ResumeGAN: An Optimized Deep Representation Learning Framework for Talent-Job Fit via Adversarial Learning

Yong Luo∗
Nanyang Technological University, Singapore
yluo180@gmail.com

Huaizheng Zhang∗
Nanyang Technological University, Singapore
huaizhen001@e.ntu.edu.sg

Yonggang Wen
Nanyang Technological University, Singapore
ygwen@ntu.edu.sg

Xinwen Zhang
Hiretual, Mountain View, California, USA
xinwenzhang@hiretual.com

ABSTRACT

Nowadays, it is popular to utilize online recruitment services for talent recruitment and job recommendation. Given the vast amounts of online talent profiles and job-posts, it is labor-intensive and exhausted for recruiters to manually select only a few potential candidates for further consideration, and also nontrivial for talents to find the most matched job positions. Recently, some deep learning-based approaches are developed to automatically matching the talent resumes and job requirements, and have achieved encouraging performance. In this paper, we propose a novel framework that targets the same task, but integrate different types of information in a more sophisticated way and introduce adversarial learning to learn more expressive representation. In addition, we build a dataset for model evaluation and the effectiveness of our framework is demonstrated by extensive experiments.

CCS CONCEPTS

• **Computing methodologies** → **Natural language processing**; **Neural networks**; *Information extraction*; *Learning latent representations*.

KEYWORDS

Resume job-post fit; talent recruitment; adversarial learning; neural network; text mining

ACM Reference Format:
Yong Luo∗, Huaizheng Zhang∗, Yonggang Wen, and Xinwen Zhang. 2019. ResumeGAN: An Optimized Deep Representation Learning Framework for Talent-Job Fit via Adversarial Learning. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3357384.3357899

∗Yong Luo and Huaizheng Zhang contributed equally to this work.

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11...$15.00
https://doi.org/10.1145/3357384.3357899

1 INTRODUCTION

In this era of intelligent knowledge management, there have been dozens of online employment websites, such as "LinkedIn", "Monster" and "Indeed", which can significantly facilitate the job search for talents and also talent recruitment for human resource managers. There are large amounts of users, resumes and job-posts on these platforms and the number may increase sharply. For example, in 2019, there have been more than 630 million users and 20 million open jobs on "LinkedIn" (https://news.linkedin.com/about-us#statistics), and there are on average 29 resumes uploaded and 7,900 jobs searched on "Monster" every minute (https://www.monster.com/about). Given the vast amounts of data, it is often costly (e.g., on average 42 days and $4,129 in 2016 [22]) for a company or organization to find an appropriate employee. Besides, it may be also labor-intensive and time-consuming for talent to find an appropriate position for her/him, and some perfect job opportunities may be missed.

Therefore, it is desirable to develop a system that is able to accurately match or assign appropriate matching scores for the different pairs of resume and job requirement. Existing solutions for this task usually rely on domain expert features and suffer from several drawbacks such as inaccurate, inefficient and subjective. Some recent works utilize the deep learning and text mining techniques to automatically learn representations for the resumes and job requirements [22, 23]. However, only the experiences of the talents are exploited in [22], and the work [23] mainly focuses on learning representation for sparse entities (such as skill id).

In this paper, we propose an adversarial learning based framework termed ResumeGAN to learn more expressive representations for the resumes and job-posts and thus lead to a better model for resume job-post fit. In particular, we first utilize ELMo [20], a recently proposed contextualized word embedding technique to represent each word/phrase in both the resumes and job-posts. Then attention [29] or hierarchical attention schemes [31] are incorporated to take the importance of different words or sentences into consideration. In the traditional resume job-post fit approaches, such as [22, 23], the talent's experiences and skills are often utilized separately. Different from these approaches, we believe that the consistency between experiences and skills is critical in discriminating good resumes from the bad ones [13], and thus we propose to exploit the consistency and utilize it as additional feature to improve the performance. More importantly, we introduce the adversarial learning mechanism [14] to impose some prior on the learned representation, and hence further improve the generalization and expressive ability.

Since there are still no public datasets for resume job-post fit, we build a new dataset to verify effectiveness of our method. The data are provided by a private online recruitment service company, and there are around 800 job-posts and $32K$ resumes together with a binary fit value for each pair of resume and job-post. Extensive experiments are conducted on this dataset by comparing with several baselines and some recently proposed approaches [22, 23]. The results demonstrate that our method is superior to all other approaches in terms of various evaluation criteria including accuracy, recall, precision and F1-measure.

The main contributions of this paper are summarized as follows:

- We design a novel network architecture of deep representation learning for resume job-post matching, where the consistency between talent's experiences and skills are exploited.
- We introduce the adversarial learning mechanism to further improve expressive ability of the representation.
- We collect large amounts of data and conduct extensive experiments for model verification.

The rest of this paper is organized as follows. Section 2 is a summarization of some closely related works. In Section 3, we provide an investigation of the collected data and then a novel resume job-post fit framework is presented in Section 4 based on the investigation. Section 5 includes some empirical comparisons and analysis, and finally we conclude this paper in Section 6.

2 RELATED WORK

This section is a summarization of related works on resume job-post matching, text representation learning and adversarial learning.

