Figure 6: The performance of BOSS with different group numbers in multi-group mixture-of-expert layer.

better performance than models with other units, which proves the necessity for capturing the feature interaction in our datasets.

5.3 Parameter Sensitivity

We also investigated the parameter sensitivity of our BOSS model in the experiments. First, we evaluated how the number of expert models in the MoE architecture affects the performance of different models on prediction. The results are shown in Figure 5. Note that, since the results on five datasets have similar trends, here we only introduce the result on the *Techlonogy* dataset. From the results we can find that as the number of experts increases, the model performance first increases and then decreases. With the consideration of both model performance and the computational cost, we selected the number of the expert models as 6 in our experiments. Then, to evaluate how the number of groups in our multi-group based MoE models affect the performance of models, we tested the performance of our model with 1, 2, 4, 6, and 12 groups of MoE models, and the results are shown in Figure 6. Note that to make sure the fairness of the comparison, we have controled the total number of experts as the same.

Table 3: Overall experimental results of BOSS and baselines.

Data	FI Unit	AUC							
		Single Model (Binary-class)	Single Model (Multi-class)	Shared Bottom	MMoE	CGC	PLE	ESMM	BOSS
Tech-nology	NA	0.8713 ± 0.0014	0.8746 ± 0.0037	0.8773 ± 0.0022	0.8804 ± 0.0019	0.8793 ± 0.0020	0.8815 ± 0.0028	0.8844 ± 0.0022	**0.8863 ± 0.0021**
	CN	0.8713 ± 0.0018	0.8780 ± 0.0028	0.8764 ± 0.0016	0.8798 ± 0.0026	0.8773 ± 0.0020	0.8804 ± 0.0030	0.8832 ± 0.0030	**0.8871 ± 0.0023**
	FM	0.8737 ± 0.0021	0.8747 ± 0.0035	0.8819 ± 0.0018	0.8812 ± 0.0032	0.8813 ± 0.0024	0.8831 ± 0.0032	0.8835 ± 0.0023	**0.8872 ± 0.0031**
	IP	0.8782 ± 0.0019	0.8746 ± 0.0043	0.8858 ± 0.0032	0.8863 ± 0.0030	0.8849 ± 0.0035	0.8875 ± 0.0030	0.8869 ± 0.0025	**0.8918 ± 0.0021**
Manu-facturing	NA	0.8319 ± 0.0026	0.8059 ± 0.0059	0.8388 ± 0.0040	0.8460 ± 0.0045	0.8435 ± 0.0052	0.8466 ± 0.0034	0.8502 ± 0.0027	**0.8527 ± 0.0046**
