Joint Representation Learning with Relation-Enhanced Topic Models for Intelligent Job Interview Assessment

DAZHONG SHEN, School of Computer Science, University of Science and Technology of China
CHUAN QIN and HENGSHU ZHU, Baidu Talent Intelligence Center, Baidu Inc.
TONG XU and ENHONG CHEN, School of Computer Science, University of Science and Technology of China
HUI XIONG, Rutgers, the State University of New Jersey

The job interview is considered as one of the most essential tasks in talent recruitment, which forms a bridge between candidates and employers in fitting the right person for the right job. While substantial efforts have been made on improving the job interview process, it is inevitable to have biased or inconsistent interview assessment due to the subjective nature of the traditional interview process. To this end, in this article, we propose three novel approaches to intelligent job interview by learning the large-scale real-world interview data. Specifically, we first develop a preliminary model, named Joint Learning Model on Interview Assessment (JLMIA), to mine the relationship among job description, candidate resume, and interview assessment. Then, we further design an enhanced model, named Neural-JLMIA, to improve the representative capability by applying neural variance inference. Last, we propose to refine JLMIA with Refined-JLMIA (R-JLMIA) by modeling individual characteristics for each collection, i.e., disentangling the core competences from resume and capturing the evolution of the semantic topics over different interview rounds. As a result, our approaches can effectively learn the representative perspectives of different job interview processes from the successful job interview records in history. In addition, we exploit our approaches for two real-world applications, i.e., person-job fit and skill recommendation for interview assessment. Extensive experiments conducted on real-world data clearly validate the effectiveness of our models, which can lead to substantially less bias in job interviews and provide an interpretable understanding of job interview assessment.

CCS Concepts: • **Information systems → Data mining;**

Additional Key Words and Phrases: Interview assessment, latent variable model, neural topic model, representation disentanglement, sequential data

This work was done when Dazhong Shen was an intern at Talent Intelligent Center, Baidu Inc. supervised by the second and third authors.

This is a substantially extended and revised version of [55], which appears in the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI'18).

This work was partially supported by grants from the National Natural Science Foundation of China (Grant No.91746301, 71531001, 61836013, U20A20229, and 62072423).

Authors' addresses: D. Shen, School of Computer Science, University of Science and Technology of China, Hefei, Anhui, 230027, China; email: sdz@mail.ustc.edu.cn; C. Qin and H. Zhu (corresponding author), Baidu Talent Intelligence Center, Baidu Inc. Beijing, 100085, China; emails: chuanqin0426@gmail.com, zhuhengshu@baidu.com; T. Xu and E. Chen, School of Computer Science, University of Science and Technology of China, Hefei, Anhui, 230027, China; emails: {tongxu, cheneh}@ustc.edu.cn; H. Xiong (corresponding author), Rutgers, the State University of New Jersey, Newark, NJ 07102, United States; email: hxiong@rutgers.edu.

1046-8188/2021/08-ART15 $15.00

https://doi.org/10.1145/3469654

ACM Reference format:
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 (August 2021), 36 pages.
https://doi.org/10.1145/3469654

1 INTRODUCTION

As one of the most important functions in human resource management, talent recruitment aims at acquiring the right talents for organizations and always has a direct impact on business success. As indicated in an article from Forbes, US corporations spend nearly 72 billion dollars each year on a variety of recruiting services, and the worldwide amount is likely three times bigger [4]. In particular, job interview, usually consisting of the pre-screening stage followed by multiple in-person interview rounds, is considered as one of the most useful tools and the final testing ground for evaluating potential employees in the hiring process. It has attracted more and more attention in human resource management [10, 25, 62]. While substantial efforts have been made on the improvement of the job interview process, the traditional interview process has a substantial risk of bias due to the subjective nature of the process. Sometimes, due to the mismatching between interviewers and candidates, several interviewers, such as HR recruiters and junior interviewers, may not have the sufficient domain knowledge to find crucial directions for assessing the candidate's abilities. As a result, their decisions in the pre-screening stage or initial interview may be biased. Meanwhile, different interviewers may have different technical backgrounds or different experience levels in personal qualities, which may cause inconsistent evaluation criteria and partial or biased assessment reports for in-person interviews.

Recently, the **Artificial Intelligence** (**AI**) trend has made its way to talent recruitment, such as job recommendation [37, 43, 45, 46, 74, 77], talent flow [12, 31, 38, 67], and market trend analysis [68, 76]. Among them, some efforts have also been made on enhancing the quality and experience of job interview [47, 56]. They usually aim at recommending discriminative interview questions based on the job description or resume to test the candidate's qualification. However, none of them has involved the interview assessment collected from the experienced interviewers, which contains the crucial focus and perspectives investigated in the real-world interviews. Along this line, a critical challenge is how to reveal the latent relationships between job position and candidate, and further form perspectives for effective interview assessment. Intuitively, experienced interviewers could discover the topic-level correlation between job description and resume, and then design interview details to measure the suitability of applicants. As a motivation example shown in Figure 1, a candidate for "image algorithm engineer", who has rich experience in image processing based on machine learning, should be interviewed with questions not only about "programming", "C++", but also specific machine learning approaches, such as "**Convolutional Neural Networks** (**CNN**)", based on their historical experiences. Thus, there often exists a strong correlation among job descriptions, resumes and interview assessments.

Meanwhile, in order to capture the effective representation of all three collections, respectively, another essential challenge is how to model individual characteristics for each collection. To be specific, first, job descriptions are usually more abstract than resumes and interview assessments, and candidates with different backgrounds may be suitable for the same job. Thus, there exist distant diversity levels between job descriptions with resumes and interview assessments. Second, not all content in a resume is equally important to demonstrate the candidate's competences. For example, some individual capacities, that may not link to the job very well, catch less attention from recruiters and interviewers, like the second experience in Figure 1. Therefore, it would be more

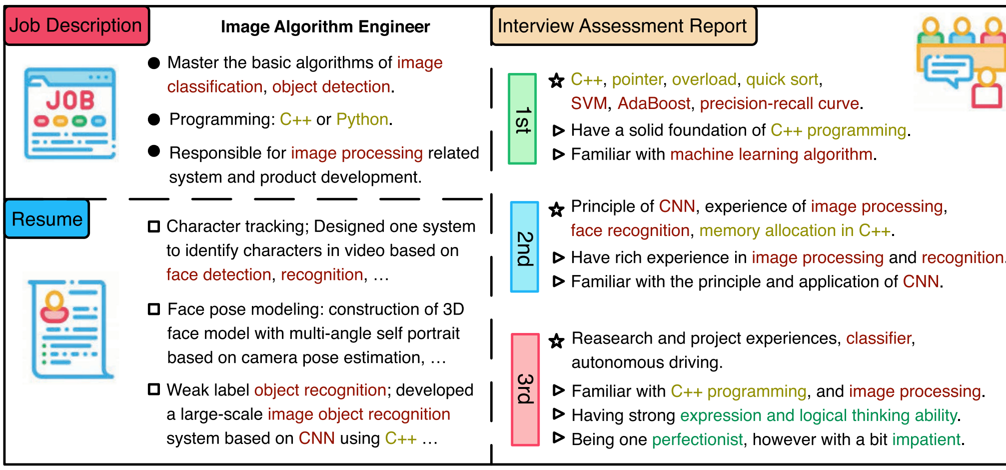


Fig. 1. A motivating example. Due to the limited space, we have condensed essential context without impacting meaning for all three types of texts. In particular, in the interview assessment report, ☆ and ▹ represents the skill keywords investigated and assessment reported by interviewers in each interview round, respectively.

reasonable to disentangle the candidate's core competences related to this specific job with other personal capacities. Third, in order to assess the fitness between job posting and candidates thoroughly and in-depth, the goal and focus of evaluation often vary over multiple interview rounds, meanwhile. Generally, a three-round interview commonly consists of a basic aptitude investigation round, core technical estimation round, and another evaluation round from a comprehensive and high-level perspective in turn. Take the same candidate in Figure 1 as an example. In the first interview round, the basic programming and machine learning skills and concepts have been investigated, such as "C++", "quick sort", "**Support Vector Machine (SVM)**", and "precision-recall". Then, detailed technology and algorithm related to job requirements and candidates' experiences were estimated deeply in the second round with keywords "**Convolutional Neural Network (CNN)**", "image processing" , "face recognition", and "memory allocation". Last, the third round aimed at evaluating candidates from a holistic perspective, involving more explorations on historical projects, even personality and emotion. Thus, how to model the evolution of the semantic information over different interview rounds becomes another problem.

In this article, we propose three approaches (i.e., **Joint Learning Model on Interview Assesment (JLMIA)**, Neural-JLMIA, and **Refined-JLMIA** (**R-JLMIA**)) to address the above challenges. The core goals of our approaches are to mine semantic topic spaces in the job description, resume, and interview assessment, respectively, learn their effective representations, and further reveal the relationships among them. As the overview shown in Figure 2, we design our approaches based on topic models, which are well-known for the effective representation with high interpretability to explain the hidden decision logic [11]. With the learned topic distribution for each document as the representation, a variety of applications can be enabled for effective and efficient real-world job interviews. Here, we develop two real-world applications, i.e, person-job fit and skill recommendation for interview assessment. They aim at assisting recruiters or junior interviewers in measuring the matching degree between the job and candidate, and helps interviewers design objective and effective interview procedures, respectively.

To be specific, first, we aim at directly achieving the primary goal of modeling the relationship among job descriptions, candidate resumes, and interview assessments. To this end, we adopted

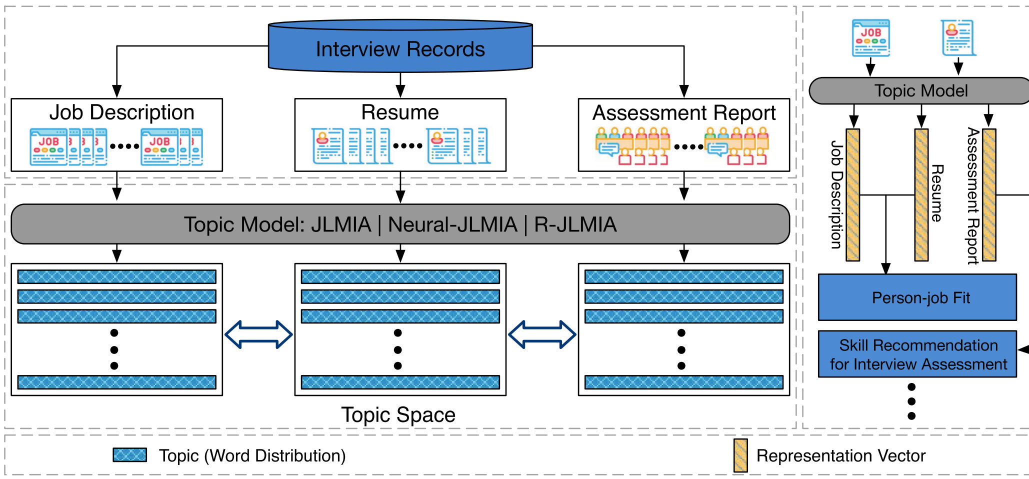


Fig. 2. An overview of our approaches and applications. As shown in the left side, with all three collections as the input, our approaches can output their topic spaces, respectively, where a semantic topic is represented as a word distribution on the vocabulary. Each document in each collection can be represented effectively as the distribution on the corresponding topic space, i.e., topic distribution. As shown in the right side, given the pair of a job description and a resume as the input, the application of person-job fit outputs the matching degree between job and candidate. The application of skill recommendation for interview assessment outputs the ranked keywords that may be investigated in the following interview round.

an approach proposed in our own prior work [55] called JLMIA. This approach assumes that resumes and interview assessments can be modeled using the same topic distribution, which is generated from job descriptions that the specific candidate applied to. Then, the corresponding variational inference approach is designed to infer latent variables. As a result, JLMIA has achieved great performance in representative learning perspectives of different job interview processes from the historically successful job interview records. However, variational inference used in JLMIA requires arduous mathematical derivation and lacks the flexibility of adapting to more expressive conditional information and dependence assumption. Fortunately, **Neural Topic Models (NTMs)** [39, 58] have recently been proposed to infer the semantic topic space with the neural network, which is flexible and expressive. Inspired by this, we first build a model, named Neural-JLMIA, equipped with the same dependence assumption as JLMIA but a neural variational inference similar to NMTs. Neural-JLMIA demonstrates consistent, even more comparative, performance on document modeling and real-world applications with JLMIA.

Furthermore, we propose to refine JLMIA with R-JLMIA by modeling individual characteristics for each collection. To be specific, our R-JLMIA consists of three components. First, we utilize the NTM model to distill the latent ability-aware job representation from the job description. Second, a novel NTM, named **Competence Dismantled NTM (CDNTM)**, is proposed to disentangle the core competences. That is, we define a competence-related latent variable tied with job description, and apply another secondary variable to model other personal capacities without conditional information. Third, we turn to capture the evolution of the semantic topics over different interview rounds, and propose **Sequential Assessment NTM (SANTM)**. It applies a **Long Short-Term Memory (LSTM)** network on the latent variables of different interview rounds, with the latent representations of job and resume as the conditional information. In addition, we provide solutions for two real-world applications based on our approaches, namely person-job fit and skill recommendation for interview assessment. Finally, extensive experiments conducted on large-scale real-world data clearly validate the effectiveness of our models, which can lead to substantially less bias in job interviews and provide a valuable understanding of job interview assessment.

We summarize the contributions of the article as follows:

— To our best knowledge, we are the first to provide an interpretable understanding of job interview assessment by jointly model job description, resume, and interview assessment.
— We propose three novel approaches, namely JLMIA, Neural-JLMIA, and R-JLMIA, to model the relationship among job descriptions, resumes, and interview assessments. In particular, our R-JLMIA can further capture the individual characteristics of each collection effectively, such as disentangling the core competences in resumes and learning the evolution of semantic topics in each interview round.
— Two real-world applications, namely person-job fit and skill recommendation for interview assessment are developed based on our approaches.
— Extensive experiments and discussions conducted on large-scale real-world data clearly validate the effectiveness of our models, whether in terms of text modeling or performance on two applications.

Overview. The rest of this article is organized as follows: in Section 2, we briefly introduce some related literature of our study; in Section 3, we introduce the preliminaries and formally define our problem; JLMIA model will be introduced in Section 4; then, we extended JLMIA with Neural-JLMIA in Section 5; our novel R-JLMIA model is described in Section 6; in addition, we would introduce two important applications based on R-JLMIA, i.e., person-job fit and skill recommendation for interview assessment, in Section 7; the performance of our models would be evaluated in Section 8, with some further discussions and case studies; finally, we conclude this article in Section 9.

2 RELATED WORK

In this section, we will briefly review the relevant literatures of our study, which can be grouped into two categories: Recruitment Analysis and Text Mining with Topic Model.

Recruitment Analysis. With the importance of talents at an all time high and the availability of recruitment big data, recruitment analysis has been attracting more and more attentions [47, 61]. One of the most striking topics is named Person-job Fit [53], which aims at measuring the degree of fitness between job and candidates and recommend talents for suitable positions. Early methods include treating it as a job/candidate recommendation problem, which can be dated back to Malinowski et al. [37] in 2006, where the authors tried to find a good match between talents and jobs by two distinct recommendation systems. Then, inspired by the idea of recommendation system, Lee and Brusilovsky [28] induced a comprehensive job recommendation system equipped with four recommendation methods for meeting diversified needs form job seekers. Furthermore, Paparrizos et al. [43] exploited all historical job transitions as well as the data associated with employees and institutions to predict the next job transition of employees. Also, Diaby et al. [15] combined users' profiles and social network-based data to propose jobs for Fackbook and LinkdIn users. Hong et al. [22] enhanced the recommendation performance by extending users' profile dynamically by job application records and their behaviors. Recently, the deep learning and text mining techniques has been applied to automatically learn representations for job and candidates. For instances, Zhu et al. [77] developed a bipartite CNNs to effectively learn the joint representation of applicant profile and job requirement for matching talents to suitable jobs. Qin et al. [45, 46] proposed a word-level semantic representation model for both job requirements and job seekers' experiences based on **Recurrent Neural Network (RNN)** and hierarchical attention mechanism. And, the model proposed by Bian et al. [5] can capture the global semantic interactions between two sentences from a job posting and a candidate resume, respectively. Different from those works, which need additional supervised information to enhance performance, we aims at exploring the latent relationship among job, resume, and interview assessment in an unsupervised manner, which can also provide solutions for person-job fit with interpretable representation.