2.1 Resume Job-post Matching

The matching between talent's profile and recruiter's job description has been studied for decades and some early works include [12, 15], where the matching problem is regarded as a recommendation problem. In [4], the talent's personal information in social network, such as interaction and social connection data, are exploited for job recommendation. Various collaborative filtering algorithms are compared in [33] and different types of information, such as talent's used-liked jobs are considered in the job recommendation system. The talent circles are extracted and identified for talent recruitment in [28] and the scalability bottleneck of the generalized linear mixed models (GLMix) is tackled in [32] for job recommendation. In these approaches, hand-crafted features are utilized and traditional recommendation models are adopted. A latent variable model is developed in [24] for person-job fit by jointly modeling the resume, job-post and job interview information. A literature survey of some job recommender systems can be found in [1, 25], and there is an empirical study presented in [7].

More recent works focus on utilizing the text embedding and deep neural networks to automatically learn useful features/representations for better resume job-post fit. For example, a convolutional neural network (CNN) based model is presented in [34], where the designed hierarchical representation structure enables the interpretability of the model, which is further improved in [22] by utilizing recurrent neural network (RNN) to exploit the sequential dependence between words and introducing the hierarchical attention scheme to improve interpretability. The researchers in LinkedIn also develop a method to learn representations for talent search and recommendation, especially the sparse entities in the talent's profile, such as skill ids [23]. Our method is superior to all of these approaches in that we learn the representation in a more sophisticated way, exploit the consistency between talent's experiences and skills, as well as introducing the adversarial learning mechanism to improve the expressive ability of the representation.

2.2 Text Representation Learning

Traditional text analysis methods usually employ the TF-IDF (term-frequency and inverse-document frequency) [30] to represent the words. The resulting representation thus may be very sparse and has very high dimension, which is prone to over-fitting. To tackle these issues, the text embedding technique is developed, which aims to learn compact and continuous representation for words or phrases [2, 18, 26] and sometimes sentences [3], where the neural networks are often utilized for learning. Some representative works include word2vec [17] and Glove [18], and we refer to [26] for a literature survey of some popular word/phrase embedding approaches.

Most of previous embedding methods do not take the linguistic contexts into consideration, and more recent approaches try to overcome this drawback [16, 19, 20]. For example, a contextualized word representation approach is presented in [19] based on bidirectional language model, and the supervised neural machine translation is adopted in [16] to learn contextualized word vectors. ELMo [20] combines the internal states of bidirectional LSTM and is superior to both of these two approaches in that the representations are deep. Therefore, we adopt ELMo to map the words/phrases into continuous representations in this paper.

2.3 Adversarial Learning

Adversarial learning has become a very hot and interesting topic in recent years since generative adversarial nets (GAN) [5] was proposed. GAN is able to learn sophisticated generative model without using approximate inference, and has been widely utilized for textual, audio and visual data generation. Inspired by the idea of generating adversarial examples

Recruiter job-post

- **Major**: computer science.
- **Title**: computer scientist, research programmer, scientific programmer, computational scientist.
- **Education level**: Ph.D.
- **Skill**: machine learning, algorithms, research.

Talent resume (+)

- **Major**: computer science.
- **Education degree**: Ph.D.
- **Skill**: machine learning, algorithms, computer science, python, network security, C++, research, Linux, artificial intelligence, data mining, information security, ···.
- **Publication**: machine learning journal, volume 81, issue 2, special issue on adversarial learning, ···.
- **Career path**: computer & network security.
- **Focus**: security engineer.
- **Highlight**: motivated researcher with nearly ten years of research experience in machine learning, network security, and their intersections; ···.
- **Experience**: I work on applying machine learning to security problems; Analyzed large-scale netflow data collected from a system of sensors distributed across the world; Performed data-mining and analysis to provide guidance on the usefulness of the dataset for intelligence-related uses; ···.

Talent resume (-)

- **Major**: applied management and decision sciences.
- **Education degree**: Ph.D.
- **Skill**: Unix, system administration, databases, security, program management, operating systems, public speaking, team leadership, computer infrastructure, web development, ···.
- **Publication**: null.
- **Career path**: information technology & services.
- **Focus**: education, system administrator.
- **Highlight**: award winning team player, project team lead. Programmer in Java, SQL, C#, C++ and other languages; Experienced classroom instructor with over 10 years experience of IT/IS at college level; ···.
- **Experience**: managed Cyber Lync and Learn program; taught Java II on ground sessions for Spring 2018 term; Facilitated dozens of 5 week long college courses in Operating Systems; Lead a team of web application developers using Microsoft; Project Manager for Enterprise Wide Backup Project; ···.

Figure 1: An example of job-post together with its matched (+) and mismatched (-) resumes. The job-post usually consists of some keywords/phrases in the search of talents by recruiters. The resume often has much more information, which can be grouped as three main categories: experiences, skills and other factors.

[6], a large number of methods are developed to improve the robustness of the model by simultaneously applying perturbations to examples to maximally fool the model and making the model to predict correctly on these harnessing examples.

Another important direction of utilizing GAN is to combine it with autoencoder [14] so that we can impose arbitrary prior on the hidden representation of deep generative model. This can be used to disentangle style and content, and the imposed prior can also be served as a regularization term to improve the generalization ability of the model, similar to variational autoencoder [11]. In this paper, we utilize the adversarial autoencoder to make the learned representation follow some prior distribution so that the expressive ability of the representation can be improved.