	CN	0.8324 ± 0.0028	0.8051 ± 0.0031	0.8400 ± 0.0022	0.8446 ± 0.0041	0.8418 ± 0.0030	0.8457 ± 0.0043	0.8509 ± 0.0030	**0.8522 ± 0.0037**
	FM	0.8323 ± 0.0042	0.8056 ± 0.0053	0.8436 ± 0.0046	0.8486 ± 0.0046	0.8482 ± 0.0043	0.8459 ± 0.0052	0.8510 ± 0.0050	**0.8528 ± 0.0036**
	IP	0.8353 ± 0.0036	0.8068 ± 0.0087	0.8442 ± 0.0040	0.8500 ± 0.0057	0.8485 ± 0.0048	0.8491 ± 0.0079	0.8534 ± 0.0045	**0.8559 ± 0.0060**
Service	NA	0.8383 ± 0.0038	0.7848 ± 0.0058	0.8427 ± 0.0036	0.8418 ± 0.0053	0.8404 ± 0.0050	0.8422 ± 0.0041	**0.8454 ± 0.0038**	0.8452 ± 0.0042
	CN	0.8367 ± 0.0038	0.7851 ± 0.0034	0.8402 ± 0.0036	0.8411 ± 0.0030	0.8400 ± 0.0026	0.8425 ± 0.0041	0.8441 ± 0.0020	**0.8463 ± 0.0036**
	FM	0.8372 ± 0.0035	0.7810 ± 0.0052	0.8427 ± 0.0035	0.8426 ± 0.0041	0.8439 ± 0.0044	0.8425 ± 0.0047	0.8459 ± 0.0050	**0.8475 ± 0.0047**
	IP	0.8431 ± 0.0045	0.7886 ± 0.0062	0.8458 ± 0.0051	0.8436 ± 0.0045	0.8444 ± 0.0051	0.8453 ± 0.0051	0.8487 ± 0.0049	**0.8517 ± 0.0042**
Market-ing	NA	0.7735 ± 0.0030	0.7626 ± 0.0034	0.7861 ± 0.0028	0.7917 ± 0.0041	0.7880 ± 0.0036	0.7910 ± 0.0047	0.7931 ± 0.0035	**0.7952 ± 0.0035**
	CN	0.7757 ± 0.0030	0.7635 ± 0.0038	0.7847 ± 0.0026	0.7890 ± 0.0033	0.7876 ± 0.0042	0.7908 ± 0.0031	**0.7927 ± 0.0033**	0.7910 ± 0.0034
	FM	0.7777 ± 0.0044	0.7652 ± 0.0055	0.7884 ± 0.0030	0.7891 ± 0.0046	0.7875 ± 0.0065	0.7890 ± 0.0051	0.7935 ± 0.0036	**0.7952 ± 0.0047**
	IP	0.7819 ± 0.0039	0.7680 ± 0.0053	0.7934 ± 0.0044	0.7941 ± 0.0046	0.7908 ± 0.0064	0.7948 ± 0.0057	0.7996 ± 0.0035	**0.8006 ± 0.0032**
Arts	NA	0.8724 ± 0.0032	0.8535 ± 0.0035	0.8755 ± 0.0020	0.8802 ± 0.0037	0.8773 ± 0.0039	0.8838 ± 0.0039	0.8872 ± 0.0040	**0.8903 ± 0.0037**
	CN	0.8737 ± 0.0029	0.8570 ± 0.0048	0.8769 ± 0.0025	0.8817 ± 0.0033	0.8795 ± 0.0043	0.8820 ± 0.0035	0.8862 ± 0.0030	**0.8885 ± 0.0029**
	FM	0.8696 ± 0.0044	0.8536 ± 0.0048	0.8783 ± 0.0038	0.8821 ± 0.0038	0.8809 ± 0.0049	0.8828 ± 0.0043	0.8873 ± 0.0038	**0.8914 ± 0.0040**
	IP	0.8767 ± 0.0030	0.8549 ± 0.0066	0.8820 ± 0.0045	0.8884 ± 0.0038	0.8869 ± 0.0050	0.8910 ± 0.0044	0.8916 ± 0.0040	**0.8945 ± 0.0041**