Besides the match of talents and jobs [49], researchers are also devoted to analyzing recruitment market from more novel perspective. For example, in term of market trends analysis, Zhu et al. [76] perceived recruitment market trends by leveraging unsupervised latent variable model. Li et al. [33, 34] developed company profiles and predict salaries from an employee's perspective based on collaborative topic regression. Xu et al. [68] attempted to model the popularity of job skills that help job seekers to design individual job career with suitable skills. Sun et al. [60] modeled the market-oriented job skill valuation by applying cooperative composition neural networks. Furthermore, they proposed to recommend job skills for job seekers by reinforcement learning [59]. As for the talent flow analysis, Cheng et al. [12] constructed an inter-company job-hopping network from various social media sources to model job transition activities and rank influential companies. Zhang et al. [71, 72] modeled the large-scale talent flow with factor models or graph embedding technologies. Furthermore, Meng et al. [38] exploited a hierarchical neural network structure with embedded attention mechanism for characterizing both the internal and external job mobility. In Xu et al. [67] , authors designed a talent circle detection model and created the job transition network that benefit organizations for hiring the right talent from the right source. Zhang et al. [73] utilized attention networks to forecast talent demand of companies. Wang et al. [65] applied deep collaborative models to model employees' career trajectory.

Recently, some studies have also been developed to enhance the quality and experience of job interview. Shi et al. [56] developed one two-stage deep learning model aiming to automatically generate screening questions for a given job description. Qin et al. [47] proposed a personalized question recommender system to create in-depth skill assessment questions based on job posting and resume dataset. However, most of them did not considered the interview assessments collected from experienced interviewers, which contains the important clues and focus for candidate evaluation during the real-world interviews, especially for the face-to-face interviews. Therefore, in this article, we proposed a novel approach for intelligent job interview assessment by jointly modelling job description, resume, and interview assessment from large-scale real-world interview data.

Text Mining with Topic Model. Probabilistic topic models are capable of grouping semantic coherent words into human interpretable topics. As an important member of archetypal topic models, **Latent Dirichlet Allocation** (**LDA**) [9] has a lot of extensions [30, 51, 57, 69] with various applications [13, 23, 29, 75]. Among them, some works focus on modeling multiple categories of documents and exploring the relations among them. In some cases, researchers assumed that there exist shared latent topic distribution among document collections with high correlation. For example, Mimno et al. [41] designed a polylingual topic model that discovers topics aligned across multiple languages. Pyo et al. [44] proposed a novel model to learn the shared topic distribution between users and TV programs for TV program recommendation. In some other cases, researchers attempted to model time-evolving statistical properties of sequential document datasets. For instance, In order to model the evolution of both topic's word distribution and popularity over time, Blei et al. [8] proposed the **Dynamic topic model** (**DTM**), which depicts the topic transformation with the Gaussian distribution. Further, iDTM [1], which is one variant of DTM, can accommodate the evolution of topic number by leveraging the **recurrent Chinese restaurant franchise** (**RCRF**) process. Moreover, some other models focus on explaining how the topic popularity change within the fixed semantic topic space. Wang et al. [66] presented the topics over time model to captures topic popularity over time via a beta distribution. SeqLDA [17] and AdaTM [16] utilizes Poisson-Dirichlet Process to model how topics evolve among segments in a document.

Indeed, in the last decade, the population of probabilistic topic model mostly benefited from efficient inference by utilizing exploiting conjugacy with variational techniques, especially mean field methods, and Markov chain Monte Carlo [2, 3, 24]. However, there exist several inevitable drawbacks in both methods that prevent the free exploration to the space of different modeling assumptions. On one hand, the inference methods are inevitably required to be re-derived even if there are just minor changes, which can be mathematically arduous and time consuming [58]. On the other hand, as topic models have grown more expressive, in order to capture topic dependencies or exploit conditional information, inference methods have become increasingly complex, especially for non-conjugate models [64]. Recently, several effort has been made to solve those problems [27, 42, 48]. Among them, the **variational auto-encoders** (**VAEs**) [26, 50, 54] are neural options for inferencing topic model effectively. It applies a variational distribution parameterised by a neural network to approximate the posterior of topic model. Along this line, NTM [39, 58] was designed to learn topic's word distribution and latent topic distribution with neural inference network. Meanwhile, the NTM model is equipped with high flexibility which can be easy to tailored in term of specific tasks without arduous mathematical inference. Therefore, several variants and applications have been developed in multiple fields in the past years [35, 70].

Different from existing research efforts, in this article, we aim at developing effective topic models to jointly model both the correlation among job description, candidate resume and interview assessment, and the individual characteristic of each collection.

3 PROBLEM STATEMENT

In this article, we aim to jointly model job description, resume, and interview assessment from the representative perspectives. Meanwhile, we should take account of both the relationship among three collections and the individual characteristic for each of them. For facilitating illustration, some important mathematical notations used in this article are listed in Table 1. Then, the preliminary discussions about our data and the formal definition of our problem can be found in the following.

Formally, our dataset contains the recruitment documents of M unique applications, i.e., $S = \{S_m = (J_m, R_m, A_m)\}_{m=1}^{M}$, where R_m and J_m are the resume of the mth candidate and the job description that she applied, and A_m is the interview assessment tuple for this candidate. Specifically, $A_m = [A_{m,1}, A_{m,2}, \ldots, A_{m,D}]$ contains D interview rounds, generally $D = 3$. As the example shown in Figure 1, job description J_m contains detailed job requirements. Resume R_m mainly consists of the past work and project experiences of these candidates. Meanwhile, the assessment report $A_{m,d}$ records the evaluation of the candidate's competences. According to the different goals of interviews, each record $A_{m,d}$ may evaluate the candidate in different aspects and depths. Since all of the job descriptions, resumes, and interview assessments are textual data, we use bag-of-words to represent them, e.g., $J_m = \{w_{m,n}^J\}_{n=1}^{N_m^J}$, similar to R_m and $A_{m,d}$. In particular, we only consider the skill keywords or other ability-related words, such as education-related keywords, as the candidate word bag to reduce unnecessary input noise. Along this line, we can formally describe our problem of modeling the relationship and individual characteristic of the job description, resume, and interview assessment as follows:

Problem 1. Given the recruitment document of M applications $S_m = \{(J_m, R_m, A_m)\}_{m=1}^{M}$, our goal is two-fold: (G1) discover the strong relationships among job description J_m, resumes R_m, and interview assessments A_m; (G2) model the individual characteristic or structure for each collection, i.e., (G2.1) model the distant diversity level between job description J_m and other two collections (R_m, A_m); (G2.2) disentangle the representations of core competences and secondary abilities in resume R_m; (G2.3) explore the evolution of semantic topics over interview rounds $\{A_{md}\}_{d=1}^{D_m}$.

Table 1. Mathematical Notations

Symbols	Descriptions
J_m, R_m, A_m, (A_{md})	The job description that the mth candidate applied, their resume, and the corresponding assessment report (in the dth interview round).
K^J, K^R, K^A	The topic number in the job description, resume and assessment report.
V^J, V^R, V^A	The vocabulary in job description, resume and interview assessment.
φ^J, φ^R, φ^A	The topic sets in job description, resume and interview assessment.
θ^J_m, θ^R_m, θ^A_m, (θ^A_{md})	The topic distributions of job description, resume and assessment report (in the dth interview round) for the mth candidate.
θ^I_m	The topic distribution shared by the resume and assessment report for the mth candidate.
e^J_m, e^R_m, e^A_m, (e^A_{md})	The embedding for job description, resume and assessment report (in the dth interview round) for the mth candidate.
$w^J_{m,n}$, $w^R_{m,n}$, $w^A_{m,n}$, $(w^A_{md,n})$	The nth word of job description, resume and assessment report (in the dth interview round) for the mth candidate.
z^J_m, $z^R_{c,m}$, $z^R_{s,m}$, z^A_{md}	The latent variable for job description that the mth candidate applied, the core competence variable and secondary variable for their resume, and the latent variable for assessment report in the dth interview round.
C	The hyper-parameter of the quotient of K^A or K^R divided by K^J in JLMIA.
$c^J_{m',n}$, $c^R_{m'_i,n}$, $c^A_{m'_i,n}$	The topic assignments of the nth word of the m'th job description, the resume and assessment report for the m'_ith candidate that apply this job in JLMIA model.
$h^A_{m,i}$	The hidden state of the LSTM layer for the dth interview round of the mth candidate in R-JLMIA.
t^J, t^R, t^A	The topic-specific vectors for job description, resume and interview report.
v^J, v^R, v^A	The word embedding vectors for job description, resume and interview report.
α	The parameter of the Dirichlet prior for $\theta^J_{m'}$ in JLMIA.
δ	The variance parameter of the Gaussian prior for $\theta^I_{m'_i}$ in JLMIA.
β^J, β^R, β^A	The parameters of Dirichlet prior for φ^J, φ^R, and φ^A in JLMIA.
μ^*_*, δ^*_*	The mean and variance parameters of the prior and posterior of the Gaussin latent variables in Neural-JLMIA and R-JLMIA.
ζ	The variational parameters to preserve the lower bound of ELBO.
$f_*(\cdot)$	The full connect network layer.
$g(\cdot)$	The non-liner active function.
M	The number of unique applications.

4 TECHNICAL DETAILS OF JLMIA

In this section ,we first describe model formulation of JLMIA that can address the goal (G1) and (G2.1) in some degrees. Then, the variational inference algorithm will be introduced to infer latent variables.

4.1 Model Formulation

To model the latent semantics in the job description, resume, and interview assessment, we assume there exist latent topics, represented by φ^J, φ^R, and φ^A, in all of them. Each topic φ^*_i (i.e., φ^J_i, φ^R_i, or φ^A_i) is represented by a word distribution over the corresponding vocabulary. And, our tasks are further transformed to model the relationships among these latent topics. On one hand, to model the strong correlation among job description J_m, resume R_m, and interview assessment A_m,

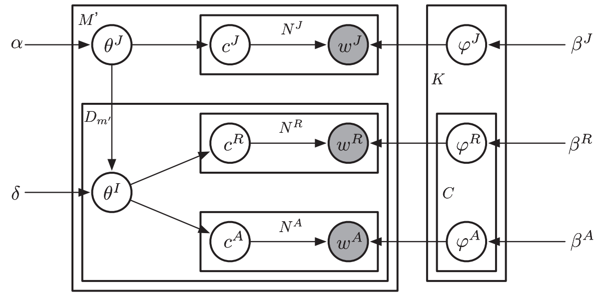


Fig. 3. The graph representation for JLMIA model.

ALGORITHM 1: The Generative Process of JLMIA for Resume and Interview Assessment

(1) For each topic k of candidate interview record:
  (a) Draw φ_k^R from the Dirichlet prior $Dir(\beta^R)$.
  (b) Draw φ_k^A from the Dirichlet prior $Dir(\beta^A)$.
(2) For each job description $J_{m'}$:
  (a) Sample topic distribution $\theta_{m'}^J \sim Dir(\alpha)$.
(3) For each document pair $(R_{m'_i}, A_{m'_i})$ collected from candidates applied to $J_{m'}$:
  (a) Sample topic distribution $\theta_{m'_i}^I \sim N(h(\theta_{m'}^J, C), \delta^2)$
  (b) For the r-th word $w_{m'_i,r}^R$ in resume $R_{m'_i}$:
    (i) Draw topic assignment $c_{m'_i,r}^R \sim Multi(\pi(\theta_{m'_i}^I))$.
    (ii) Draw word $w_{m'_i,r}^R \sim Multi(\varphi^R_{c^R_{m'_i r}})$.
  (c) For the e-th word $w_{m'_i,e}^A$ in interview assessment $A_{m'_i}$:
    (i) Draw topic assignment $c_{m'_i,e}^A \sim Multi(\pi(\theta_{m'_i}^I))$.
    (ii) Draw word $w_{m'_i,e}^A \sim Multi(\varphi^A_{z^A_{m'_i e}})$.

we directly assume R_m and A_m share the same pair-specific distribution θ_m^I over topics, while θ_m^I are generated from the logistic-normal distribution with mean parameter related to the topic distribution of job description θ_m^J. On the other hand, for revealing the differences between the diversity of those three collections, the topic numbers of φ^J, φ^R, and φ^A are set as $|K^A| = |K^R| = C \cdot |K^J| = CK$. In other words, for each topic in φ^J, there are C topics in φ^R (φ^A) related to it. The graphical model of JLMIA is shown in Figure 3. Since the generative process of job description is the same as LDA [9], here, we only list the generative process for resume and interview assessment $\{R_m, A_m\}_{m=1}^{|M|}$, shown in Algorithm 1. For convenience, we augment all applications to the same job $J_{m'}$ as $I_{m'} = \{R_{m'_i}, A_{m'_i}\}_{i=1}^{D_{m'}}$, where $m' = 1, 2, \ldots, M'$, and $\sum_{m'=1}^{M'} D_{m'} = M$. In particular, $h(\theta, C)$, in line $3.(a)$, is a vector concatenating C log vectors of θ, i.e., $h(\theta_m^J, C)_k = log\theta_{m,k'}^J$, $k' = k \; mod \; K$, $1 \leq k' \leq K$. $\pi(\theta)$, in line $3.(b).i$ and $3.(c).i$, is the logistic transformation, i.e., $\pi(\theta_{m'_i}^I)_k = \frac{exp\{\theta_{m'_i,k}^I\}}{\sum_{t=1}^{CK} exp\{\theta_{m'_i,t}^I\}}$.

Due to the non-conjugacy of the logistic normal and multinomial, the latent parameters posterior is intractable. Thus, we propose a variational inference algorithm for JLMIA.

4.2 Variational Inference for JLMIA

Here, we develop a variational inference algorithm for JLMIA based on mean-field variational families. The basic idea behind variational inference is to optimize the free parameters of a distribution over the latent variables, so that the distribution is close in **Kullback–Liebler** (**KL**) divergence to true posterior. In our model, let us denote all latent variable parameters by Φ and all hyperparameters by Ω. Following the generative process, the joint distribution can be factored as

$$p(S,\Phi|\Omega) = p(\Phi|\Omega)\prod_{m'=1}^{M'} P(J_{m'}, I_{m'}|\Phi), \tag{1}$$

where each component can be calculated by

$$\begin{aligned} p(J_{m'}, I_{m'}|\Phi) &= p\left(J_{m'}|z^J_{m'}, \varphi^J\right)\prod_{i=1}^{D_{m'}} p\left(R_{m'_i}|z^R_{m'_i}, \varphi^R\right) p\left(A_{m'_i}|z^A_{m'_i}, \varphi^A\right), \\ p(\Phi|\Omega) &= \prod_{m'=1}^{M'}\prod_{i=1}^{D_{m'}} p\left(\theta^A_{m'_i}|\theta^J_{m'}, \delta^2\right)\prod_{r=1}^{N^R_{m'_i}} p\left(c^R_{m'_i r}|\theta^I_{m'_i}\right)\prod_{e=1}^{N^A_{m'_i}} p\left(c^A_{m'_i e}|\theta^I_{m'_i}\right) \\ &\quad\times\prod_{m'=1}^{M'} p\left(\theta^J_{m'}|\alpha\right)\prod_{j=1}^{N^J_{m'}} p\left(c^J_{m'j}|\theta^J_{m'}\right)\times\prod_{k=1}^{K} p\left(\varphi^J_k|\beta^J\right)\prod_{k=1}^{CK} p\left(\varphi^R_k|\beta^R\right) p\left(\varphi^A_k|\beta^A\right). \end{aligned} \tag{2}$$

Then, corresponding to this joint distribution, we posit the fully factorized variational families as following, where the detailed description of each term can be found in the Appendix:

$$\begin{aligned} q(\Phi) = &\prod_{k=1}^{K} q(\varphi^J_k)\prod_{k=1}^{CK} q(\varphi^R_k) q(\varphi^A_k)\prod_{m'=1}^{M'} q(\theta^J_{m'})\prod_{j=1}^{N^J_{m'}} q(c^J_{m'j}) \\ &\times\prod_{m'=1}^{M'}\prod_{i=1}^{D_{m'}}\prod_{k=1}^{CK} q(\theta^I_{m'_i,k})\prod_{r=1}^{N^R_{m'_i}} q(c^R_{m'_i r})\prod_{e=1}^{N^A_{m'_i}} q(c^A_{m'_i e}). \end{aligned} \tag{3}$$

According to [7], minimizing the KL divergence between the variational distribution and true posterior is equivalent to maximize the log likelihood bound of job interview records, which is the **evidence lower bound** (**ELBO**):

$$\begin{aligned} log\ p(S|\Omega) &\geq E_q[log\ p(S,\Phi|\Omega)] + H(q) \\ &= E_q[log\ p(\Phi|\Omega)] + \sum_{m'=1}^{|M'|} E_q[log\ p(J_{m'}, I_{m'}|\Phi)] + H(q), \end{aligned} \tag{4}$$

where the expectation $E_q[\cdot]$ is taken with respect to the variational distribution in Equation (3), and $H(q)$ denotes the entropy of that distribution.