3 AN INVESTIGATION OF THE DATA

In this paper, we aim to conduct automatic resume job-post fit by fully exploiting the information contained in the talent's resume and the recruiter's job-post. An example of our collected job-post and its matched or mismatched resume are shown in Figure 1. In an online recruitment system, the recruiter usually filters large amounts of resumes by using some mandatory requirements, such as the major, education level and the number of working years. Then the match is conducted in a relatively small subset of the resumes by using the other requirements, such as job titles and required skills. The talent resume often contains more descriptions, e.g., the skills, working experiences, the talent's major focus and a highlight of her/his abilities. We divide the descriptions of resume into three major groups:

- **Experiences**, which usually consist of multiple sentences, i.e., $\{x^R_{11}, \cdots, x^R_{1L^e}\}$.
- **Skills**, which are often some separate words/phrases, i.e., $\{x^R_{21}, \cdots, x^R_{2L^s}\}$.
- **Other factors**, such as the talent's focused field or topic x^R_{31} and a short highlight description $x^R_{3L^o}$ of the talent.

The job-post often consists of some individual words/phrases $\{x^J_1, \cdots, x^J_M\}$ that describe the requirements of the job.

4 THE PROPOSED FRAMEWORK

Based on these observations and investigations, we develop the following framework for resume job-post fit. The network architecture of the proposed framework is shown in Figure 2. In particular, for all the words/phrases in the resume or job-post, we first utilize the ELMo [20] embedding technique to map them into continuous and compact representations. Figure 3 is an illustration of the similarity scores between the mapped embeddings. From the results, we can see that the words that are close to each other semantically tend to have high similarity score, such as ("deep learning", "machine learning"), ("C++", "C#") and ("engineering", "software").

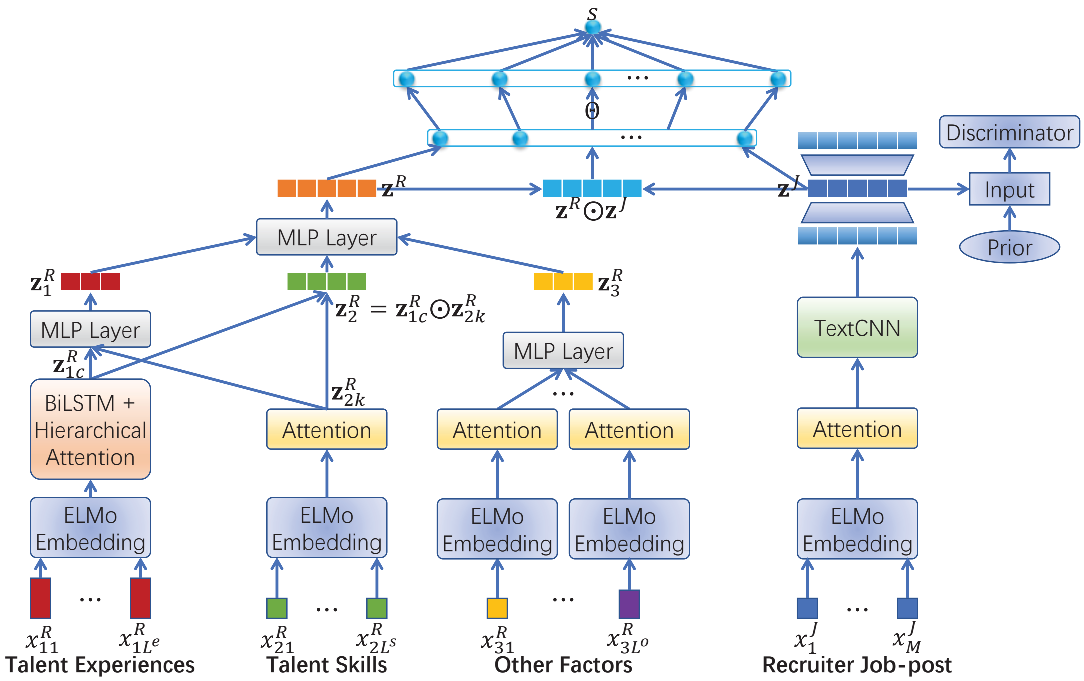

Figure 2: Network architecture of the proposed adversarial learning based resume job-post fit framework. The ELMo embedding [20] technique is utilized to map all the words/phrases in talent's resume and recruiter's job-post into continuous representation. Then hierarchical attention [31] with BiLSTM is utilized to learn aggregated representation for resume experiences. The high-level representations of resume skills and job-post are obtained by applying the attention scheme and CNN (for text processing) [10]. The consistency between talent's experiences and skills are exploited by calculate element-wise product of their high-level representations, and the adversarial learning mechanism is introduced to improve the expressive ability of the job-post representation. Finally, a concatenation of the combined resume representation and improved job-post representation and their element-wise product are utilized for prediction, where the logistic loss is adopted.