Table 4: Inference time evaluation for different group numbers of expert models.

# Groups	#Parameters	Inference Time Cost per 1M Samples
1	27,880,124	78.4 ± 7.1 sec
2	29,175,068	88.1 ± 8.6 sec
4	31,813,364	125.9 ± 14.5 sec
6	34,403,252	140.6 ± 17.4 sec
12	42,124,508	210.3 ± 20.9 sec

From the result we can find that as the number of groups increases, the model performance increases simultaneously. However, as shown in Table 4, the computational cost increases significantly when the number of groups are large. Moreover, the improvement from 1 group to 2 groups is significant in Technology, Manufacturing and Arts datasets, supported that the multi-group strategy intrinsically helped the model to adapt the bilateral property in online recruitment. Therefore, to achieve the best trade-off between the performance and computational cost, we selected the group number as 2 in our experiments. Furthermore, we can find that the model performs the worst if we set the group number as 1, which is the same with most of the existing shared-embedding MMoE models. The relational behind this is that there are multiple participants and stages in our bilateral recommendation system, therefore, it is necessary to utilize multi-group of embedding for capturing information from different perspective.

To further prove the above perspective, we conducted extra experiments on the apply rate prediction of job seekers to check whether shared embedding works if there is only one participant. Notably, as shown in Figure 7, the experimental result demonstrates

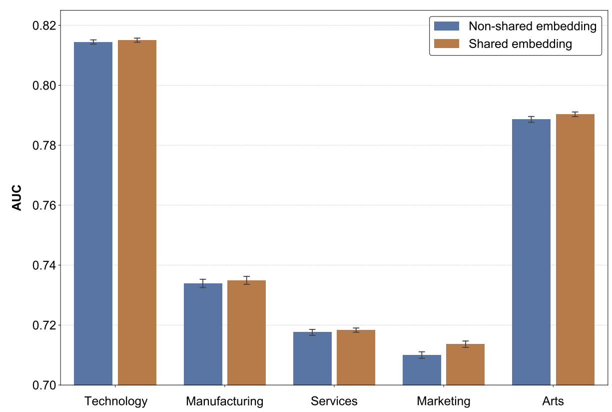

Figure 7: The performance of BOSS with different embedding setups for predicting the applications of job seekers.

that shared embedding is a better choice for job seeker apply rate prediction, which validates our conclusion.

5.4 Case Study

In order to validate that our two groups of expert models can capture the difference between the two participants of our platform, i.e., job seekers and recruiters, we randomly selected several data samples and computed corresponding output from the two groups of MoE models. We further projected them into a two-dimensional space with t-SNE [31]. The result is shown in Figure 8. We can find that the output from different expert models in different groups forms different clusters. The result demonstrates the effectiveness of our multi-group MoE models in terms of capturing the difference between the two participants.

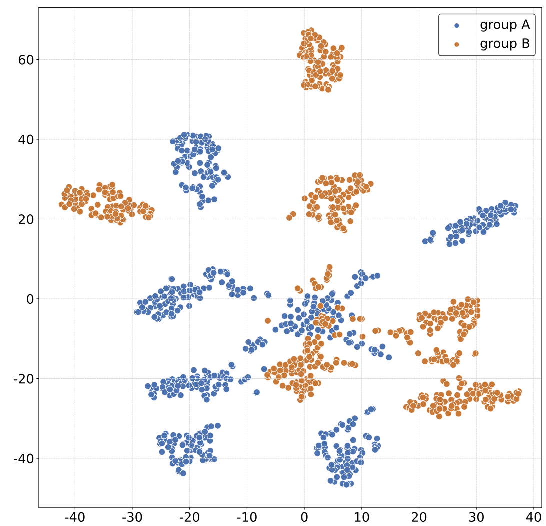

Figure 8: The output differences of two groups of expert models.

5.5 Online A/B Testing

To further validate the effectiveness of our proposed BOSS, we conducted a series of A/B tests in our online recruitment platform. The online experiment was conducted on a representative sample of users of the platform. Specifically, we chose the job category of information technology as target scenario, and randomly picked 50% users as the control group, and the other 50% users as the experimental group.

Figure 9 shows the results of online A/B testing in over half a month compared with the baseline framework. We discover that BOSS outperforms the result of the baseline framework significantly. Generally, BOSS achieved 6.15% average performance gains against the introduced baselines. This demonstrates the effectiveness of our proposed framework, and BOSS is deployed online to serve a main traffic.

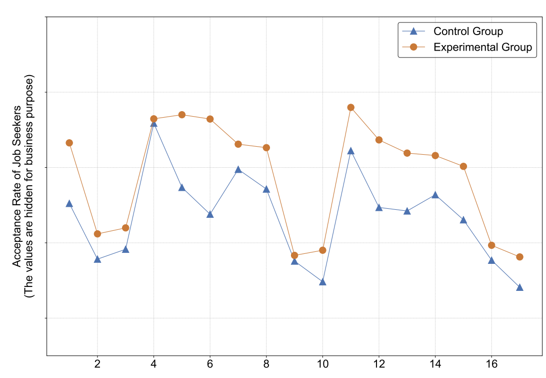

Figure 9: The online experimental results of Group A (Control Group) and Group B (Experimental Group).