The largest challenge to maximize *ELBO* is the non-conjugacy of logistic normal and multinomial distributions, which leads to the difficulty in computing the excepted log probability of topic assignments in documents of each candidate interview records. Similar to [63], we introduce a new variational parameter $\zeta = \{\zeta_{m'_{1:|D_m|}}\}_{m'=1:|M|}$ to preserve the lower bound of *ELBO*. Here, we take the $E_q[logp(z^R_{m'_i r}|\theta^A_{m'_i})]$ as an example to explain it (the $E_q[logp(c^A_{m'_i e}|\theta^A_{m'_i})]$ can be computed in a similar way):

$$
\begin{aligned}
E_q\left[logp\left(c^R_{m'_i r}|\theta^I_{m'_i}\right)\right] &= E_q\left[\theta^I_{m'_i, c^R_{m'_i r}}\right] - E_q\left[log\left(\sum_{k=1}^{CK} exp\left\{\theta^I_{m'_i,k}\right\}\right)\right] \\
&\geq E_q\left[\theta^I_{m'_i, z^R_{m'_i r}}\right] - \zeta^{-1}_{m'_i}\left(\sum_{k=1}^{CK} E_q\left[exp\left\{\theta^I_{m'_i,k}\right\}\right]\right) + 1 - log(\zeta_{m'_i}).
\end{aligned}
\tag{5}
$$

For maximizing the *ELBO*, we develop an EM-style algorithm with a coordinate ascent approach to optimize parameters, the details of which can be found in the Appendix.

5 NEURAL VERSION OF JLMIA

It has been reported that the variational inference algorithm used in JLMIA suffers from arduous mathematical derivation and lacks the flexibility of adapting to more expressive conditional information and dependence assumption [58, 64]. Therefore, we turn to provide the NTM based version of JLMIA, namely Neural-JLMIA, which is equipped with the same assumptions as JLMIA but variational distributions parametrized by neural networks. To be specific, we will first provide the formal definition of semantic topic space in NTM. Then, the technical details of Neural-JLMIA will be introduced.

5.1 The Definition of Semantic Topic Space

Similar to the probability topic model, we assume that there exist topic sets, represented by φ^J, φ^R, and φ^A, in a job description, resume, and job interview assessment, with the size K^J, K^R, and K^A, respectively. Then, we following [39] and introduce two parameters for each semantic topic space, namely topic-specific vectors $t^J \in \mathbb{R}^{K^J \times H}$, $t^R \in \mathbb{R}^{K^R \times H}$, and $t^A \in \mathbb{R}^{K^A \times H}$, and word embedding vectors $v^J \in \mathcal{R}^{|V^J| \times H}$, $v^R \in \mathbb{R}^{|V^R| \times H}$, $v^A \in \mathbb{R}^{|V^A| \times H}$ to generate φ^J, φ^R, φ^A. Take φ^J as an example, we have

$$
\varphi^J = Softmax(t^J \cdot (v^J)^T)^T, \tag{6}
$$

where, H is the word embedding size for each word.

Note that v^* can be initialized by the result of some word embedding models such as Word2Vec [19], and Skip-gram model [40], after training on general datasets.

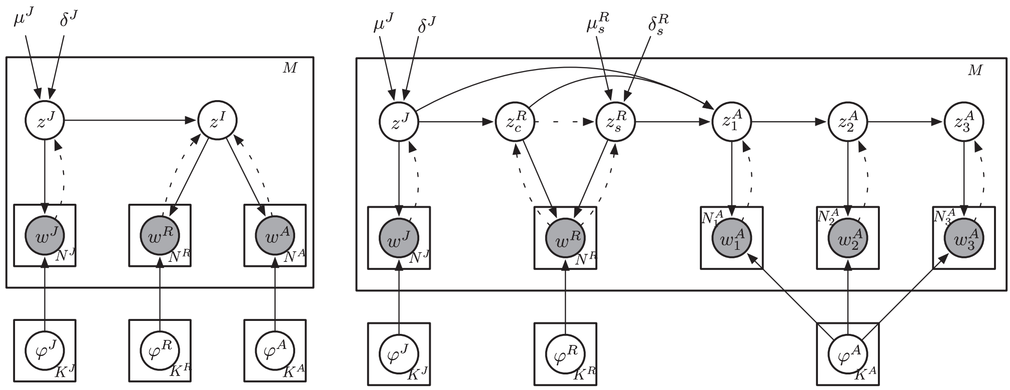
Furthermore, following the idea in LDA-based topic model, we assume that each job description J_m, resume R_m, and interview assessment A_m have topic distributions $\theta^J_m \in \mathbb{R}^{K^J}$, $\theta^R_m \in \mathbb{R}^{K^R}$, and $\theta^A_m \in \mathbb{R}^{K^A}$, over corresponding topic sets, respectively. Then, we can generate each word in those three collections similar as that in [39]. Take the generation process for job description J_m as the example, we can generate each word $w^J_{m,n} \in J_m$ by word distribution $\theta^J_m \cdot \varphi^J$. In other word, the generative possibility of given job descriptions J_m conditioning on θ^J and β^J can be computed by

$$
p(J_m|\theta^J_m, \beta^J) = \prod_{n=1}^{N^J_m} \theta^J_m \cdot \varphi^J_{*, w^J_{m,n}}. \tag{7}
$$

Along with those formal definitions above, we turn into infer the topic space φ of each document collection and the topic distribution θ of each document.

5.2 Neural-JLMIA

Same as JLMIA, there exists two main assumptions in Neural-JLMIA: (1) the resume and assessment share the same topic distribution $\theta^A = \theta^R = \theta^I$. θ^I can be produced from the share latent



(a) Neural-JLMIA (b) R-JLMIA

Fig. 4. The directed graphs for Neural-JLMIA and R-JLMIA model. We use solid lines to denote the generative model, and dashed lines to denote the variational approximation to the intractable posterior.

variable z^I between resume and assessment and the generative process of z^I are conditioned on the latent variable z^J of job description; (2) the topic number in semantic topic space of job description, resume, and interview assessment are constrained by $K^J < K^R = K^A$. Note that, comparing with JLMIA, which is constrained by $K^R = K^A = CK^J, C \geq 1$, assumption 2, here is looser and more reasonable. Guided by those assumptions, we define the generation and inference of Neural-JLMIA in the following, where the directed graph can be found in Figure 4(a).

5.2.1 Generative Process. Here, with assumption 1, we construct Neural-JLMIA based on the following generative probabilistic model:

$$p(J, R, A, z^J, z^I) = p(J|\theta^J, \beta^J)p(R|\theta^I, \varphi^R)p(A|\theta^I, \varphi^A)p(z^I|z^J)p(z^J), \tag{8}$$

where θ^J and θ^I are topic distributions projected from z^J and z^I, respectively; In addition, $p(J|\theta^J, \beta^J)$, $p(R|\theta^I, \varphi^R)$, and $p(A|\theta^I, \varphi^R)$ can be formulated similar as Equation (7). Here, we further produce other items above for each application m following NTM [39, 58]:

$$\begin{aligned} p(z^J_m) &= N(\mu^J, (\delta^J)^2), \\ \theta^J_m &= Softmax(f_{dJ}(z^J_m)), \\ p(z^I_m|z^J_m) &= N(\mu^I(z^J_m), (\delta^I(z^J_m))^2), \\ \mu^I_m &= f_{\mu^I}(z^J_m),\ \log(\delta^I_m)^2 = f_{\delta^I}(z^J_m), \\ \theta^I_m &= Softmax(f_{dI}(z^I_m)), \end{aligned} \tag{9}$$

where, the latent random variable z^J is sampled from the Gaussian distribution prior $p(z^J)$ with parameters $\mu^J = 0$ and $\delta^J = I$, and the projection between z^J and θ^J is deterministic without sampling produce. Here, we apply a **full connect** (**FC**) layer $f_*(\cdot)$ followed by Softmax function as the implementation of the projection network.

5.2.2 Inference Process. Following the framework of neural variational inference, to approximate the true posterior, we assume the infer model has the factorized form $q(z^J, z^I|J, R, A) = q(z^J|J)q(z^I|R, A)$ and introduce an inference network to generate variational parameters. Here, we first specify $q(z^J|J)$ as

$$\begin{aligned} q(z_m^J|J_m) &= N(\mu^{J'}(J_m), (\delta^{J'}(J_m))^2), \\ \mu^{J'} &= f_{\mu^{J'}}(e_m^J),\ \log(\delta^{J'})^2 = f_{\delta^{J'}}(e_m^J), \end{aligned} \tag{10}$$

where continuous vector e_m^J is the embedding of job description outputted by a encoder network. In this article, for simplicity, we adopt this encoder architecture by FC layers with **bag-of-words** (**BOW**) vector J_m^{bow} of the job description as input, i.e.,

$$h_m^J = g(f_{eJ1}(J_m^{bow})),\ e_m^J = g(f_{eJ2}(h_m^J)), \tag{11}$$

where h_m^J represents the hidden layer. Note that the encoders of resume and interview assessment have similar definitions with different network parameters and inputs.

Then, we define $q(z^I|R, A)$ as following:

$$\begin{aligned} q(z_m^I|R_m, A_m) &= N(\mu^{I'}(z_m^R, z_m^A), (\delta^{I'}(z_m^R, z_m^A))^2), \\ \mu_m^{I'} &= f_{\mu^{I'}}(e_m^I),\ \log(\delta_m^{I'})^2 = f_{\delta^{I'}}(e_m^I), \end{aligned} \tag{12}$$

where e^I is the joint embedding of resume R and interview assessment A. It is produced by the similar network as the Equation (11) with the BOW vector of the combination of R and A under the union vocabulary $V^I = V^R \cup V^A$ as the input.

Then, according to the variational inference framework, we can infer all latent variables in the Neural-JLMIA model by minimizing the following loss function for each instance (J_m, R_m, A_m):

$$\begin{aligned} \mathcal{L}_m = {} & D_{KL}(q(z_m^J|J_m)||p(z^J)) + D_{KL}(q(z_m^I|R_m, A_m)||p(z^J)) \\ & -\mathbb{E}_{q(z_m^J)}[logp(J_m|\theta_m^J, \varphi^J)] - \mathbb{E}_{q(z_m^R)}[\log p(R_m|\theta_m^I, \varphi^R)] - \mathbb{E}_{q(z_m^A)}[\log p(A_m|\theta_m^I, \varphi^A)], \end{aligned} \tag{13}$$

where the terms in the last line are the conditional likelihood of the generation for job description, resume, and assessments, respectively, which can be approximated for each J_m, R_m, and A_m by sampling [26] with the formulation:

$$\begin{aligned} \mathbb{E}_{q(z_m^J)}[logp(J_m|\theta_m^J, \varphi^J)] &= \sum_{n=1}^{N_m^J} \log \theta_m^J \cdot \varphi^J_{*, w_{m,n}^J}, \\ \mathbb{E}_{q(z_m^I)}[\log p(R_m|\theta_m^I, \varphi^R)] &= \sum_{n=1}^{N_m^R} \log \theta_m^I \cdot \varphi^R_{*, w_{m,n}^R}, \\ \mathbb{E}_{q(z_m^I)}[\log p(A_m|\theta_m^I, \varphi^A)] &= \sum_{n=1}^{N_m^A} \log \theta_m^I \cdot \varphi^A_{*, w_{m,n}^A}. \end{aligned} \tag{14}$$

6 TECHNICAL DETAILS OF R-JLMIA

After introducing JLMIA and Neural-JLMIA, which can mine the strong relationship among job descriptions, resumes, and interview assessments effectively, here, we turn to refine JLMIA with R-JLMIA to tackle all challenges (i.e., G1, G2.1–G2.3) together, i.e., modeling both the relationship and individual characteristics among three collections, simultaneously. Figure 4(b) shows the directed graph of R-JLMIA. To be specific, similar to Neural-JLMIA, R-JLMIA infers latent topic spaces φ^J, φ^R, and φ^A in job descriptions, resumes, and interview assessments, respectively, through variational distributions parametrized by neural networks. And the topic distribution θ^J, θ^R, and $\theta^A_{1:D}$ is produced from latent variables, z^J, $z^R = \{z_c^R, z_s^R\}$ and $z^A_{1:D}$ for each document in each collection by neural networks.

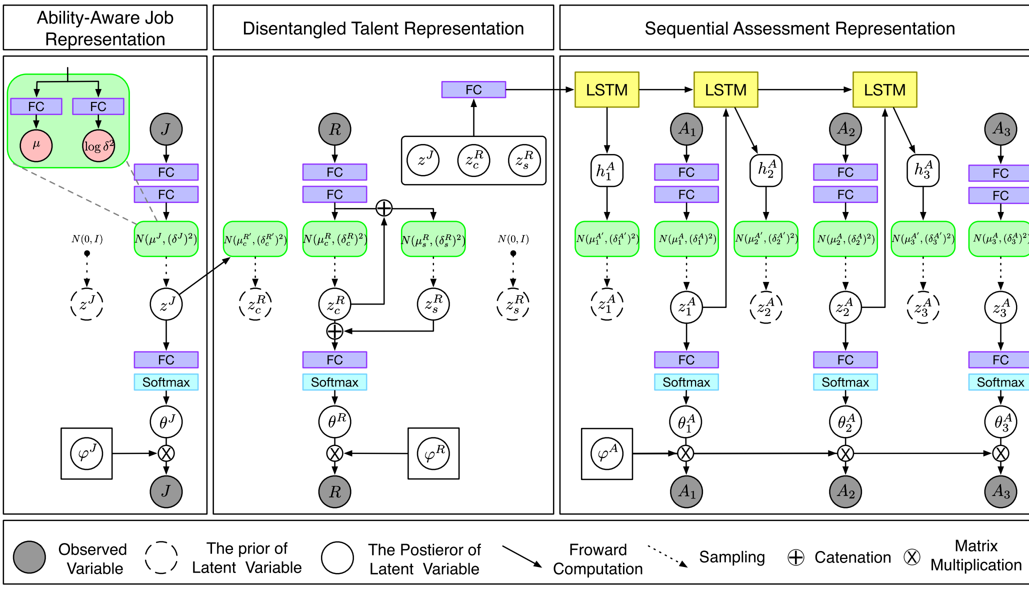


Fig. 5. The network architecture of R-JLMIA.