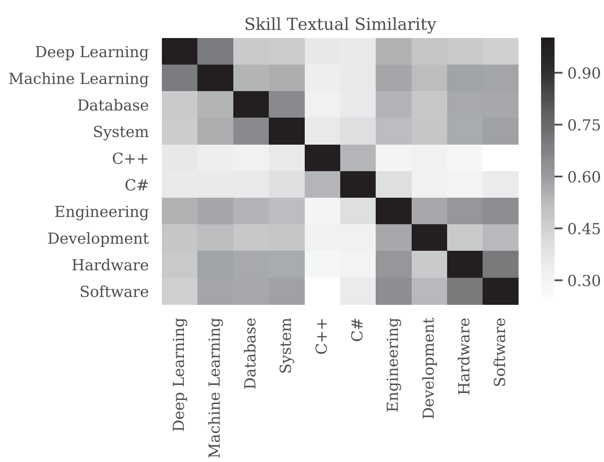

Figure 3: Similarity scores of the mapped embeddings by ELMo [20]. The words/phrases that are semantically close tend to have high similarity scores.

4.1 Experience Representation

Then we adopt the hierarchical attention technique developed in [31] to learn a unified representation $\mathbf{z}_{1c}^R$ for all the talent's experience embeddings. We suppose there are L^e experiences (sentences), where the l-th experience consists of T_l words $\{w_{lt}\}_{t=1}^{T_l}$, and the embedding of the word w_{lt} is $\mathbf{e}_{lt}$. Then the representation of the l-th experience is given by

$$\mathbf{h}_l = \sum_t \alpha_t \mathbf{h}_{lt}, \qquad (1)$$

where $\mathbf{h}_{lt} = \text{BiLSTM}(\mathbf{e}_{lt})$ is the output of bidirectional L-STM [9], and α_t is a weight determined by the attention scheme [29]. The representations of different experiences are aggregated in a similar way, i.e., $\mathbf{z}_{1c}^R = \sum_l \alpha_l \mathbf{h}_l$. More details can be found in [31], and multi-layer perception (MLP) is added to obtain high-level and comparable representation.

4.2 Skill Representation

For the talent's skills, an attention layer is utilized to assign larger weights to more important skills. In particular, for a input of L^s skills with the embeddings $\{\mathbf{e}_l\}_{l=1}^{L^s}$, the resulting representation is $\mathbf{z}_{2k}^R = \sum_l \beta_l \mathbf{e}_l$, where β_l is the weight generated by the attention scheme.

4.3 Representation Combination

A natural way to combine the resulting experience and skill representations is to concatenate them. But there is a lack of interactions between the experience and skill by simply concatenation and the dimension of the concatenated representation may be high. Therefore, we add some MLP layers to map the concatenated representation into low-dimensional common representation $\mathbf{z}_1^R$, i.e.,

$$\mathbf{z}_1^R = f([\mathbf{z}_{1c}^R; \mathbf{z}_{2k}^R]; \Theta_1^R), \tag{2}$$

where Θ_1^R is the set of parameters in the MLP layers. In this paper, we choose the activation function to be hyperbolic tangent ("tanh") since it is often a better choice than "sigmoid", which may suffer from saturation, and we found that "tanh" outperforms "ReLU" in our experiments.

Besides, we believe that the consistency between the experience and skill is critical in discriminating good resumes from bad ones. Thus the consistency is also useful in resume job-post fit since if a resume is not good enough, it will have low probability to be selected, no matter whether the experiences or skills match the job requirements or not. To evaluate the consistency between the experience and skill, we calculate the element-wise product [8] of their representations, i.e.,

$$\mathbf{z}_2^R = \mathbf{z}_{1c}^R \odot \mathbf{z}_{2k}^R. \tag{3}$$

We do not calculate the cosine similarity score since such a single value is not comparable to other vector representations, which may dominate the performance and thus the calculated similarity score has little contribution in the final prediction.

4.4 Job-post Representation

In regard to the job-post, we pass the embeddings of job requirements through a convolutional neural network (CNN) for text [10], which is termed "TextCNN" to obtain high-level representation. In this paper, we propose to add an attention layer before "TextCNN" to identify important job requirements. In particular, for a input of M job requirements with the weighted embeddings $\{\theta_l \mathbf{e}_l\}_{l=1}^{M}$, where θ_l is the weight generated by the attention scheme, a filter $\mathbf{w}_i$ is applied to extract a single-value feature z_i by conducting convolution and max-over-time pooling operations. The resulting representation is obtained by utilizing multiple filters with varying window sizes to derive a vector of features, i.e., $[z_1, \cdots, z_I]^T$. One advantage of utilizing CNN is that the number of parameters can be much fewer than multi-layer perception (MLP) and thus reduce the model complexity.

In addition, we introduce the adversarial learning mechanism to further improve the expressiveness of the representation. This is performed by adding an adversarial autoencoder on the output of the "TextCNN" layer. In particular, autoencoder is applied to learn the hidden representation $\mathbf{z}^J$ by minimizing the reconstruction error. In the learning phase, a regularization term is added to match the aggregated posterior distribution $q(\mathbf{z}^J)$ with an arbitrary prior $p(\mathbf{z})$. The formulation is given by

$$\epsilon(\Theta_{\mathrm{AE}}) = \min_{\Theta_{\mathrm{AE}}} \max_{D} \mathcal{L}(\mathbf{z}^J; \Theta_{\mathrm{AE}}) + \gamma \mathcal{R}_{\mathrm{GAN}}(\mathbf{z}^J; D), \tag{4}$$

where $\gamma \geq 0$ is a trade-off hyper-parameter and

$$\mathcal{R}_{\mathrm{GAN}}(\mathbf{z}^J; D) = \mathbb{E}_{\mathbf{z}\sim p(\mathbf{z})}[\log D(\mathbf{z})] + \mathbb{E}[\log(1 - D(\mathbf{z}^J))]. \tag{5}$$

Here, D is a discriminator that aims to tell apart the hidden code $\mathbf{z}$ sampled from the prior from the generated code $\mathbf{z}^J$. In our implementation, we choose the prior to be a Laplace distribution, where the probability density function is given by:

$$g(x; \mu, \lambda) = \frac{1}{2\lambda} \exp(-\frac{|x - \mu|}{\lambda}). \tag{6}$$

Here, the parameter μ is the distribution peak and λ is the exponential decay. The encoder consists of some MLP layers with ReLU activation or batch normalization, and the discriminator is constructed using some MLP layers together with a sigmoid based loss.