6 CONCLUDING REMARKS

In this paper, we studied the problem of reciprocal recommendation for online recruitment. Specifically, we proposed a Bilateral Occupational-Suitability-aware recommender System (BOSS), which simultaneously considers of the reciprocal, bilateral, and sequential properties of realistic recruitment scenarios. In BOSS, we first proposed a multi-group based mixture-of-experts module to independently learn the preference representations of job seekers and recruiters. Then, with a specially-designed multi-task learning module, the BOSS can progressively model the action sequence of recruitment process through a bilateral probabilistic manner. As a result, the reciprocal recommendations can be efficiently implemented by leveraging the product of different action probabilities of job seekers and recruiters. Finally, the extensive experiments conducted on 5 real-world large-scale datasets and online environment with state-of-the-art baselines have clearly validated the effectiveness of our recommender system BOSS.

REFERENCES

[1] 2022. Online Recruitment Market 2022: Global Industry Analysis, Size, Share, Growth, Trends, and Forecast 2027. https://www.marketwatch.com/press-release/online-recruitment-market-2022-global-industry-analysis-size-share-growth-trends-and-forecast-2027-2022-11-03 Accessed on: January 13, 2023.

[2] Nikolaos D. Almalis, George A. Tsihrintzis, and Nikolaos Karagiannis. 2014. A content based approach for recommending personnel for job positions. In *IISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications*. 45–49. https://doi.org/10.1109/IISA.2014.6878720

[3] Rich Caruana. 1993. Multitask Learning: A Knowledge-Based Source of Inductive Bias. In *Proceedings of the Tenth International Conference on International Conference on Machine Learning* (Amherst, MA, USA) *(ICML'93)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 41–48.

[4] Le Dai, Yu Yin, Chuan Qin, Enhong Chen, and Hui Xiong. 2022. Decomposing Complementary and Substitutable Relations for Intercorporate Investment Recommendation. In *2022 IEEE International Conference on Data Mining (ICDM)*. 909–914. https://doi.org/10.1109/ICDM54844.2022.00106

[5] Snorre S. Frid-Nielsen. 2019. Find My next Job: Labor Market Recommendations Using Administrative Big Data. In *Proceedings of the 13th ACM Conference on Recommender Systems* (Copenhagen, Denmark) *(RecSys '19)*. Association for Computing Machinery, New York, NY, USA, 408–412. https://doi.org/10.1145/3298689.3346992

[6] Alireza Gharahighehi, Konstantinos Pliakos, and Celine Vens. 2021. Recommender Systems in the Real Estate Market—A Survey. *Applied Sciences* 11, 16 (2021). https://doi.org/10.3390/app11167502

[7] Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, and Qing He. 2022. A Survey on Knowledge Graph-Based Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering* 34, 8 (Aug 2022), 3549–3568. https://doi.org/10.1109/TKDE.2020.3028705

[8] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web* (Perth, Australia) *(WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 173–182. https://doi.org/10.1145/3038912.3052569

[9] Wenxing Hong, Lei Li, Tao Li, and Wenfu Pan. 2013. IHR: An Online Recruiting System for Xiamen Talent Service Center. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Chicago, Illinois, USA) *(KDD '13)*. Association for Computing Machinery, New York, NY, USA, 1177–1185. https://doi.org/10.1145/2487575.2488199

[10] Yunfan Hu, Zhaopeng Qiu, and Xian Wu. 2022. Denoising Neural Network for News Recommendation with Positive and Negative Implicit Feedback. In *Findings of the Association for Computational Linguistics: NAACL 2022*. Association for Computational Linguistics, Seattle, United States, 2320–2329. https://doi.org/10.18653/v1/2022.findings-naacl.178

[11] Benjamin Kille, Fabian Abel, Balázs Hidasi, and Sahin Albayrak. 2015. Using Interaction Signals for Job Recommendations. In *Mobile Computing, Applications, and Services*, Stephan Sigg, Petteri Nurmi, and Flora Salim (Eds.). Springer International Publishing, Cham, 301–308.