Then, our first goal to handle the relationship among different collections (G1) can be further transformed to model the relationship among those latent variables. On one hand, considering that candidates tend to write their resume with the job description as guidance, we assume that the latent variable z^R is generated with the latent variable z^J as the conditional information. On the other hand, noting that interviewers would like to design an interview process based on both job description and resume, we assume that both z^R and z^J guide the generative process of the latent variable $z^A_{1:D}$ in each interview round. Therefore, the generative model of R-JLMIA can be formulated as follows:

$$p(J, R, A, z^J, z^R_c, z^R_s, z^A_{1:D}) = p(J, z^J)p(R, z^R_c, z^R_s|z^J)p(A_{1:D}, z^A_{1:D}|z^{JR}), \tag{15}$$

where each term in the right side represents the individual generative process of each collection. Note that, compared with JLMIA and Neural-JLMIA, the assumption here is more flexible and reasonable without the constraint that resume and interview assessment must share the same topic distribution or latent variable.

In particular, three components are introduced to model each term in Equation (15), respectively, which are illustrated in Figure 5 with the overview of network structure. We first introduce the main idea in each component as follows:

- In Ability-aware Job Representation, a NTM model is utilized to learn the latent representation z^J of the job description.
- In Disentangled Talent Representation, a novel NTM model, named CDNTM, is proposed to disentangle the latent representation of candidates' core competences demonstrated in resume (G2.2). That is, CDNTM ties the core competence related variable z^R_c with the job latent variable z^J and uses one secondary variable z^R_s to model other personal capacities without conditional information.
- In Sequential Assessment Representation, with the combination $z^{JR} = z^J \oplus z^R_c \oplus z^R_s$ as conditional information, we propose SANTM to model the sequential structure in interview

assessments, learn latent representation z_d^A for assessment report A_d in each interview round (G2.3).

Besides, we assume different collections possess distinct semantic topic spaces with independent topic number K^J, K^R, and k^A. In particular, we set the topic number in job descriptions is smaller than that of the other two collections, i.e., $K^J \leq K^R$ and $K^J \leq K^A$ (G2.1). In the following, the technical details for each component would be introduced, successively.

6.1 Ability-Aware Job Representation

Here, we aim at distilling the ability-aware job representation from the job description. Specifically, following NTM [39, 58], we define the generative model $p(z_m^J, J_m)$ for each application J_m as

$$\begin{aligned} p(J_m, z_m^J) &= p(J_m|\theta^J, \varphi^J)p(z_m^J), \\ p(z_m^J) &= N(\mu^J, (\delta^J)^2), \\ \theta_m^J &= Softmax(f_{dJ}(z_m^J)), \end{aligned} \tag{16}$$

where, the prior distribution $p(z_m^J)$ is the standard Gaussian distribution in which $\mu^J = 0$ and $\delta^J = I$. For convenience, we name this NTM as **Ability-aware Job NTM** (**AJNTM**).

In the inference process, we define one Gaussian distribution $q(z_m^J|J_m)$ to approximate the true posterior as follows:

$$\begin{aligned} q(z_m^J|J_m) &= N(\mu^{J'}(J_m), (\delta^{J'}(J_m))^2), \\ \mu_m^J &= f_{\mu^{J'}}(e_m^J), \ \log(\delta_m^J)^2 = f_{\delta^{J'}}(e_m^J), \end{aligned} \tag{17}$$

where continuous vector e_m^J is the embedding of job description outputted by the similar networks as that for e^J in the Equation (11).

Then, we can approximate the posterior distribution over z^J by minimizing the negative variational lower bound for likelihood of job description as follows:

$$\mathcal{L}_m^J = D_{KL}(q(z_m^J|J_m)||p(z^J)) - \mathbb{E}_{q(z_m^J)}[\log p(J_m|z_m^J, \varphi^J)]. \tag{18}$$

6.2 Disentangled Talent Representation

After obtaining ability-aware job representation of job description, we turn to extract latent representation for candidate's resume. It is supposed to represent candidates' individual ability level and imply the fitness to job post they applied. However, not all contents in a resume play equal roles in demonstrating the candidate's competence. Intuitively, the content related to the job description represents the core competence of candidates and catch more attention from recruiters and interviewers. By contrast, some other certain information plays a secondary role in the candidate's application and interview assessment. Along this line, we design a novel NTM structure, named CDNTM, to disentangle the representation of talents' core competences in resume with the job representation z^J as conditional information. Specifically, we assume that the latent representation of the candidate's resume can be divided into two parts, i.e., the competence-related variable z_c^R (or named core variable) and secondary variable z_s^R. The competence-related variable aims at capturing the candidate's core competence required by the job posting. Therefore, we assume z_c^R is generated from job representation z^J. And the secondary variable z_s^R can be a vector that represents other uncertain information. Guided by the directed graph in the Figure 4(b), we can formulate the generative model $p(R_m, z_{c,m}^R, z_{s,m}^R)$ for each R_m as follows:

$$p(R_m, z_{c,m}^R, z_{s,m}^R) = p(R_m|\theta_m^R, \varphi^R)p(z_{c.m}^R|z_m^J)p(z_{s,m}^R). \tag{19}$$

To be specific, we first model the prior of competence-related variable $z^R_{c,m}$ guided by the job representation z^J_m, which has been learned by AJNTM, and can be regarded as the observed conditional information in CDNTM. That is

$$\begin{aligned} p(z^R_{c,m}|z^J) &= N(\mu^R_c(z^J_m), (\delta^R_c(z^J_m))^2), \\ \mu^R_{c,m} &= f_{\mu^R_c}(z^J_m), \ \log(\delta^R_{c,m})^2 = f_{\delta^R_c}(z^J_m). \end{aligned} \tag{20}$$

In addition, we assume that the secondary variable z^R_s is distributed by the standard Gaussian $N(0, I)$. Then, we can generate the topic distribution θ^R_m as follows:

$$\theta^R_m = Softmax(f_{dR}(z^R_{c,m} \oplus z^R_{s,m})), \tag{21}$$

where $\oplus$ is the vector catenation operation.

Note that, we treat the mean $\mu^{R'}_{c,m}$ and s.d. $\delta^{R'}_{c,m}$ of the prior as neural functions of job representation z^J_m alone. This is sound and reasonable because we aims at capturing the core ability of candidates that both required by the job description and demonstrated in the resume, suggesting that either the job description or resume is capable of inferring the underlying semantics of core ability in job-resume pairs, i.e., the representation of latent variable $z^R_{c,m}$.

As for inference process, we construct the approximate posterior distributions for latent variables $z^R_{c,m}$ and $z^R_{s,m}$ by introducing an inference model. Contrast to the variational neural prior, where generate the prior of $z^R_{c,m}$ only based on conditional information z^J_m, here, we treat the posterior distributions of $z^R_{c,m}$ and $z^R_{s,m}$ as the output of neural network only with the resume R_m as the input. Specifically, we assume that the inference model has a factorized form $q(z^R_{c,m}.z^R_{s,m}|R_m) = q(z^R_{c,m}|R_m)q(z^R_s|z^R_{c,m}, R_m)$, specified as Gaussian distributions both:

$$\begin{aligned} q(z^R_{c,m}|R_m) &= N(\mu^{R'}_c(R_m), (\delta^{R'}_c(R_m))^2), \\ q(z^R_{s,m}|R_m, z^R_{c,m}) &= N(\mu^{R'}_s(R_m, z^R_{c,m}), (\delta^{R'}_s(R_m, z^R_{c,m}))^2). \end{aligned} \tag{22}$$

In particular, the variational parameters $\mu^R_{c,m}, \delta^R_{c,m}, \mu^R_{s,m}$, and $\delta^R_{s,m}$ can be computed by inference networks conditioned on resume R_m:

$$\begin{aligned} \mu^{R'}_{c,m} &= f_{\mu^{R'}_c}(e^R_m), \ \log(\delta^{R'}_{c,m})^2 = f_{\delta^{R'}_c}(e^R_m), \\ \mu^{R'}_{s,m} &= f_{\mu^{R'}_s}(e^R_m \oplus z^R_{c,m}), \ \log(\delta^{R'}_{s,m})^2 = f_{\delta^{R'}_s}(e^R_m \oplus z^R_{c,m}), \end{aligned} \tag{23}$$

where e^R_m is the embedding of resume R_m defined similarly as the Equation (11).

After the definitions of three components in CDNTM above, we can approximate the posterior of latent variables z^R_c and z^R_s by minimize the negative variational lower bound as follows:

$$\begin{aligned} \mathcal{L}^R_m = {} & D_{KL}(q(z^R_{c,m}|R_m)||p(z^R_{c,m}|z^J_m)) + D_{KL}(q(z^R_{s,m}|R_m, z^R_{c,m})||p(z^R_{s,m})) \\ & - \mathbb{E}_{q(z^R_{c,m}, z^R_{s,m})}[\log p(R_m|z^R_{c,m}, z^R_{s,m}, \varphi^R)], \end{aligned} \tag{24}$$

where, the first term aims at closing the distance between the prior and posterior of competence-related $z^R_{c,m}$. In other word, this term spurs the latent variable $z^R_{c,m}$ to represent the core competence of candidates and imply the fitness between job and candidates.

6.3 Sequential Assessment Representation

With the processes of Ability-aware Job Representation and Disentangled Talent Representation, we can learn the representations for both job description z^J and resume z^R_c, z^R_s. Then, in order to learn the representation for sequential interview assessment, we proposed one model, named SANTM, to handle the sequential structure in multiple round interview assessments. Specifically, given the sequence of interview assessments $A_m = [A_{m,1}, A_{m,2}, \ldots, A_{m,D}]$, we consider the following conditional generative model, guided by the direct graph model in the Figure 4(b):

$$p(A, z^A_{m,1:D}|z^{JR}_m) = \prod_{d=1}^{D} p(A_{m,d}|\theta^A_{m,d}, \varphi^A)p(z^A_{m,d}|z^A_{m,<d}, z^{JR}_m), \tag{25}$$

where z^{JR}_m is the combination of representations of job description and resume, i.e., $z^{JR}_m = z^J_m \oplus z^R_{c,m} \oplus z^R_{s,m}$. In other word, the generation of each interview assessment round would be guided by the job, resume, and historical assessment report. In the following, we define each term in the right side.

To be specific, we first integrate the information of historical interview assessments $z^A_{m,<d}$ by LSTM network [21]:

$$h^A_{m,d} = LSTM(h^A_{m,d-1}, z^A_{m,d-1} \oplus z^{JR}), \tag{26}$$

where $h^A_{m,0} = \mathbf{0}$, and $z^A_{m,0} = \mathbf{0}$. Note that several other types of neural networks can also model the historical interview assessments, such as the linear layer and **Gated Recurrent Unit** (**GRU**) [14]. However, the LSTM layer achieves the best performance in our experiments.

Then, we can define the prior distribution $p(z^A_{m,d}|z^A_{m,<d}, z^{JR}_m)$ as follows:

$$\begin{aligned} p(z^A_{m,d}|z^A_{m,<d}, z^{JR}_m) = N(\mu^A(z^A_{m,<d}, z^{JR}_m), (\delta^A(z^A_{m,<d}, z^{JR}_m))^2), \\ \mu^A_{m,d} = f_{\mu^A}(h^A_{m,d}), \log(\delta^A_{m,d})^2 = f_{\delta^A}(h^A_{m,d}). \end{aligned} \tag{27}$$

Next, the topic distribution $\theta^A_{m,d}$ of the assessment $A_{m,d}$ can be produced by

$$\theta^A_{m,d} = Softmax(f_{d^A}(z^A_{m,d})). \tag{28}$$

Last, to infer the posterior distribution of latent variable $z^A_{m,1:D} = q(z^A_{m,1:D}|A_{m.1:D})$, which can been factorized as $\Pi^D_{d=1} q(z^A_{m,d}|A_{m,d})$, we define each item as follows:

$$\begin{aligned} q(z^A_{m,d}|A_{m,d}) = N(\mu^{A'}_d(A_{m,d}), (\delta^{A'}_d(A_{m,d}))^2), \\ \mu^{A'}_{m,d} = f_{\mu^{A'}}(e^A_{m,d}),\ \log(\delta^{A'}_{m,d})^2 = f_{\delta^{A'}}(e^A_{m,d}), \end{aligned} \tag{29}$$

where the network $f^A_\mu(\cdot)$, $f^A_\delta(\cdot)$ are shared among different interview rounds; $e^A_{m,d}$ is the embedding of interview assessment $A_{m,d}$ defined similarly as the Equation (11).

Guided by the neural variation inference framework, the variational lower bound for interview assessment sequence $A_m = [A_{m,1}, A_{m,2}, \ldots, A_{m,D}]$ of each candidate can be formulated as follows:

$$\mathcal{L}^A_m = \sum_{d=1}^{D} D_{KL}\left(q\left(z^A_{m,d}|A_{m,d}\right)||p\left(z^A_{m,d}|z^A_{m,<d}, z^{JR}_m\right)\right) - \mathbb{E}_{q(z^A_{m,d})}\left[\log p\left(A_{m,d}|z^A_{m,d}, z^{JR}_m, \varphi^A\right)\right]. \tag{30}$$

Note that, when computing the prior for $z^A_{m,d}$, we use the generation model $p(z^A_{m,d}|z^A_{m,<d}, z^{JR}_m)$ with latent representation of historical assessment reports $z^A_{m,<d}$ as the input, where $z^A_{m,<d}$ are inferred from the posterior $q(z^A_{m,<d}|A_{m,<d})$ as shown in Figure 5, not from their prior like that in [32].

6.4 Joint Learning

As mentioned above, our R-JLMIA consists of three components, i.e., Ability-aware Job Representation, Disentangled Talent Representation, and Sequential Assessment Representation. Here, in order to jointly learn the representations for the job, resume, and interview assessment, we optimize the composite loss with the consideration of $\mathcal{L}_m^J$, $\mathcal{L}_m^R$, and $\mathcal{L}_m^A$ for each training instance (J_m, R_m, A_m) as follows:

$$\begin{aligned}
\mathcal{L}_m^* &= \mathcal{L}_m^J + \mathcal{L}_m^R + \mathcal{L}_m^A \\
&= \underbrace{-\mathbb{E}_{q(z_m^J)}[logp(J_m|\theta_m^J, \varphi^J)] - \mathbb{E}_{q(z_m^R)}[\log p(R_m|\theta_m^I, \varphi^R)] - \textstyle\sum_{d=1}^D \mathbb{E}_{q(z_{m,d}^A)}[\log p(A_{m,d}|\theta_{md}^A, \varphi^A)]}_{\mathcal{L}^{BCE}} \\
&\quad + \underbrace{D_{KL}(q(z_m^J|J_m)||p(z_m^J)) + D_{KL}(q(z_{s,m}^R|R, z_{c,m}^R)||p(z_{s,m}^R))}_{\mathcal{L}^{KL1}} \\
&\quad + \underbrace{D_{KL}(q(z_{c,m}^R|R_m)||p(z_{c,m}^R|z_m^J)) + \textstyle\sum_{d=1}^D D_{KL}(q(z_{m,d}^A|A_{m,d})||p(z_{m,d}^A|z_{m,<d}^A, z_m^{JR})),}_{\mathcal{L}^{KL2}}
\end{aligned} \tag{31}$$

where the loss function can also be re-divided into three elements in term of individual optimization goal and functionality. $\mathcal{L}_m^{BCE}$ is actually the conditional neg-likelihood for the job description, resume, and interview assessment, which can be approximated by sampling similar as Equation (14). $\mathcal{L}_m^{KL1}$ and $\mathcal{L}_m^{KL2}$ try to close the distance between the prior and posterior for all latent variables. In particular, with the customized prior by the conditional information, the $\mathcal{L}_m^{KL2}$ establishes the relationship among different document collections.

7 APPLICATIONS

In real-world interview scenarios, once a candidate applies one job, the recruiters need to identify whether the candidate is suitable for the job based on their resume or preliminary communication by telephone in the pre-screening stage. If so, the candidate would be sent to the following in-person interviews for further evaluation in a face-to-face manner. Here, to enhance those two stages in the job interview process, we turn to introduce two applications enabled by R-JLMIA, i.e., Person-Job Fit and Skill Recommendation for Interview Assessment.