4.5 Other Factors Representation and Final Prediction

For the other entries in the resume, we first apply the attention scheme to their embeddings to assign larger weights to more important words/phrases, and then aggregate the results. Some MLP layers are added to find the low-dimensional common representation $\mathbf{z}_3^R$.

After obtaining all the representations $\{\mathbf{z}_1^R, \mathbf{z}_2^R, \mathbf{z}_3^R\}$ of resume, we concatenate them and using some MLP layers to map the resulting vector as a common representation $\mathbf{z}^R$, which has the same dimension as $\mathbf{z}^J$. Finally, we concatenate $\mathbf{z}^R$ and $\mathbf{z}^J$ as well as their element-wise product for prediction, where additional MLP layers are utilized together with the logistic loss, i.e.,

$$\epsilon(\Theta) = \mathrm{Sigmoid}(f([\mathbf{z}^R; \mathbf{z}^J; \mathbf{z}^R \odot \mathbf{z}^J]; \Theta), y), \tag{7}$$

where Θ denotes the set of parameters to be learned and $y = \{0, 1\}$ is the groundtruth label, which indicates whether a pair of resume and job-post fits or not decided by experts (such as the human resource managers).

5 EXPERIMENTS

In this section, we first provide a detailed description of the collected data for resume job-post matching, and then verify the effectiveness of our method by comparing it with several baselines and some recently proposed approaches [22, 23].

5.1 The Collected Data

We conduct experiments on a self-collected real-world dataset, which is built by utilizing the information provided by a private online recruitment company. All the secret personal information are removed to protect user privacy. The dataset consists of 811 job-posts and 32,082 resumes in total. A recruiter may have several job-posts, each of them usually has multiple good-fit and bad-fit resumes. The same resume of

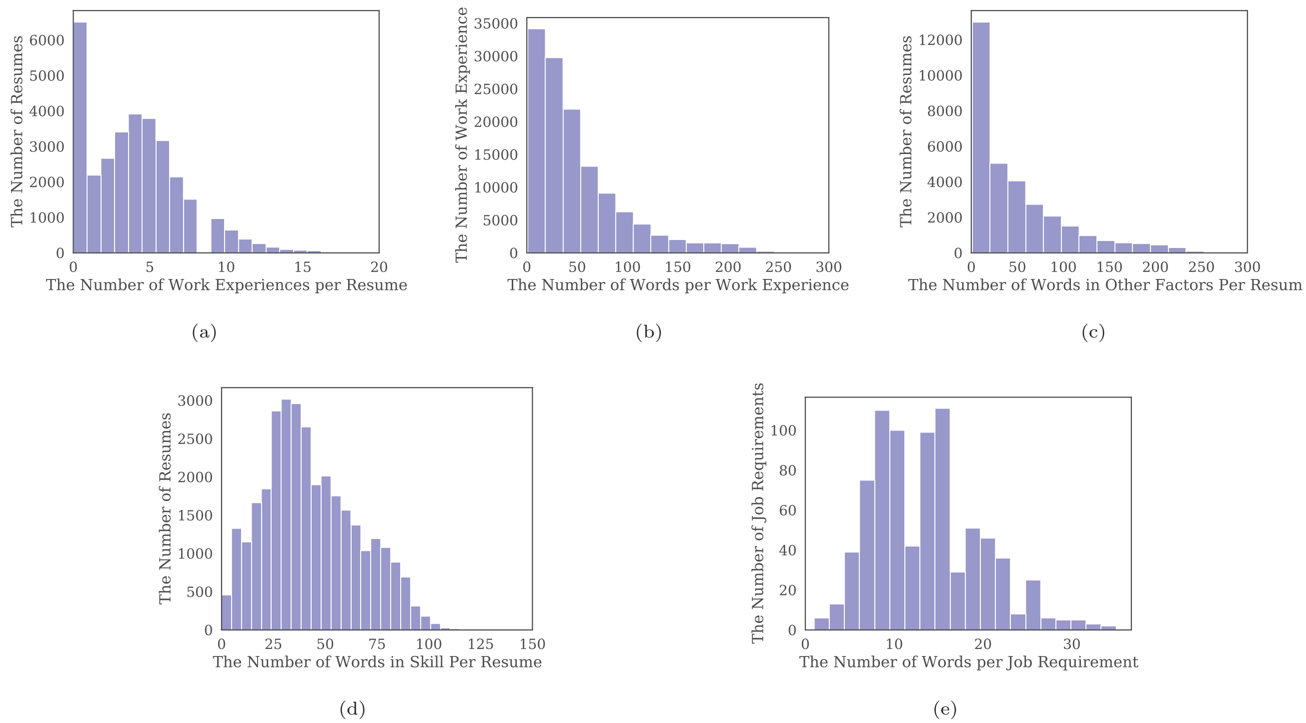

Figure 4: Distribution of the information contained in our self-collected dataset.