[12] Xinhang Li, Zhaopeng Qiu, Xiangyu Zhao, Zihao Wang, Yong Zhang, Chunxiao Xing, and Xian Wu. 2022. Gromov-Wasserstein Guided Representation Learning for Cross-Domain Recommendation. In *Proceedings of the 31st ACM International Conference on Information& Knowledge Management* (Atlanta, GA, USA) *(CIKM

'22). Association for Computing Machinery, New York, NY, USA, 1199–1208. https://doi.org/10.1145/3511808.3557338

[13] Jianxun Lian, Fuzheng Zhang, Min Hou, Hongwei Wang, Xing Xie, and Guangzhong Sun. 2017. Practical Lessons for Job Recommendations in the Cold-Start Scenario. In *Proceedings of the Recommender Systems Challenge 2017* (Como, Italy) *(RecSys Challenge '17)*. Association for Computing Machinery, New York, NY, USA, Article 4, 6 pages. https://doi.org/10.1145/3124791.3124794

[14] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling Task Relationships in Multi-Task Learning with Multi-Gate Mixture-of-Experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (London, United Kingdom) *(KDD '18)*. Association for Computing Machinery, New York, NY, USA, 1930–1939. https://doi.org/10.1145/3219819.3220007

[15] Xiao Ma, Liqin Zhao, Guan Huang, Zhi Wang, Zelin Hu, Xiaoqiang Zhu, and Kun Gai. 2018. Entire Space Multi-Task Model: An Effective Approach for Estimating Post-Click Conversion Rate. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 1137–1140. https://doi.org/10.1145/3209978.3210104

[16] Saskia Nuñez von Voigt, Erik Daniel, and Florian Tschorsch. 2021. Self-Determined Reciprocal Recommender Systemwith Strong Privacy Guarantees. In *Proceedings of the 16th International Conference on Availability, Reliability and Security* (Vienna, Austria) *(ARES 21)*. Association for Computing Machinery, New York, NY, USA, Article 31, 6 pages. https://doi.org/10.1145/3465481.3465769

[17] Iván Palomares, Carlos Porcel, Luiz Pizzato, Ido Guy, and Enrique Herrera-Viedma. 2021. Reciprocal Recommender Systems: Analysis of state-of-art literature, challenges and opportunities towards social recommendation. *Information Fusion* 69 (2021), 103–127. https://doi.org/10.1016/j.inffus.2020.12.001

[18] 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 *The 41st International ACM SIGIR Conference on Research& Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 25–34. https://doi.org/10.1145/3209978.3210025

[19] Chuan Qin, Hengshu Zhu, Tong Xu, Chen Zhu, Chao Ma, Enhong Chen, and Hui Xiong. 2020. An Enhanced Neural Network Approach to Person-Job Fit in Talent Recruitment. *ACM Trans. Inf. Syst.* 38, 2, Article 15 (feb 2020), 33 pages. https://doi.org/10.1145/3376927

[20] Chuan Qin, Hengshu Zhu, Chen Zhu, Tong Xu, Fuzhen Zhuang, Chao Ma, Jingshuai Zhang, and Hui Xiong. 2019. DuerQuiz: A Personalized Question Recommender System for Intelligent Job Interview. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2165–2173. https://doi.org/10.1145/3292500.3330706

[21] Zhaopeng Qiu, Xian Wu, Jingyue Gao, and Wei Fan. 2021. U-BERT: Pre-training user representations for improved recommendation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4320–4327. https://doi.org/10.1609/aaai.v35i5.16557

[22] Yanru Qu, Han Cai, Kan Ren, Weinan Zhang, Yong Yu, Ying Wen, and Jun Wang. 2016. Product-Based Neural Networks for User Response Prediction. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*. 1149–1154. https://doi.org/10.1109/ICDM.2016.0151