7.1 Person-Job Fit

Formally, given a job description J_g and a resume R_g, the objective of person-job fit is to measure their matching degree. Note that, in the pre-screening stage, the recruiters need to identify whether the pair of the job and the candidate is suitable to each other once one applies for this job. Thus, here, we follow [5] and formulate the person-job fit as a classification task, not a recommendation task. To be specific, we proposed two approaches based on R-JLMIA to evaluate the person-job fit, i.e., similarity based approach and classifier based approach, corresponding to unsupervised and supervised methods, respectively.

7.1.1 Similarity Based Approach. In R-JLMIA, the latent variable z_c^R aims at representing the abilities shared by both job description and resume. Thus, one natural option to measure the person-job fit is to project the job description and resume into the semantic space of z_c^R and, then, computing the similarity between their projections. Specifically, we need to first infer the latent representation z_g^J of job description by the posterior network $q(z_g^J|J_g)$. To avoid sampling, we set $z_g^J = \mu_g^J$. Then, we can infer the distribution of projections of J_g and R_g by the prior network and posterior network of z_c^R in CDNTM as follows:

$$N(\mu_{c,g}^R, \delta_{c,g}^R) = p(z_{c,g}^R | z_g^J),$$
$$N(\mu_{c,g}^{R'}, \delta_{c,g}^{R'}) = q(z_{c,g}^R | R_g). \tag{32}$$

Finally, we have two directions to measure the suitability between job description and resume, by computing the divergence between those distributions of projections of J_g and R_g, such as KL divergence, or similarity between the exceptions of those distributions $\mu_{c,g}^R$ and $\mu_{c,g}^{R'}$, such as Cosine similarity.

7.1.2 Classifier Based Approach. Based on R-JLMIA, we can infer the posterior of latent variable z_g^J and z_c^R by the posterior network $q(z_g^J | J_g)$ and $q(z_{c,g}^R | R_g)$. The exception μ_g^J and $\mu_{c,g}^R$ can be regarded as the representations for the ability required by J_g and the core ability demonstrated in R_g. Thus, another solution for person-job fit is using these representations instead of original BOW as features to train classifiers, such as Random Forest.

7.2 Skill Recommendation for Interview Assessment

Here, we plan to recommend skills that should be investigated and evaluated during the following interview round, which can contribute to effective job interviews in both the pre-screening stage and in-person interviews. For example, in pre-screening stage, the recommended skills can enhance the design of the distinguishing questions in automatic interview systems [56] to screen candidates efficiently and effectively. For in-person interviews, this application provides helpful guidance to design effective interview procedures, like selecting suitable interviewers proficient in those recommended skills, and producing effective interview questions for systematically judging the competences of candidates.

Specifically, given the job description J_g, resume R_g and the historical assessment reports $A_{g,<d}$, we need forecast the topic distribution $\theta_{g,d}^A$ for the dth interview assessment round in the first step. Here, we start with inferring the latent joint representations z_g^{JR} of job and resume according to the posterior networks defined in AJNTM and CDNTM models. To avoid sampling, we use the catenation of the expectations for latent variable $z_g^J, z_{c,g}^R$, and $z_{s,g}^R$ as the joint representation, i.e., $z_g^{JR} = \mu_g^J \oplus \mu_{c,g}^R \oplus \mu_{s,g}^R$; Then, we output the latent representation for each historical assessment $A_{g,i}, i < d$ by leveraging the trained posterior network without sampling, i.e., $z_{g,i}^A = \mu_{g,i}^A$. Next, the latent representation $z_{g,d}^A = \mu_{g,d}^A$ for the next assessment can be forecasted by the prior network with Equations (26) and (27). Last, $z_{g,d}^A$ can be projected into the topic distribution $\theta_{g,d}^A$ of assessment by the Equation (28).

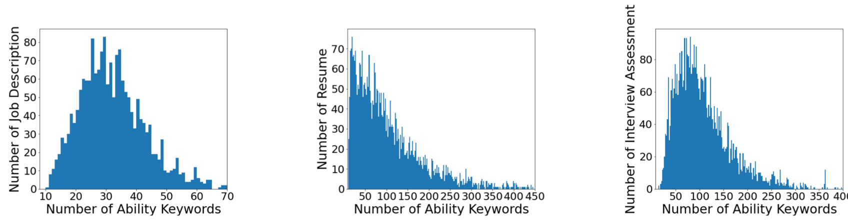
In the second step, with learned topic set φ^A, we can estimate the probability of each skill keywords s_i^A in vocabulary V^A that occurred on the next assessment round, by $M_{g,d}(s_i^A) = \theta_{g,d}^A \cdot \beta_{*,s_i^A}^A$. Then, we can recommend suitable skills for the next interview assessment by sorting the $M_{g,d}(s_i^A)$.

8 EXPERIMENTAL RESULTS

In this section, we will estimate the performance of our models based on extensive experiments conducted on a real-world interview dataset in terms of document modeling and two real-world applications, successively.

8.1 Data Description

The dataset used in the experiments is the historical interview data provided by a high-tech company in China, which contains a total of 14,702 candidate interview records. To be specific, with



(a) Job Description (b) Resume (c) Interview Assessment

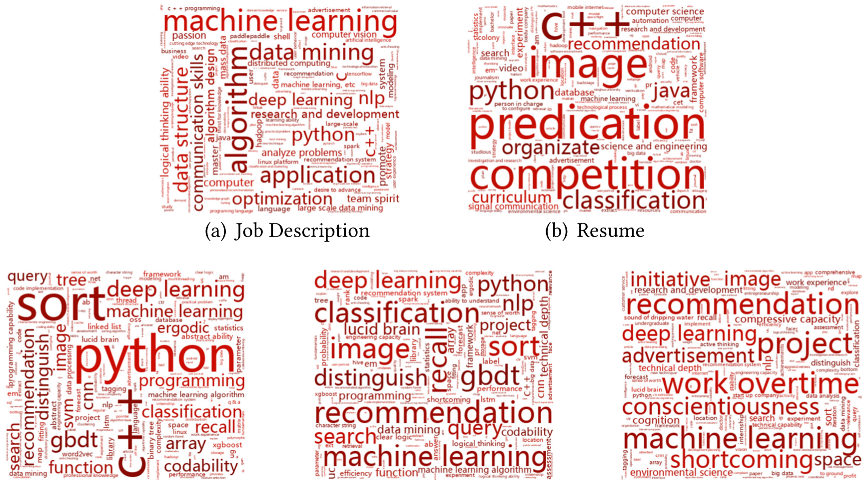
Fig. 6. The word distributions of job description, resume and interview assessment.

Table 2. The Statistics of the Dataset

Statistics	Values
# of job descriptions	1,920
# of resumes	3,890
# of interview assessments	4,664
Average ability keywords per job description	32.302
Average ability keywords per resume	107.720
Average ability keywords per assessment report in 1st interview round	40.078
Average ability keywords per assessment report in 2nd interview round	36.182
Average ability keywords per assessment report in 3rd interview round	31.194
The size of vocabulary for job description	2,796
The size of vocabulary for resume	6,320
The size of vocabulary for interview assessment	5,146

the help of several staffing experts, we manually screened records with high-quality interview assessment written by senior interviewers, and removed the records which lack details in job description or resume. In addition, in order to guarantee the quality of interview assessment, we only persevered the records with complete sequential interview assessment reports that consists of all three assessment reports during three-round interview assessments. After that, the filtered dataset contains 4,664 interview records related to 1,920 job positions and 3,890 candidates. Note that it is reasonable that the number of interview records is larger than that of resumes because a candidate may apply for multiple job positions and have experienced different interview procedures. Moreover, to generate the ability-related words in our collections, we followed the idea in [47] and trained a LSTM-CRF model to extract the possible ability-related words. Then, several staffing experts further cleaned the keywords. Finally, some statistics related to our pruned dataset can be found in Table 2. Furthermore, we illustrate the word number distribution of job descriptions, resumes, and interview assessments in Figures 6(a–c), respectively.

Moreover, as the discussion above, there exists a strong relationship among job descriptions, resumes, and interview assessments. Here, we provide an intuitive perspective about this relationship by illustrating keyword distribution. Specifically, we selected interview records related to job positions for data mining engineers as the example shown in Figure 7. As we all know, in job descriptions, this category of job positions not only requires basic programming ability, such as "Python" or "C++", but also expects that candidates acquire sufficient knowledge related to "machine learning" and "data mining", such as "deep learning' or "nlp" (namely natural language processing). Correspondingly, in resumes, candidates prefer to declare the programming language they adept at, and list experiences or projects that contain "prediction' or "classification" tasks to demonstrate their proficiency in data mining and machine learning. In addition, as for assessment



(a) Job Description (b) Resume

(c) 1st Interview Round (d) 2nd Interview Round (e) 3rd Interview Round

Fig. 7. The word cloud representation of job description, resume, and assessment reports in three rounds related to the jobs for data mining engineers, where the size of each keyword is proportional to its frequency.

reports in three interview rounds, we can find that the first interview round pays more attention to basic aptitude investigation about programming, while the second interview round is more concerned about core technology requirement, namely detailed technology of "machine learning", with keywords "**gradient boosting decision tree (gbdt)**", "classification", "nlp", and "recommendation". In addition, it seems that interviewers in the third round would like to evaluate candidates from a more comprehensive and high-level perspective, relevant to both "machine learning" based "projects" and personal quality based assessment, such as "shortcoming" and "conscientiousness", even personal attitude to "working overtime".

8.2 Experimental Setup

In this section, we introduce the detailed settings in our experiments, which can be split into two parts: Parameter Setting and Training Details.

8.2.1 Parameters Settings. In JLMIA, we empirically set fixed parameters as $\{\delta^2, \beta^J, \beta^R.\beta^E\} = \{0.01, \mathbf{0.1}, \mathbf{0.1}, \mathbf{0.1}\}$. In Neural-JLMIA and R-JLMIA, we first need to embed each word in job descriptions, resumes, and interview assessments. Here, we used the parameters of pre-trained Skip-gram models within those three collections, respectively, to initialize the word embedding vectors parameters v^J, v^R, and v^A. The embedding size of each word is set as 256. Note that those word embeddings would be further fine-tuned in the training process. Then, as for the document embedding layers of three collections, we also set all dimensions of embedding vectors e^J, e^R, e^A as 256. Besides, the dimensions of all latent variables z^J and z^I in Neural-JLMIA and z^J, z^R_c, z^R_s, and z^A_d in R-JLMIA were set as 128. Moreover, the dimension of hidden state for LSTM in SANTM model was set as 128. And, we implement the non-linear active function g by ReLU [18] in Neural-JLMIA, AJNTM, and CDNTM models, and by LeakyReLU [36] with negative slope parameter 0.2 in SANTM model. In addition, we use batch normalization following each FC layer, which tends to improve model performance and speed up training [6, 20]. As for topic number parameters K^J, K^R, and K^A, we would discuss the selection for them with more details in Section 8.4.

8.2.2 Training Detail. As for JLMIA, we utilized the EM algorithm to inference all variational parameters and latent variables based on the generative process in Algorithm 1, the details of which can be found in the Appendix. Here, we turn to introduce training details of Neural-JLMIA and R-JLMIA. Specifically, for R-JLMIA, to achieve better convergence result, we first pre-trained AJNTM model with the loss function in Equation (18) for 1,000 epochs. Then trained the combination of AJNTM and CDNTM model for 1,000 epochs with the joint loss function by summing Equations (18) and (24). Last, we trained the overall model according to the complete loss function in Equation (31). Similarly, as for Neural-JLMIA, we initialized partial parameters related to modeling job description with pre-trained AJNTM, then we trained overall Neural-JLMIA based on loss function in Equation (13). Moreover, we set batch size as 32, and apply the Adam optimization algorithm with the learning rate of 0.001 to optimize parameters. Here, we randomly select 80% interview document tuples as the training dataset, 10% for the validation dataset, and the other is used to test the performance.

8.3 Benchmark Methods

To evaluate the performance of our models, we introduce several classic or state-of-the-art representation learning methods as baseline models. Specifically,

— **Bag-of-words Representation (BOW).** We use the BOW vector of each job description, resume, and interview assessment under their vocabularies as their representations, respectively. The *i*th dimension of each vector for each document is the frequency of the *i*th word of the corresponding vocabulary.
— **Word Embedding based Representation (Word2Vec) [40].** We, respectively, trained the Skip-Gram models in job descriptions, resumes, and interview assessments. Then, the average word vector of each document in three collections is regarded as its representation.
— **Latent Dirichlet Allocation (LDA) [9].** We merged the job description the candidate applied, their resume, and corresponding interview assessment as a document for training LDA. Then, for testing, we infer the topic distributions of those three documents as their representations with the individual document as the input, respectively. In addition, we also use the combination of the job description and resume for training LDA in terms of the application of person-job fit. We call those two models as LDA and LDA_jr, respectively.
— **Neural Topic Model (NTM) [39].** NTM is a state-of-the-art variant of LDA with the neural inference network . Similar to LDA, we merged the job description a candidate applied for, their resume, and corresponding interview assessment as a document for training NTM. Then, the topic distributions of those three documents are inferred as their representations with the individual document as the input, respectively. The network structure of encoder and decoder is set same as that in AJNTM. In addition, we also trained NTM without assessment reports as input. We call those two models as NTM and NTM_jr, respectively.
— **Batch Normalization-Variational Autoencoder (BNVAE) [78]**. BNVAE is a state-of-the-art variant of VAE [26] for text modeling, which uses the batch normalization on mean parameters of the Gaussian latent variables. Here, in the training procedure, we use a 256-dimension word embedding layer and a shared LSTM layer with 256 hidden size as the encoder to embedding the job description the candidate applied, their resume, and corresponding interview assessment, respectively. Then, the mean of the last states of those three collections is fed to full connection layers to produce the parameters of the Gaussian latent variable with the size of 128. The MLP network with a hidden layer sized 256 has been used as the decoder to project the latent variable into the word distribution on vocabulary. Then, for testing, we infer the expectation of the latent variable as the representation with each

collection as input, respectively. In addition, we also trained BNVAE without assessment reports as input. We call those two models as BNVAE and BNVAE_jr, respectively.

Besides, in order to evaluate the effectiveness of different components and assumptions in our approaches, we also constructed several variants of our approaches as additional baselines:

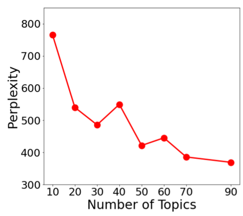
— **JLMIA_jr** is a variant of JLMIA, where the input data only contains job descriptions and resumes with the empty interview assessment.
— **Neural-JLMIA_jr** is a variant of Neural-JLMIA with only job descriptions and resumes as input data.
— **R-JLMIA_jr** is the combination of AJNTM and CDNMT models to learn the representations of job descriptions and resumes.
— **R-JLMIA w/o z_s^R** is a variant of R-JLMIA where we do not disentangle the competence-related variable z_c^R and secondary variable z_s^R in CDNTM model. In other words, we only apply a latent variable z_c^R in CDNTM to learning the representation of resumes, which amounts to removing the latent variable z_s^R.
— **R-JLMIA_a** is a variant of the SANTM model where we only learn the representation of interview assessments by replacing the joint representation z^{JR}, learned by AJNTM and CDNTM, with the embedding of the combination of the job description and resume. Specifically, the embedding vector can be produced by the MLP networks with the splice of their BOW vectors as the input.
— **R-JLMIA_Linear** is a variant of R-JLMIA where we replace the LSTM unit in SANTM with a full connection layer followed by the Relu activation.
— **R-JLMIA_GRU** is a variant of R-JLMIA where we replace the LSTM unit in SANTM with GRU [14].

Actually, we can also regard JLMIA and Neural-JLMIA as the variants of R-JLMIA, where the basic assumptions of JLMIA and Neural-JLMIA are contained in that of R-JLMIA.