a talent may be labeled as good-fit by some recruiters, but bad-fit by others. This leads to a total of 23, 530 positive and 15, 194 negative resume job-post pairs. We randomly select 80% for training, 10% for validation and the remained 10% for test. The distribution and statistics of the information contained in our dataset are shown in Figure 4 and Table 1 respectively. From the results, we can see that there are many resumes that do not have any experiences and thus it is necessary to also utilize other information for prediction. Besides, the average numbers of words in resume's experiences, skills and other factors are comparable and much more than that in recruiter's job-post. Therefore, more text processing operations should be performed on the resume to extract useful information, and this is consistent with our framework.

Table 1: Statistics of the information contained in our self-collected dataset.

Statistics	Values
#{Job-posts}	811
#{Resumes}	32, 082
#{Positive Pairs}	23, 530
#{Negative Pairs}	15, 194
Average Words Per Job Requirement	13.543
Average Work Experience Per Resume	4.090
Average Words Per Work Experience	49.691
Average Skill Words Per Resume	43.212
Average Other Feature Words Per Resume	49.460

5.2 Experimental Setup

We first compare with several baseline approaches, i.e., applying standard supervised learning algorithms, such as logistic regression, AdaBoost and random forest as well as gradient boosting decision tree (GBDT) to a concatenation of all the embeddings of words/phrases in the input resume and corresponding job-post. Then we compare the following approaches:

- **LinkedIn [23]**: a recently proposed talent search approach developed by LinkedIn researchers, where LinkedIn Economic Graph [27] is introduced to learn semantic representations for sparse entities in resume or job-post.
- **APJFNN [22]**: a competitive resume job-post fit approach that is based on hierarchical attention scheme.
- **ResumeGAN**: The proposed resume job-post matching method, where the dimensions of the hidden layer representations $\mathbf{z}^R$ and $\mathbf{z}^J$ are empirically set as 512.

The performance are evaluated using a variety of popular criteria including accuracy, precision, F1-measure and area under the ROC curve (AUC) [21].

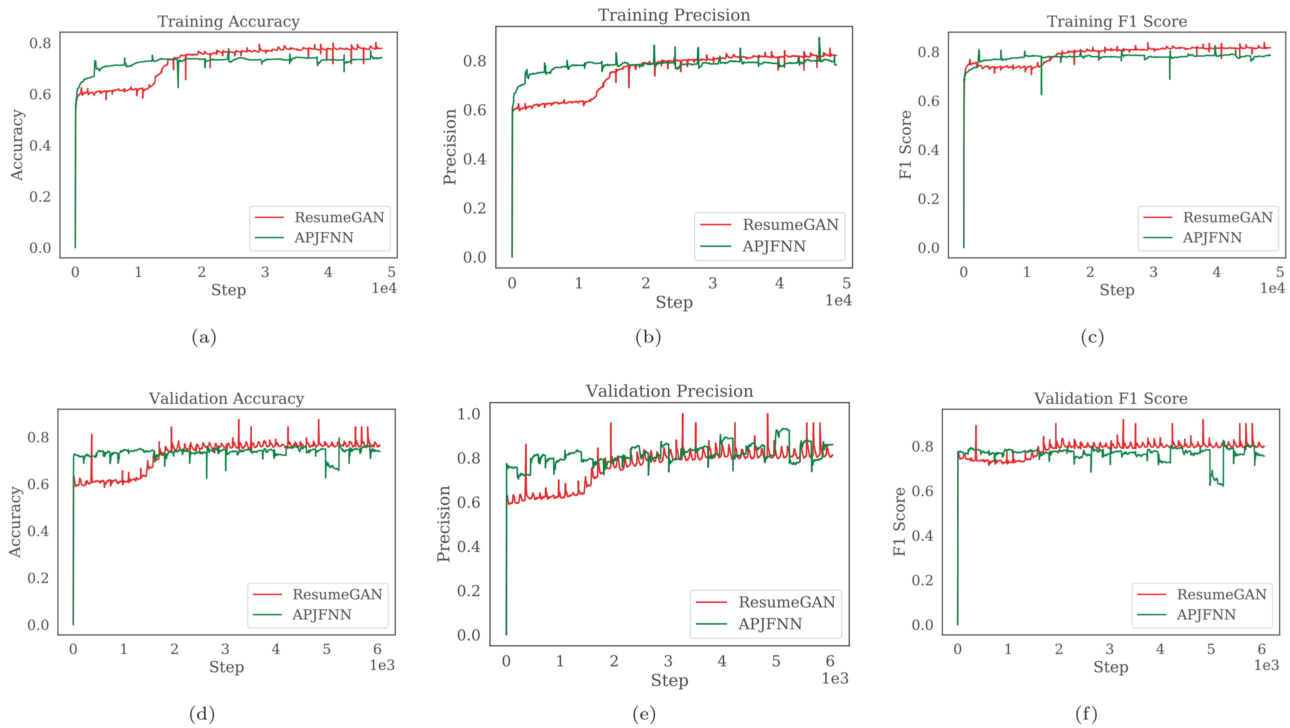

Figure 6: The training and validation performance in different epochs of our method and the competitive APJFNN [22]. The proposed method outperforms APJFNN in most cases of both the training and validation phases.