[23] Steffen Rendle. 2010. Factorization Machines. In *2010 IEEE International Conference on Data Mining*. 995–1000. https://doi.org/10.1109/ICDM.2010.127

[24] Michael Reusens, Wilfried Lemahieu, Bart Baesens, and Luc Sels. 2018. Evaluating Recommendation and Search in the Labor Market. *Know.-Based Syst.* 152, C (jul 2018), 62–69. https://doi.org/10.1016/j.knosys.2018.04.007

[25] Sebastian Ruder. 2017. An Overview of Multi-Task Learning in Deep Neural Networks. *CoRR* abs/1706.05098 (2017). arXiv:1706.05098 http://arxiv.org/abs/1706.05098

[26] Dazhong Shen, Chuan Qin, Hengshu Zhu, Tong Xu, Enhong Chen, and Hui Xiong. 2021. Joint Representation Learning with Relation-Enhanced Topic Models for Intelligent Job Interview Assessment. *ACM Trans. Inf. Syst.* 40, 1, Article 15 (sep 2021), 36 pages. https://doi.org/10.1145/3469654

[27] Zheng Siting, Hong Wenxing, Zhang Ning, and Yang Fan. 2012. Job recommender systems: A survey. In *2012 7th International Conference on Computer Science & Education (ICCSE)*. 920–924. https://doi.org/10.1109/ICCSE.2012.6295216

[28] Yi Su, Magd Bayoumi, and Thorsten Joachims. 2022. Optimizing Rankings for Recommendation in Matching Markets. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 328–338. https://doi.org/10.1145/3485447.3511961

[29] Ying Sun, Fuzhen Zhuang, Hengshu Zhu, Xin Song, Qing He, and Hui Xiong. 2023. Modeling the Impact of Person-Organization Fit on Talent Management With Structure-Aware Attentive Neural Networks. *IEEE Transactions on Knowledge and Data Engineering* 35, 3 (March 2023), 2809–2822. https://doi.org/10.1109/TKDE.2021.3115620

[30] Hongyan Tang, Junning Liu, Ming Zhao, and Xudong Gong. 2020. Progressive Layered Extraction (PLE): A Novel Multi-Task Learning (MTL) Model for Personalized Recommendations. In *Proceedings of the 14th ACM Conference on Recommender Systems* (Virtual Event, Brazil) *(RecSys '20)*. Association for Computing Machinery, New York, NY, USA, 269–278. https://doi.org/10.1145/3383313.3412236

[31] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9, 86 (2008), 2579–2605. http://jmlr.org/papers/v9/vandermaaten08a.html

[32] Chao Wang, Hengshu Zhu, Peng Wang, Chen Zhu, Xi Zhang, Enhong Chen, and Hui Xiong. 2021. Personalized and Explainable Employee Training Course Recommendations: A Bayesian Variational Approach. *ACM Trans. Inf. Syst.* 40, 4, Article 70 (dec 2021), 32 pages. https://doi.org/10.1145/3490476

[33] Ruoxi Wang, Bin Fu, Gang Fu, and Mingliang Wang. 2017. Deep & Cross Network for Ad Click Predictions. In *Proceedings of the ADKDD'17* (Halifax, NS, Canada) *(ADKDD'17)*. Association for Computing Machinery, New York, NY, USA, Article 12, 7 pages. https://doi.org/10.1145/3124749.3124754

[34] Hong Wenxing, Chen Yiwei, Qi Jianwei, and Hui Yin. 2015. iHR+: A mobile reciprocal job recommender system. In *2015 10th International Conference on Computer Science & Education (ICCSE)*. 492–495. https://doi.org/10.1109/ICCSE.2015.7250296

[35] Hong-Jian Xue, Xin-Yu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. 2017. Deep Matrix Factorization Models for Recommender Systems. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence* (Melbourne, Australia) *(IJCAI'17)*. AAAI Press, 3203–3209.