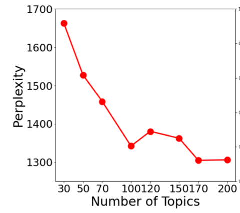
8.4 Evaluation on Jointly Document Modeling

Here, we aim at evaluating the quality of document modeling for job descriptions, resumes, and interview assessments, respectively. Specifically, as a document generative model, we use the perplexity (PPL) as the main metric to evaluate the generative capacity and the quality of learned representation. In document modeling, perplexity is computed as $exp(-\frac{1}{M}\sum_{m=1}^{M}\frac{1}{N_m}p(d_m))$, where M is the number of documents, N_m represents the length of the mth document d_m, i.e., J_m, R_m, or A_m. Since $p(d_m)$ is intractable in our models and baselines. We use the variational lower bound to estimate the perplexity following [39], which is the upper bound of perplexity. And, following [52], test dataset was selected as the held-out dataset.

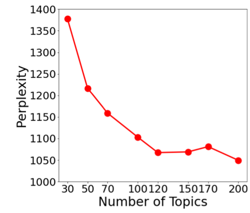
8.4.1 Performance Analysis. Here, we first evaluate the parameter sensitivity for R-JLMIA by varying the numbers of topic K^J, K^R, and K^A, respectively. Specifically, we first analyzed the impact of K^J on the performance of AJNTM for modeling the job description. Then, the performance of CDNTM for modeling resume would be evaluated with pre-trained AJNTM model and suitable and fixed K^J. Finally, the performance of SANTM for modeling the interview assessment will be estimated with pre-trained AJNTM and CDNTM models, where the K^R and K^J are fixed at suitable numbers. As the result shown in Figure 8, we can observe that the perplexity of each word tends to decrease as the number of topic increase, which is same as that in probabilistic topic models. And the suitable parameters should be located at the minimum number when the perplexity curve starts to converge. Thus, we chose the topic number K^J, K^R, and K^A as 50, 100, and 120, successively for the following experiments.



(a) Job Description



(b) Resume



(c) Interview Assessment

Fig. 8. The performance of R-JLMIA for modeling different collections with different parameters.

Table 3. The Performance of Our Models and Baselines over the Document Modeling

	Job Description		Resume		Assessment	
	K	PPL	K	PPL	K	PPL
LDA_jr	70	812.58	70	1,800.75	–	–
LDA	150	993.25	150	1,917.84	150	1,486.33
NTM_jr	100	1,008.81	100	1,324.90	–	–
NTM	150	1,610.61	150	1,784.01	150	1,161.09
BNVAE_jr	–	1,325.24	–	2,067.28	–	–
BNVAE	–	1,705.85	–	2,390.00	–	1,691.79
JLMIA_jr	30	877.34	60	1,781.35	–	–
JLMIA	50	916.89	100	1,622.95	100	1,200.59
Neural-JLMIA_jr	50	414.88	100	1,346.56	–	–
Neural-JLMIA	50	412.99	150	1,477.48	150	1,114.90
R-JLMIA w/o z_s^R	50	407.64	100	1,367.15	120	1,124.98
R-JLMIA_jr	50	**395.81**	100	1,342.19	–	–
R-JLMIA_a	–	–	–	–	120	1,537.45
R-JLMIA_Linear	50	402.57	100	1,371.60	120	1,151.07
R-JLMIA_GRU	50	396.84	100	1,340.51	120	1,110.48
R-JLMIA	50	399.41	100	**1,337.82**	120	**1,067.31**

We show the parameter of the number of topics if possible.
The best result is highlighted in bold.

Then, we compare the performance of different models, where we also tune the topic number in each baseline liked that in R-JLMIA. According to the result shown in Table 3, we realize that R-JLMIA outperforms all other models. Meanwhile, compared with LDA and NTM, all JLMIA based models (i.e., JLMIA, Neural-JLMIA, and R-JLMIA) have achieve a significant improvement on the performance of document modeling. It verifies the rationality of our assumptions on the relation and individual character among different document collections. Interestingly, we also find that Neural-JLMIA performs better than JLMIA, which may be owing to the flexibility and powerful fitting ability of neural variational inference.

In addition, we further compare our models with their variants. As for R-JLMIA, we first find that removing the latent variable z_s^R leads to a significant performance decrease on modeling resume and interview assessment. It demonstrate the necessity of disentangling competence-related variables and secondary variables. Moreover, without modeling job description and resume, R-JLMIA_a also fails to perform better on modeling interview assessment. It shows that MLP networks can not capture sufficient representation from job descriptions and resumes. In particular, when the input data only contains job description and resume, i.e., comparing R-JLMIA_jr, JLMIA_jr, and Neural-JLMIA_jr, we note that R-JLMIA_jr achieves the best performance. That further verifies the extensibility of R-JLMIA for different situations. In addition, compared with R-JLMIA_Linear and R-JLMIA_GRU, R-JLMIA with the LSTM layer to capture the evolution of semantic information over different interview rounds can model the assessment reports better.

8.5 Performance of Person-Job Fit

Here, we evaluate the performance of different models in terms of person-job fit. Specifically, given a job description and a resume, we aim to measure the matching degree based on similarity based approaches and classifier based approaches mentioned above.

8.5.1 Data Preparation. As a classification task, the person-job fit requires both positive and negative samples to evaluate our approaches and baselines. However, in our datasets, only positive samples have been persevered, where candidates have passed the pre-screening stage and been evaluated in in-person interviews, even hired. Thus, we also need to prepare unsuitable pairs of job descriptions and resumes as negative samples. Although, we can intuitively regard the failed job applications as negative samples, we do not know the exact reasons behind these failures. For example, some failed applications are just due to the low pay benefits, or other similar reasons in offer negotiation. We cannot distinguish those samples from other candidates without the right skills due to the incomplete records in our data source. Therefore, we manually generated the same number of negative samples for both the training dataset, if needed, and the testing dataset by randomly selecting resumes and job descriptions from our dataset, i.e., the successful job interview records. Along this line, the experiments will only focus on the representation of latent topics, while interference from other factors will be impaired.

8.5.2 Adjustment for Benchmark Methods. For comparison, here, we selected Cosine and KL as the similarity metrics in similarity based approach and selected Random Forests, and GBDT as classifiers in classifier based approach. Please note that because the similarity between two BOW vectors under different vocabulary or two Word2Vec vectors under two collections are meaningless, we did not treat it as baselines, here. And for LDA, NTM, JLMIA, Neural-JLMIA, and their variants, KL divergence of the representation vectors of the job description and resume is actually the distance between two topic distributions for them, same as that in [55]. For BNVAE and BNVAE_jr, the Cosine similarity is computed between the expectations of the latent variables for job description and resume, and the KL divergence is derived on the distributions of those two latent variables.

8.5.3 Performance Analysis. Table 4 shows the person-job fit performance of our models and baselines with the area under the **precision-recall curve** (**PR AUC**) and **receiver operating characteristic curve** (**ROC AUC**) as metrics. From the results, we find that R-JLMIA achieves the best performance in both similarity based approaches and classifier based approaches. It indicates that our proposed approaches can effectively capture the latent relationship between job description and resume. JLMIA based models consistently outperform LDA based models, but not neural network based baselines, i.e., NTM and BNVAE, which indicates the powerful modeling ability of neural networks. Neural-JLMIA tends to capture better performance than both JLMIA, and other baselines, which demonstrates the effectiveness of neural variational inference and our approaches again.

In addition, we also have some interesting findings by comparing different variants of our approaches. First, we note that with modeling assessment, JLMIA based models have demonstrated a significant performance increase on similarity based approaches compared with JLMIA_jr, Neural-JLMIA_jr, and R-JLMIA_jr, respectively, with similar performance on classifier based approaches.

Table 4. The Person-Job Fit Performance of Our Approaches and other Baselines

	Cosine Similarity		Kullback–Leibler		Random Forest		GBDT	
	PR	ROC	PR	ROC	PR	ROC	PR	ROC
BOW	–	–	–	–	0.6733	0.6947	0.6122	0.6882
Word2Vec	–	–	–	–	0.8035	0.8256	0.7568	0.8066
LDA_jr	0.7340	0.6954	0.7299	0.7118	0.7932	0.8072	0.7789	0.8048
LDA	0.7184	0.6690	0.6996	0.6860	0.7491	0.7873	0.7479	0.7827
NTM_jr	0.7501	0.7471	0.7492	0.7601	0.8899	0.8782	0.8802	0.8727
NTM	0.7398	0.7784	0.7057	0.7505	0.8982	0.8908	0.8737	0.8686
BNVAE_jr	0.7724	0.7555	0.7273	0.7140	0.9030	0.8852	0.8714	0.8713
BNVAE	0.8029	0.8035	0.7471	0.7441	0.8918	0.8754	0.8705	0.8645
JLMIA_jr	0.7295	0.7417	0.6773	0.6775	0.8167	0.8386	0.8228	0.8303
JLMIA	0.7762	0.7710	0.7436	0.7425	0.8387	0.8486	0.8242	0.8347
Neural-JLMIA_jr	0.6926	0.7311	0.6852	0.7356	0.9006	0.8887	0.8963	0.8853
Neural-JLMIA	0.7845	0.8155	0.8051	0.8341	0.9020	0.8878	0.8731	0.8754
R-JLMIA w/o z_s^R	0.8506	0.8740	0.8839	0.9013	0.9337	0.9217	0.9344	0.9242
R-JLMIA_jr	0.8326	0.8561	0.8909	0.9032	0.9304	0.9177	0.9310	0.9227
R-JLMIA_Linear	0.8470	0.8657	0.8879	0.8929	0.9283	0.9163	0.9308	0.9239
R-JLMIA_GRU	**0.8667**	0.8764	0.8891	0.8930	0.9313	0.9174	0.9244	0.9216
R-JLMIA	0.8481	**0.8767**	**0.9050**	**0.9101**	**0.9346**	**0.9243**	**0.9361**	**0.9277**

The best result is highlighted in bold.

It shows that JLMIA based model can capture more discriminative representation with the guidance of interview assessment reports. However, the contrary phenomenons have appeared in LDA, NTM, and BNVAE based models. Second, by comparing R-JLMIA and R-JLMIA w/o z_s^R, we find that removing the latent variable z_s^R has not lead to a significant change on the person-job fit performance. It may be because both models have captured the core competences from resumes, which are the most essential criterion for measuring person-job fit. Third, we find that R-JLMIA achieves better performance than R-JLMIA_Linear and R-JLMIA_GRU. It indicates that the LSTM layer used in R-JLMIA may model the dependence over different interview rounds better, and, thus, provide better guidance for capturing the relation between job description and resume. Last but not least, our designed KL based similarity in R-JLMIA based models achieve consistently better performance than cosine similarity with a significant margin. It suggests that the similarity of distributions of latent variable z^J and z_c^R can provide a more accurate measurement of person-job fit than the similarity of expectation of topic distributions.

8.6 Performance of Skill Recommendation for Interview Assessment

To evaluate the performance of our methods on another application, i.e., skill recommendation for interview assessment, we compare the recommended skill keywords with the true assessment report, and use $F@N$ as the metric, which is the harmonic mean of the precision $P@N$ and recall $R@N$ for the top N recommended keywords:

$$\begin{aligned} F@N &= \frac{2P@N * R@N}{P@N + R@N}, \\ P@N &= \frac{N^+}{N}, \; R@N = \frac{N^+}{N_A}, \end{aligned} \tag{33}$$

where N_A is the number of keywords in the assessment report, N^+ denotes the number of keywords in recommended keywords that are the same as or strongly related to a keyword in the true assessment report. In particular, we consider the strong relationship among keywords when computing those criteria. For example, "python" is one of the "programming languages". "CNN" is the abbreviation of "convolutional neural network". Here, to figure out those strong relationship terms, we first cluster those keywords based on the simple rules, like that cluster two phrases when one is included in the other, such as "C++" and "C++ programming". Then, several recruiting experts are asked to screen those relationship terms or add some new ones. Finally, 17,900 strong relationship terms have been found in the assessment vocabulary.

8.6.1 *Adjustment for Benchmark Methods.* For the comparison, we need to adjust baselines for recommending skills. Specifically, for topic model based approaches, we need to predicate the topic distribution $\theta^A_{g,d}$ of the next interview assessment $A_{g,d}$ based on the job description J_g, resume R_g and historical assessment reports $A_{<d}$ in the first step. Here, we list the detailed adjustments for different models as follows:

— **LDA and NTM.** The topic distribution $\theta^{JRA}_{g,<d}$ of the combination of the corresponding job description, resume, and historical assessment reports can be inferred as the prediction of the next interview assessment.
— **JLMIA.** First, the topic distribution of job description which can be further projected into the topic space in interview assessment as θ^{J2A}_g. Then, the topic distributions θ^R_g and $\theta^A_{g,<d}$ of resume or historical assessment can also be regarded as the prediction of interview assessment. Finally, the average of θ^{J2A}_g, θ^R_g, and $\theta^A_{g,<d}$ is regarded as the predication.
— **Neural-JLMIA.** As the advanced variant of JLMIA, Neural-JLMIA can learn the joint topic distribution $\theta^{RA}_{g,<d}$ of the combination of resume and historical assessment. Thus, different from that in JLMIA, the average of $\theta^{RA}_{g,<d}$ and θ^{J2A}_g is used as the final prediction.

Then, in the second step, similar as that in R-JLMIA, we can figure out the generative probability $M_{g,d}(s^A_i)$ of each keyword s^A_i, respectively, to recommend suitable skills for interview.

For **BNVAE**, we used the tuple of the job description, resume, and historical assessment reports as the input of the encoder, and infer the expectation of the latent variable as the input of the decoder to predict the word distribution, i.e., the generative probability $M_{g,d}(s^A_i)$ of each keyword s^A_i, in the next interview round. In addition, we also add another basic baseline, which recommends skill keywords by sorting keywords according to their frequency in historical interview reports of each interview round, called **Frequency**.

8.6.2 *Performance Analysis.* Table 5 shows the performance of skill recommendation for interview based on F@5, F@10, and F@15. According to the results, we can find that R-JLMIA and its variants outperform other baselines with a significant margin. In addition, due to the lack of a specific design for modeling the sequence structure of interview assessment, JLMIA and Neural-JLMIA have demonstrated similar performance as LDA, NTM, and BNVAE. Meanwhile, compared with R-JLMIA, R-JLMIA w/o z^R_s, and R-JLMIA_a have suffered from performance decrease, especially in terms of F@5 metric in all three interview rounds. It indicates that modeling jobs and resumes and disentangling the competence-related and secondary variables are both beneficial for capturing the meaningful representation of job or resume, which guides skill recommendation for interview assessment. In addition, R-JLMIA and R-JLMIA_GRU achieve better performance than R-JLMIA_Linear, which indicates that the simple linear layer cannot capture the evolution of the semantic information over different interview rounds. In particular, R-JLMIA tends to achieve the best performance with the LSTM layer.

Table 5. The Performance of Our Approaches and Baselines over Skill Recommendation for Interview Assessment

	1st Round			2nd Round			3rd Rund		
	F@5	F@10	F@15	F@5	F@10	F@15	F@5	F@10	F@15
Frequency	0.0646	0.0371	0.0797	0.1011	0.0883	0.1167	0.1283	0.1153	0.1249
LDA	0.1188	0.1521	0.1771	0.1363	0.1687	0.1862	0.1267	0.1513	0.1616
NTM	0.1294	0.1659	0.1874	0.1404	0.1852	0.2051	0.1289	0.1600	0.1786
BNVAE	0.1431	0.1796	0.2006	0.1541	0.1988	0.2191	0.1325	0.1635	0.1773
JLMIA	0.1247	0.1655	0.1911	0.1466	0.1817	0.2037	0.1169	0.1492	0.1651
Neural-JLMIA	0.1195	0.1547	0.1755	0.1361	0.1712	0.1939	0.1269	0.1572	0.1790
R-JLMIA w/o z_s^R	0.1580	0.2091	0.2360	0.1554	0.2042	0.2300	0.1255	0.1787	0.2013
R-JLMIA_a	0.1481	0.2038	0.2312	0.1375	0.1773	0.2008	0.1137	0.1530	0.1745
R-JLMIA_Linear	0.0757	0.1284	0.1582	0.1417	0.1890	0.2112	0.1290	0.1678	0.1888
R-JLMIA_GRU	0.1652	0.2122	**0.2443**	**0.1672**	0.2098	0.2338	0.1438	0.1868	0.2070
R-JLMIA	**0.1703**	**0.2145**	0.2413	0.1620	**0.2120**	**0.2339**	**0.1523**	**0.1931**	**0.2137**

The best result is highlighted in bold.