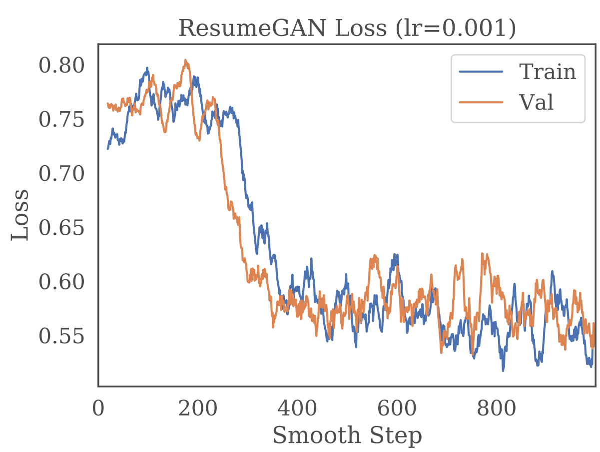

Figure 5: The training and validation losses of our method.

5.3 Experimental Results

In the following, we first show the convergence of our method, and then present the main comparison results. Finally, some analysis of the important hyper-parameters is provided.

5.3.1 Convergence Analysis. We show the training and validation losses w.r.t. different epochs in Figure 5. It can be seen from the results that: 1) the losses have some oscillations at first and then drop sharply to much lower values. This indicates that our method can converge using only a small number of epochs and thus the training process is efficient; 2) the training and validation process have similar trends. This demonstrates that over-fitting is appropriately alleviated in our method.

5.3.2 A Comparison with Other Approaches. The training and validation performance of our method and the competitive APJFNN are shown in Figure 6. From the results, we can see that the proposed method outperforms APJFNN in both the training and validation process, especially under the accuracy and F1-measure criteria.

The test performance of different approaches are reported in Table 2 and Figure 7. From the results, we observe that: 1) in the traditional algorithms, both random forest and GBDT are very competitive; 2) the recently proposed approaches that utilize sophisticated deep learning and text mining techniques can always achieve better results than the traditional algorithms and our method outperforms all other approaches consistently, even without the adversarial learning procedure. This demonstrates the effectiveness of the designed network

Table 2: The performance in terms of different criteria of the compared approaches. Ours (w/o GAN) is the our method without adversarial learning.

Method	Accuracy	F1	AP
Logistic Regression	0.725	0.786	0.757
Random Forest	0.726	0.770	0.798
AdaBoost	0.711	0.771	0.757
Decision Tree	0.671	0.726	0.750
GBDT	0.727	0.780	0.776
LinkedIn [23]	0.737	0.782	0.778
APJFNN [22]	0.759	0.807	0.812
Ours (w/o GAN)	0.771	0.809	0.814
ResumeGAN	**0.783**	**0.820**	**0.836**

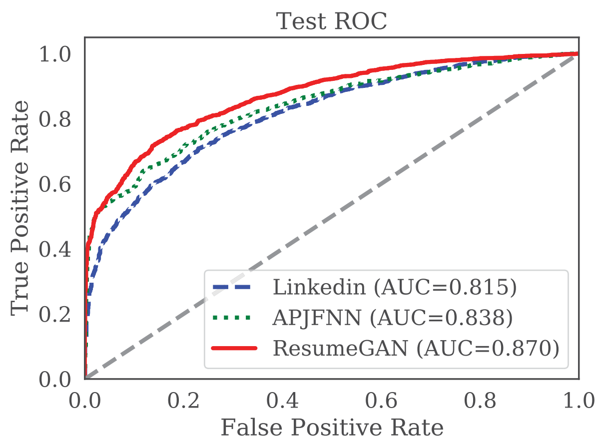

Figure 7: The test ROC curves and AUC scores of different approaches.

architecture; 3) by introducing the adversarial learning mechanism, the performance can be further improved, in terms of all different criteria. This demonstrates the advantage of utilizing the adversarial learning strategy.

5.3.3 Hyper-parameter Analysis. Finally, we conduct analysis for some important hyper-parameters. For example, the output of our model is a real-valued score, and we need to apply a threshold to map the resulting score into a binary label, which indicates that the test resume and job-post match or not. The performance of our method w.r.t. different threshold values are shown in Figure 8. From the results, we can see that the optimal performance are achieved around 0.5 under the different evaluation criteria, and hence we set the threshold to be 0.5 in our method. Figure 9 is the confusion matrix at the threshold 0.5 in our method. It can be observed from the figure that our method performs well in predicting either good or bad matches. This demonstrates the reliability of our method.

In Figure 10, we show the performance of our method w.r.t. different learning rates. It can be observed from the results that our method achieves much steady performance in a wide range of learning rate, e.g., from 0.0005 to 0.01. This indicates that our method is insensitive to the learning rate. The best performance is achieved at 0.001 under all different evaluation criteria. Thus we adopt a fixed learning rate 0.001 in this paper and intend to design adaptive learning rate to further improve the performance in the future.