[36] Chen Yang, Yupeng Hou, Yang Song, Tao Zhang, Ji-Rong Wen, and Wayne Xin Zhao. 2022. Modeling Two-Way Selection Preference for Person-Job Fit. In *Proceedings of the 16th ACM Conference on Recommender Systems* (Seattle, WA, USA) *(RecSys '22)*. Association for Computing Machinery, New York, NY, USA, 102–112. https://doi.org/10.1145/3523227.3546752

[37] K. Yao, J. Zhang, C. Qin, P. Wang, H. Zhu, and H. Xiong. 2022. Knowledge Enhanced Person-Job Fit for Talent Recruitment. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE Computer Society, Los Alamitos, CA, USA, 3467–3480. https://doi.org/10.1109/ICDE53745.2022.00325

[38] Hongtao Yu, Chaoran Liu, and Fuzhi Zhang. 2011. Reciprocal recommendation algorithm for the field of recruitment. *Journal of Information & Computational Science* 8, 16 (2011), 4061–4068.

[39] Ezgi Yıldırım, Payam Azad, and Şule Gündüz Öğüdücü. 2021. biDeepFM: A multi-objective deep factorization machine for reciprocal recommendation. *Engineering Science and Technology, an International Journal* 24, 6 (2021), 1467–1477. https://doi.org/10.1016/j.jestch.2021.03.010

[40] Xiaoxue Zang, Toshihiko Yamasaki, Kiyoharu Aizawa, Tetsuhiro Nakamoto, Eitaro Kuwabara, Shinichi Egami, and Yusuke Fuchida. 2017. You Will Succeed or Not? Matching Prediction in a Marriage Consulting Service. In *2017 IEEE Third International Conference on Multimedia Big Data (BigMM)*. 109–116. https://doi.org/10.1109/BigMM.2017.11

[41] Zhi Zheng, Zhaopeng Qiu, Hui Xiong, Xian Wu, Tong Xu, Enhong Chen, and Xiangyu Zhao. 2022. DDR: Dialogue Based Doctor Recommendation for Online Medical Service. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) *(KDD '22)*. Association for Computing Machinery, New York, NY, USA, 4592–4600. https://doi.org/10.1145/3534678.3539201

[42] Zhi Zheng, Zhaopeng Qiu, Tong Xu, Xian Wu, Xiangyu Zhao, Enhong Chen, and Hui Xiong. 2022. CBR: Context Bias Aware Recommendation for Debiasing User Modeling and Click Prediction. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 2268–2276. https://doi.org/10.1145/3485447.3512099

[43] Zhi Zheng, Chao Wang, Tong Xu, Dazhong Shen, Penggang Qin, Baoxing Huai, Tongzhu Liu, and Enhong Chen. 2021. Drug Package Recommendation via Interaction-Aware Graph Induction. In *Proceedings of the Web Conference 2021* (Ljubljana, Slovenia) *(WWW '21)*. Association for Computing Machinery, New York, NY, USA, 1284–1295. https://doi.org/10.1145/3442381.3449962

[44] Zhi Zheng, Chao Wang, Tong Xu, Dazhong Shen, Penggang Qin, Xiangyu Zhao, Baoxing Huai, Xian Wu, and Enhong Chen. 2023. Interaction-Aware Drug Package Recommendation via Policy Gradient. *ACM Trans. Inf. Syst.* 41, 1, Article 3 (jan 2023), 32 pages. https://doi.org/10.1145/3511020

[45] 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 Trans. Manage. Inf. Syst.* 9, 3, Article 12 (sep 2018), 17 pages. https://doi.org/10.1145/3234465

[46] Gözde Özcan and Sule Gündüz Ögüdücü. 2016. Applying different classification techniques in reciprocal job recommender system for considering job candidate preferences. In *2016 11th International Conference for Internet Technology and Secured Transactions (ICITST)*. 235–240. https://doi.org/10.1109/ICITST.2016.7856703