8.7 Case Study

After quantitative evaluations, we turn to provide two case studies to illustrate the effectiveness and interpretability of R-JLMIA on person-job fit with representation disentanglement and skill recommendation for interview assessment, respectively.

8.7.1 Person-Job Fit with Representation Disentanglement. Taking the senior product manager position as an example, we show the word cloud of job descriptions and resumes and infer their topic distributions on corresponding semantic space based on the posterior of latent variables z^J, $z_c^R \oplus z_s^R$, respectively. Furthermore, in order to illustrate the degree of person-job fit, we further compared semantic information of the prior $z_c^{R'}$ and posterior z_c^R by presenting the topic distributions in resume space projected from them with the secondary variable $z_s^R = 0$, respectively. In addition, the capability to disentangle talents' core competence can be demonstrated by comparing the topic distributions generated from z_c^R and z_s^R by setting the other as 0, respectively.

All results can be found in Figure 9, where the top six keywords in each related topic are provided. We can find that this job position is mainly responsible for "product design" based on "effect evaluation" and "user feedback" and require "communication skills" and "innovation". Correspondingly, the top three topics are all about those with more related keywords, like "operation", "product planning", "user experience", "data analysis", and "competitive produce". For the resume, we find several experiences related to "product manager" for "mobile games". As a result, the top three topics are equipped with more keywords "planning", "user experience", "market", and "game". Meanwhile, according to Figure 9(e), we can note that topic distributions generated from the prior $z_c^{R'}$ and posterior z_c^R are similar with the same high possibility topics #32, #54, and #80. Those topics are all about the product manager from different aspects, like technological process, popularization, and user experience. Therefore, we can claim that the candidate is suitable for this job. In addition, from Figure 9(f), we find that R-JLMIA disentangles successfully the core competences related to the product manager with the secondary capability, i.e., experiences related to "mobile games".

8.7.2 Recommendation for Interview. Table 6 shows the example for an application to algorithm developer. We present the major job requirements, candidate's experiences, skill keywords investigated by interviewers, and the recommended skill keywords by R-JLMIA and LDA. From the results, we can find that recommended skills by R-JLMIA have similar focuses as the true assessment report in all three rounds. To be specific, for the first round, both true assessment report and recommended skills focus on the foundational requirement in the job description, where data

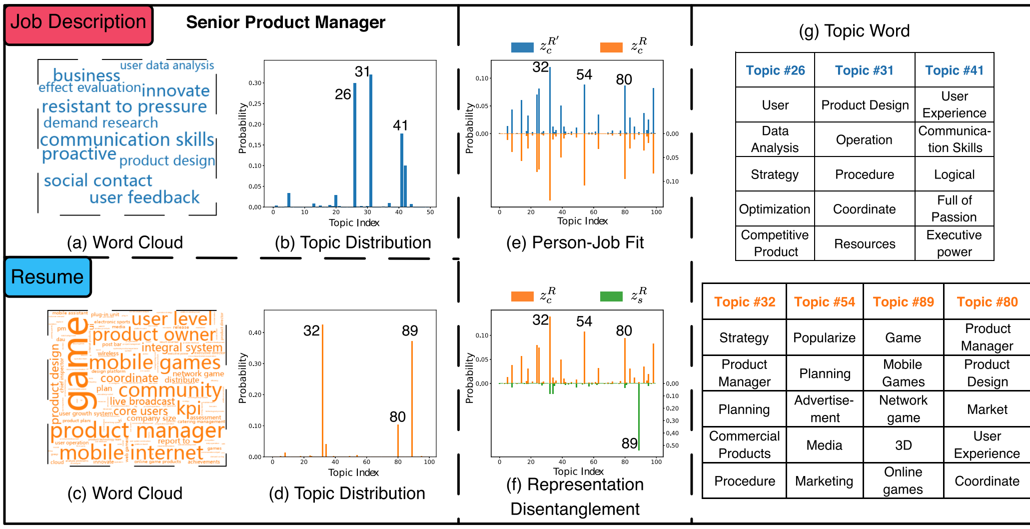


Fig. 9. The case study of person-job fit and representation disentanglement.

structure related keywords, such as "array" and "sorting", and "clear thinking" are involved in both them with keywords. However, the true assessment report also includes the design of algorithm with keywords, "dynamic programming", while programming related keywords, such as "Python", "C++", and "programming ability", are more involved in recommended skill keywords, By contrast, the second round focuses more on professional technology related to "machine learning" in both true assessment report and recommended skills, despite there exist multiple different keywords in them. For the third round, interviewers would like to provide a more comprehensive evaluation from the historical experience, such as "CTR estimation", to several personal qualities. It also has been captured by R-JLMIA with recommended keywords "work experience', "shortcoming", "cognitive", and so on. Finally, LDA tends to recommend similar skills for all three interview rounds (there exist several differences among the orders of keywords for different interview rounds). By contrast, R-JLMIA cannot only provide a more accurate recommendation, but also capture the evolution of topics among interview rounds with distant skill keywords in each round. In addition, note that interviewers may focus on one partial aspect during each interview round due to the limited time or interviewer's preferences. Therefore, we also highlighted other skills (blue) related to the job description or resume in the recommended list, which has been missed in the true interview report. We can find that R-JLMIA provides more reasonable and diverse skill keywords for interviewers.

9 CONCLUDING REMARKS

In this article, we first developed preliminary model, named JLMIA, to mine the relationship among job description, candidate resume, and interview assessment. Then, we further designed an enhanced model, named Neural-JLMIA, to improve the representative capability by applying neural variance inference. Last, we proposed to refine JLMIA with R-JLMIA by modeling individual characteristics for each collection, i.e., disentangling the core competence from resume and capturing the evolution of the semantic topics over different interview rounds. As a result, our approaches can effectively learn the representative perspectives of different job interview processes from the successful job interview records in history. Furthermore, we exploited our approaches for two real-world applications, namely person-job fit and skill recommendation for interview assessment.

Table 6. The Case Study of Skill Recommendation for Interview Assessment

Algorithm Developer	
Job Description	– Have a deep understanding of data structure and algorithm design, and familiar with C/C++ or Python programming.
	– Experience in NLP or machine learning is preferred.
	– Clear thinking, good communication skills, pressure resistance and team work ability.
Resume	□ Having experience in programming projects based on C/C++ and Python. . .
	□ Have experience in machine learning and deep learning, such as, intelligent scheduling of online car hailing, advertising system, Click-Through-Rate(CTR) estimation. . .
Skills in Interview Assessment	(1) Clear thinking, algorithm design, feature extraction, dynamic programming, optimization space, array, sorting.
	(2) Data mining, data preprocessing, feature selection, classification, clear thinking, stack, offline, python, anti cheating.
	(3) CTR estimation, offline, model training, team development, advertisement, executive power, shortcomings, cognitive ability.
Skills Recommended by R-JLMIA	(1) Python, C++, thread, array, programming ability, string, map, sorting, linked list, clear thinking.
	(2) data mining, recommender, clear thinking, Python, gbdt, NLP, C++, SVM, programming ability, classification.
	(3) Pressure resistance, responsibility, initiative, cognitive, work experience, shortcomings, initiative, clear thinking, values, project.
Skills Recommended by LDA	(1) Advertisement, hadoop, machine learning, Python, spark, recommend, hive, big data, data mining, sorting.
	(2) Advertisement, hadoop, machine learning, Python, spark, recommend, hive, big data, data mining, sorting.
	(3) Advertisement, hadoop, machine learning, Python, spark, recommend, hive, sorting, big data, data mining.

The correct keywords recommended are highlighted by red, while we also color other keywords related to job description and resume with blue. Skills recommended by R-JLMIA and LDA are sorted by their generative probability.

Extensive experiments conducted on real-world data clearly validate the effectiveness of JLMIA, which can lead to substantially less bias in job interviews and provide an interpretable understanding of job interview assessment.

APPENDIX

A EM ALGORITHM OF VARIATIONAL INFERENCE FOR JLMIA

In this Appendix, we give some details of the EM-style algorithm of variational inference outlined in Section 4.2

First of all, we define each variational distribution term of the variational families in Equation (3). To be specific, the variational distribution of each topic proportion vector $\theta^J_{m'}$ is Dirichlet parameterized by vector $\gamma^J_{m'}$. The variational distribution of $\theta^A_{m'd,k}$, the kth dimension of topic proportion vector $\theta^I_{m'd}$, is univariate Gaussians $\{\gamma^I_{m'd,k}, \delta^2\}$. The variational distribution of $c^J_{m'j}$, $c^R_{m'dr}$, and $c^A_{m'de}$ are specified by free Multinomial with parameters $\phi^J_{m'j,1:K}$, $\phi^R_{m'dr,1:CK}$ and $\phi^A_{m'de,1:CK}$

respectively. The variational distribution of φ_k^J, φ_k^R, and φ_k^A are Dirithlet parameterized by $\lambda_{k,1:|V^J|}^J$, $\lambda_{k,1:|V^R|}^R$, and $\lambda_{k,1:|V^E|}^A$, where $|V^J|$, $|V^R|$ and $|V^A|$ are the lengths of vocabularies of job description, resume and interview assessment, respectively.

Actually, we find each term of *ELBO* in JLMIA is similar to some parts of *ELBO* in LDA model [9] or CTM model [63], except $E_q[logp(\theta_{m'd}^I|\theta^J,\delta^2)]$, which can be computed by

$$E_q[log\ p(\theta_{m'd}^I|\theta^J,\delta^2)] = E_q[log\ N(\theta_{m'd}^I|h(\theta_{m'}^J,C),\delta^2 I)] =$$
$$-\frac{CK}{2}(log\ \delta^2 + log\ 2\pi) - \frac{1}{2\delta^2}\sum_{k=1}^{CK} E_q[(\theta_{m'd,k}^I - log\ \theta_{m',k'}^J)^2],$$
$$E_q[(\theta_{m'd,k}^I - log\ \theta_{m',k'}^J)^2] = \delta^2 + \Psi'(\gamma_{m',k'}^J) - \Psi'(|\gamma_{m',1:K}^J|)$$
$$+(\gamma_{m'd,k}^I - \Psi(\gamma_{m',k'}^J) + \Psi(|\gamma_{m',1:K}^J|))^2,$$

where we assume that $|\gamma_{m',1:K}^J| = \sum_{i=1}^K \gamma_{m',k}^J$, and $k' = k\ mod\ K$. Similar symbols are not described later for simplicity. And the $\Psi(\cdot)$ is Digamma function with derivative $\Psi'(\cdot)$.

Then, we describe our EM-style algorithm. In E-step, we employ coordinate ascent approach to optimize all variational parameters. First, we optimize the $\zeta_{m'd}$ in Equation (5):

$$\hat{\zeta}_{m'd} = \sum_{k=1}^{CK} exp\left\{\gamma_{m'd,k}^I + \delta^2/2\right\}.$$

Second, we optimize $\phi_{m'j,1:K}^J$, $\phi_{m'dr,1:CK}^R$ and $\phi_{m'de,1:CK}^A$ for each coordinate. Assume that $w_{m'j}^J = c$, $w_{m'dr}^R = t$ and $w_{m'de}^A = i$:

$$\hat{\phi}_{m'j,k}^J \propto exp\{\Psi(\lambda_{k,c}^J) - \Psi(|\lambda_{k,1:|V^J|}^J|) + \Psi(\gamma_{m',k}^J) - \Psi(|\gamma_{m',1:K}^J|)\},$$
$$\hat{\phi}_{m'dr,k}^R \propto exp\{\Psi(\lambda_{k,t}^R) - \Psi(|\lambda_{k,1:|V^R|}^R|) + \gamma_{m'd,k}^I\},$$
$$\hat{\phi}_{m'de,k}^A \propto exp\{\Psi(\lambda_{k,i}^A) - \Psi(|\lambda_{k,1:|V^A|}^A|) + \gamma_{m'd,k}^I\}.$$

Third, we optimize γ_m^J. Due to no analytic solution, we use Newton's method for each coordinate:

$$\frac{dELBO}{d\gamma_{m',i}^J} = -\frac{1}{2\delta^2}\sum_{d=1}^{D'_m}\sum_{k=1}^{CK}\Big(2(\Psi(\gamma_{m',k'}^J) - \Psi(|\gamma_{m',1:K}^J|) - \gamma_{m'd,k}^I)$$
$$\times(\delta_{k'}^i\Psi'(\gamma_{m',k'}^J) - \Psi'(|\gamma_{m',1:K}^J|)) + \delta_{k'}^i\Psi''(\gamma_{m',k'}^J) - \Psi''(|\gamma_{m',1:K}^J|)\Big)$$
$$+\sum_{k=1}^{K}(|\phi_{m'1:N_m^J,k}^J| + \alpha_k - \gamma_{m',k}^J)(\delta_k^i\Psi'(\gamma_{m',k}^J) - \Psi'(|\gamma_{m',1:K}^J|)),$$

where function $\delta_x^y = 1$, only if $x = y$, otherwise, $\delta_x^y = 0$.

Fourth, we optimize $\gamma_{m'd,1:CK}^I$. Due to no analytic solution, again, we use conjugate gradient algorithm with derivative:

$$\frac{dELBO}{d\gamma_{m'd,k}^I} = -\frac{1}{\delta^2}\left(\gamma_{m'd,k}^I - \Psi(\gamma_{m',k'}^J) + \Psi(|\gamma_{m',1:K}^J|)\right) + |\phi_{m'd1:N_{m'd}^R,k}^R|$$
$$+|\phi_{m'd1:N_{m'd}^A,k}^A| - (N_{m'd}^R + N_{m'd}^A)\zeta_{m'd}^{-1}exp\{\gamma_{m'd,k}^I + \delta^2/2\}.$$

Last, we optimize λ^J, λ^R, and λ^A. Their calculation process are similar, token $\lambda^J_{k,c}$ and $\lambda^A_{k,i}$ as examples:

$$\lambda^J_{k,c} = \beta^J_c + \sum_{m'=1}^{M'} \sum_{j=1}^{N^J_{m'}} \phi^J_{m'j,k} \delta^c_{w^J_{m'j}},$$

$$\lambda^A_{k,i} = \beta^A_i + \sum_{m'=1}^{M'} \sum_{d=1}^{D_{m'}} \sum_{e=1}^{N^A_{m'd}} \phi^J_{m'de,k} \delta^i_{w^A_{m'de}}.$$

In the M-step, we maximize the *ELBO* with respect to parameter α, similar to LDA, and regard the other hyper-parameters in Ω as fixed parameters.

Table A.1. The Performance Based on Precise@K of our Approaches and Baselines over Skill Recommendation for Interview Assessment

	1st Round			2nd Round			3rd Rund		
	P@5	P@10	P@15	P@5	P@10	P@15	P@5	P@10	P@15
Frequency	0.1773	0.1627	0.1632	0.1109	0.1407	0.1385	0.1773	0.1642	0.1390
LDA	0.2690	0.2338	0.2236	0.3118	0.2647	0.2355	0.2762	0.2223	0.1969
NTM	0.2996	0.2657	0.2454	0.2743	0.2444	0.2154	0.2317	0.2021	0.1829
BNVAE	0.3229	0.2776	0.2565	0.3294	0.2906	0.2662	0.2618	0.2249	0.2015
JLMIA	0.2835	0.2507	0.2357	0.3383	0.2747	0.2475	0.2510	0.2113	0.1907
Neural-JLMIA	0.2713	0.2323	0.2113	0.3068	0.2556	0.2335	0.2614	0.2221	0.2057
R-JLMIA w/o z^R_s	0.3903	0.3430	0.3076	0.3863	0.3282	0.2946	0.2845	0.2581	0.2353
R-JLMIA_a	0.3612	0.3241	0.2976	0.3264	0.2702	0.2484	0.2389	0.2117	0.1934
R-JLMIA_Linear	0.2050	0.2101	0.2061	0.3256	0.2825	0.2556	0.2750	0.2348	0.2153
R-JLMIA_GRU	0.4012	0.3477	0.3184	**0.3967**	0.3278	0.2956	0.3131	0.2651	0.2378
R-JLMIA	**0.4216**	**0.3574**	**0.3208**	0.3966	**0.3454**	**0.3058**	**0.3399**	**0.2793**	**0.2483**

The best result is highlighted in bold.

Table A.2. The Performance Based on Recall@K of our Approaches and Baselines over Skill Recommendation for Interview Assessment

	1st Round			2nd Round			3rd Rund		
	R@5	R@10	R@15	R@5	R@10	R@15	R@5	R@10	R@15
Frequency	0.0395	0.0734	0.1058	0.0223	0.0644	0.0988	0.0514	0.0905	0.1134
LDA	0.0763	0.1127	0.1466	0.0872	0.1238	0.1539	0.0822	0.1147	0.1371
NTM	0.0825	0.1206	0.1516	0.0732	0.1131	0.1383	0.0691	0.1025	0.1283
BNVAE	0.0888	0.1266	0.1621	0.0907	0.1338	0.1715	0.0782	0.1128	0.1380
JLMIA	0.0799	0.1235	0.1607	0.0936	0.1358	0.1730	0.0762	0.1153	0.1455
Neural-JLMIA	0.0766	0.1160	0.1501	0.0874	0.1288	0.1658	0.0838	0.1217	0.1584
R-JLMIA w/o z^R_s	0.0990	0.1504	0.1914	0.0973	0.1482	0.1886	0.0805	0.1366	0.1759
R-JLMIA_a	0.0932	0.1486	0.1890	0.0871	0.1319	0.1684	0.0746	0.1198	0.1589
R-JLMIA_Linear	0.0464	0.0925	0.1284	0.0906	0.1420	0.1799	0.0843	0.1306	0.1682
R-JLMIA_GRU	0.1040	0.1527	**0.1982**	**0.1059**	**0.1543**	**0.1933**	0.0933	0.1442	0.1832
R-JLMIA	**0.1067**	**0.1532**	0.1934	0.1018	0.1530	0.1894	**0.0981**	**0.1476**	**0.1875**

The best result is highlighted in bold.

REFERENCES

[1] Amr Ahmed and Eric P. Xing. 2012. Timeline: A dynamic hierarchical Dirichlet process model for recovering birth/death and evolution of topics in text stream. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*. 20–29.

[2] Hagai Attias. 2000. A variational baysian framework for graphical models. In *Advances in Neural Information Processing Systems*, S. Solla, T. Leen, and K. Müller (Eds.) 209–215.

[3] Matthew James Beal . 2003. *Variational Algorithms for Approximate Bayesian Inference*. University of London, London.

[4] Josh Bersin. 2013. *Corporate Recruiting Explodes: A New Breed of Service Providers*. Retrieved on from https://www.forbes.com/sites/joshbersin/2013/05/23/corporate-recruitment-transformed-new-breed-of-service-providers/.

[5] Shuqing Bian, Wayne Xin Zhao, Yang Song, Tao Zhang, and Ji-Rong Wen. 2019. Domain adaptation for person-job fit with transferable deep global match network. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 4812–4822.

[6] Nils Bjorck, Carla P. Gomes, Bart Selman, and Kilian Q. Weinberger. 2018. Understanding batch normalization. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.). 7694–7705.

[7] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American Statistical Association* 112, 518 (2017), 859–877.

[8] David M. Blei and John D. Lafferty. 2006. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*. ACM, New York, NY, 113–120. DOI: https://doi.org/10.1145/1143844.1143859

[9] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3 (Mar. 2003), 993–1022. Retrieved from http://dl.acm.org/citation.cfm?id=944919.944937.

[10] Rosalie P. Chamberlain. 2016. Five steps toward recognizing and mitigating bias in the interview and hiring process. *Strategic HR Review* 15, 5 (2016), 199–203.

[11] Jonathan Chang, Jordan Boyd-Graber, Chong Wang, Sean Gerrish, and David M. Blei. 2009. Reading tea leaves: How humans interpret topic models. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, Vol. 22. Citeseer, 288–296.

[12] Yu Cheng, Yusheng Xie, Zhengzhang Chen, Ankit Agrawal, Alok Choudhary, and Songtao Guo. 2013. Jobminer: A real-time system for mining job-related patterns from social media. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1450–1453.

[13] Zhiyong Cheng, Xiaojun Chang, Lei Zhu, Rose C. Kanjirathinkal, and Mohan Kankanhalli. 2019. MMALFM: Explainable recommendation by leveraging reviews and images. *ACM Transactions on Information Systems* 37, 2 (2019), 1–28.

[14] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *Workshop on Deep Learning, December (NIPS'14)*.

[15] Mamadou Diaby, Emmanuel Viennet, and Tristan Launay. 2013. Toward the next generation of recruitment tools: An online social network-based job recommender system. In *2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. IEEE, 821–828.

[16] Lan Du, Wray Buntine, and Huidong Jin. 2012. Modelling sequential text with an adaptive topic model. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. Association for Computational Linguistics, 535–545.

[17] Lan Du, Wray Buntine, Huidong Jin, and Changyou Chen. 2012. Sequential latent dirichlet allocation. *Knowledge and Information Systems* 31, 3 (2012), 475–503.

[18] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep sparse rectifier neural networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*. 315–323.

[19] Yoav Goldberg and Omer Levy. 2014. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv:1402.3722. Retrieved from https://arxiv.org/abs/1402.3722.

[20] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580. Retrieved from https://arxiv.org/abs/1207.0580.

[21] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9, 8 (1997), 1735–1780.

[22] Wenxing Hong, Siting Zheng, and Huan Wang. 2013. Dynamic user profile-based job recommender system. In *2013 8th International Conference on Computer Science & Education*. IEEE, 1499–1503.

[23] Bao-Xing Huai, Teng-Fei Bao, Heng-Shu Zhu, and Qi Liu. 2014. Topic modeling approach to named entity linking. *Journal of Software* 25, 9 (2014), 2076–2087.

[24] Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. 1999. An introduction to variational methods for graphical models. *Machine Learning* 37, 2 (1999), 183–233.

[25] Edgar E. Kausel, Satoris S. Culbertson, and Hector P. Madrid. 2016. Overconfidence in personnel selection: When and why unstructured interview information can hurt hiring decisions. *Organizational Behavior and Human Decision Processes* 137 (2016), 27–44.

[26] Diederik P. Kingma and Max Welling. 2013. Auto-encoding variational bayes. arXiv:1312.6114. Retrieved from https://arxiv.org/abs/1312.6114.

[27] Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M. Blei. 2017. Automatic differentiation variational inference. *The Journal of Machine Learning Research* 18, 1 (2017), 430–474.

[28] Danielle H. Lee and Peter Brusilovsky. 2007. Fighting information overflow with personalized comprehensive information access: A proactive job recommender. In *Third International Conference on Autonomic and Autonomous Systems*. IEEE, 21–21.

[29] Chenliang Li, Shiqian Chen, Jian Xing, Aixin Sun, and Zongyang Ma. 2018. Seed-guided topic model for document filtering and classification. *ACM Transactions on Information Systems* 37, 1 (2018), 1–37.

[30] Chenliang Li, Yu Duan, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. 2017. Enhancing topic modeling for short texts with auxiliary word embeddings. *ACM Transactions on Information Systems* 36, 2 (2017), 1–30.

[31] Huayu Li, Yong Ge, Hengshu Zhu, Hui Xiong, and Hongke Zhao. 2017. 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*. 917–925.

[32] Yingzhen Li and Stephan Mandt. 2018. Disentangled sequential autoencoder. In *Proceedings of the 35th International Conference on Machine Learning*.

[33] Hao Lin, Hengshu Zhu, Junjie Wu, Yuan Zuo, Chen Zhu, and Hui Xiong. 2020. Enhancing employer brand evaluation with collaborative topic regression models. *ACM Transactions on Information Systems* 38, 4 (2020), 1–33.

[34] Hao Lin, Hengshu Zhu, Yuan Zuo, Chen Zhu, Junjie Wu, and Hui Xiong. 2017. Collaborative company profiling: Insights from an employee's perspective. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence* 1417–1423.

[35] Wei Liu, Guoxi Cao, and Jianqin Yin. 2019. Bi-level attention model for sentiment analysis of short texts. *IEEE Access* 7 (2019), 119813–119822.

[36] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning*, Vol. 30. 3.

[37] Jochen Malinowski, Tobias Keim, Oliver Wendt, and Tim Weitzel. 2006. Matching people and jobs: A bilateral recommendation approach. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, Vol. 6. IEEE, 137c–137c.

[38] Qingxin Meng, Hengshu Zhu, Keli Xiao, Le Zhang, and Hui Xiong. 2019. A hierarchical career-path-aware neural network for job mobility prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 14–24.

[39] Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *Proceedings of the 34th International Conference on Machine Learning*. 2410–2419.

[40] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*. 3111–3119.

[41] David Mimno, Hanna M. Wallach, Jason Naradowsky, David A. Smith, and Andrew McCallum. 2009. Polylingual topic models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 880–889.

[42] Andriy Mnih and Karol Gregor. 2014. Neural variational inference and learning in belief networks. arXiv:1402.0030. Retrieved from https://arxiv.org/abs/1402.0030.

[43] Ioannis Paparrizos, B. Barla Cambazoglu, and Aristides Gionis. 2011. Machine learned job recommendation. In *Proceedings of the 5th ACM Conference on Recommender Systems*. ACM, 325–328.

[44] Shinjee Pyo, Eunhui Kim, et al. 2015. Lda-based unified topic modeling for similar tv user grouping and tv program recommendation. *IEEE Transactions on Cybernetics* 45, 8 (2015), 1476–1490.

[45] 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 *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 25–34.

[46] 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 Transactions on Information Systems* 38, 2 (2020), 1–33.

[47] 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*. 2165–2173.

[48] Rajesh Ranganath, Sean Gerrish, and David M. Blei. 2014. Black box variational inference. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*.

[49] Gąbor Rącz, Attila Sali, and Klaus Dieter Schewe. 2016. *Semantic Matching Strategies for Job Recruitment: A Comparison of New and Known Approaches*. Springer International Publishing. 149–168.

[50] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*. 1278–1286.

[51] Michal Rosen-Zvi, Chaitanya Chemudugunta, Thomas Griffiths, Padhraic Smyth, and Mark Steyvers. 2004. The author-topic models from text corpora. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*. 487–494.

[52] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. 2012. The author-topic model for authors and documents. arXiv:1207.4169. Retrieved from https://arxiv.org/abs/1207.4169.

[53] Tomoki Sekiguchi. 2004. Person-organization fit and person-job fit in employee selection: A review of the literature. *Osaka Keidai Ronshu* 54, 6 (2004), 179–196.

[54] Dazhong Shen, Chuan Qin, Chao Wang, Hengshu Zhu, Enhong Chen, and Hui Xiong. 2021. Regularizing variational autoencoder with diversity and uncertainty awareness. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

[55] Dazhong Shen, Hengshu Zhu, Chen Zhu, Tong Xu, Chao Ma, and Hui Xiong. 2018. A joint learning approach to intelligent job interview assessment. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 3542–3548.

[56] Baoxu Shi, Shan Li, Jae-Won Yang, Mustafa Emre Kazdagli, and Qi He. 2020. Learning to ask screening questions for job postings. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 549–558.

[57] Bradley Skaggs and Lise Getoor. 2014. Topic modeling for wikipedia link disambiguation. *ACM Transactions on Information Systems* 32, 3 (2014), 1–24.

[58] Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. arXiv:1703.01488. Retrieved from https://arxiv.org/abs/1703.01488.

[59] Ying Sun, Fuzhen Zhuang, Hengshu Zhu, Qing He, and Hui Xiong. 2021. Cost-effective and interpretable job skill recommendation with deep reinforcement learning. In *Proceedings of the Web Conference 2021*.

[60] Ying Sun, Fuzhen Zhuang, Hengshu Zhu, Qi Zhang, Qing He, and Hui Xiong. 2021. Market-oriented job skill valuation with cooperative composition neural network. *Nature Communications* 12, 1 (2021), 1–12.

[61] Mingfei Teng, Hengshu Zhu, Chuanren Liu, Chen Zhu, and Hui Xiong. 2019. Exploiting the contagious effect for employee turnover prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 1166–1173.

[62] Edwin N. Torres and Amy Gregory. 2018. Hiring manager's evaluations of asynchronous video interviews: The role of candidate competencies, aesthetics, and resume placement. *International Journal of Hospitality Management* 75 (2018), 86–93.

[63] Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, 448–456. DOI: https://doi.org/10.1145/2020408.2020480

[64] Chong Wang and David M. Blei. 2013. Variational inference in nonconjugate models. *Journal of Machine Learning Research* 14, Apr (2013), 1005–1031.

[65] Chao Wang, Hengshu Zhu, Qiming Hao, Keli Xiao, and Hui Xiong. 2021. Variable interval time sequence modeling for career trajectory prediction: Deep collaborative perspective. In *Proceedings of the Web Conference 2021*.

[66] Xuerui Wang and Andrew McCallum. 2006. Topics over time: A non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 424–433.

[67] Huang Xu, Zhiwen Yu, Jingyuan Yang, Hui Xiong, and Hengshu Zhu. 2016. Talent circle detection in job transition networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 655–664.

[68] Tong Xu, Hengshu Zhu, Chen Zhu, Pan Li, and Hui Xiong. 2018. Measuring the popularity of job skills in recruitment market: A multi-criteria approach. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*

[69] Hongzhi Yin, Bin Cui, Ling Chen, Zhiting Hu, and Xiaofang Zhou. 2015. Dynamic user modeling in social media systems. *ACM Transactions on Information Systems* 33, 3 (2015), 1–44.

[70] Jichuan Zeng, Jing Li, Yan Song, Cuiyun Gao, Michael R Lyu, and Irwin King. 2018. Topic memory networks for short text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3120–3131.

[71] Le Zhang, Tong Xu, Hengshu Zhu, Chuan Qin, Qingxin Meng, Hui Xiong, and Enhong Chen. 2020. Large-scale talent flow embedding for company competitive analysis. In *Proceedings of the Web Conference 2020*. 2354–2364.

[72] Le Zhang, Ding Zhou, Hengshu Zhu, Tong Xu, Rui Zha, Enhong Chen, and Hui Xiong. 2021. Attentive heterogeneous graph embedding for job mobility prediction. In *Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[73] Qi Zhang, Hengshu Zhu, Ying Sun, Hao Liu, Fuzhen Zhuang, and Hui Xiong. 2021. Talent demand forecasting with attentive neural sequential model. In *Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

[74] Yingya Zhang, Cheng Yang, and Zhixiang Niu. 2014. A research of job recommendation system based on collaborative filtering. In *2014 7th International Symposium on Computational Intelligence and Design*, Vol. 1. IEEE, 533–538.

[75] Wayne Xin Zhao, Wenhui Zhang, Yulan He, Xing Xie, and Ji-Rong Wen. 2018. Automatically learning topics and difficulty levels of problems in online judge systems. *ACM Transactions on Information Systems* 36, 3 (2018), 1–33.

[76] Chen Zhu, Hengshu Zhu, Hui Xiong, Pengliang Ding, and Fang Xie. 2016. Recruitment market trend analysis with sequential latent variable models. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 383–392.

[77] 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* 9, 3 (2018), 1–17.

[78] Qile Zhu, Wei Bi, Xiaojiang Liu, Xiyao Ma, Xiaolin Li, and D. Wu. 2020. A batch normalized inference network keeps the KL vanishing away. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.*

Received November 2020; revised April 2021; accepted June 2021