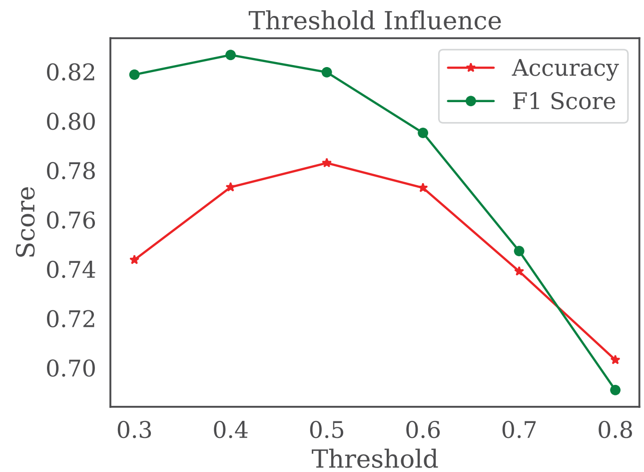

Figure 8: The performance of our method w.r.t. varied threshold, which is utilized to map the output real-valued score to binary label.

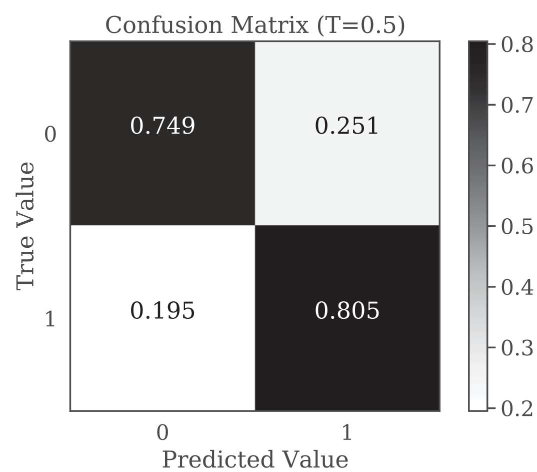

Figure 9: The confusion matrix of our method at the threshold 0.5.

6 CONCLUSION

In this paper, we present a novel framework to automatically calculate the matching score between talent's resume and

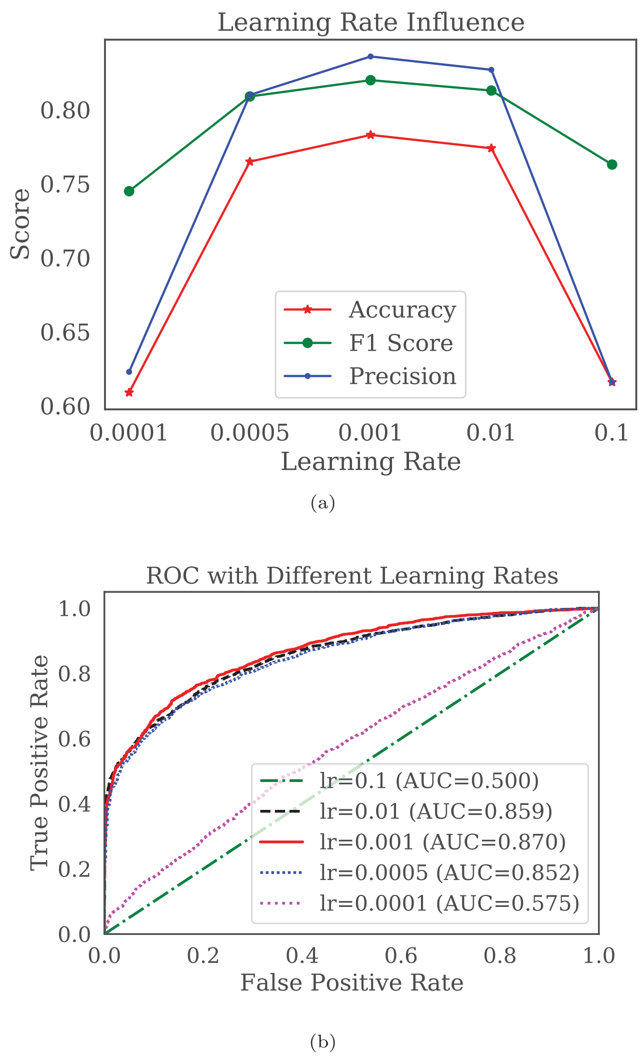

Figure 10: The performance of our method w.r.t. varied learning rate.

recruiter's job-post. In the proposed framework, sophisticated network structure is designed to learn appropriate representations for both the resume entries and job requirements. Besides, the adversarial learning mechanism is introduced to further improve the expressive ability of the learned representation. Effectiveness of our framework is demonstrated by conducting extensive experiments on a self-collected dataset.

From the results, we mainly conclude that the proposed framework outperforms other approaches consistently, and the performance can be further improved by the introduced adversarial learning strategy. In the future, we intend to collect more data from model test and design more sophisticated loss, as well as adopt adaptive learning rate.

ACKNOWLEDGMENTS

This work is supported by Singapore NRF2015ENCGDCR01001-003, administrated via IMDA and NRF2015ENCGBICRD001-012, administrated via BCA.

REFERENCES

[1] Shaha T Al-Otaibi and Mourad Ykhlef. 2012. A survey of job recommender systems. *International Journal of Physical Sciences* 7, 29 (2012), 5127–5142.

[2] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of Machine Learning Research* 3, Feb (2003), 1137–1155.

[3] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175* (2018).

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

[5] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. 2672–2680.

[6] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2015. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*.

[7] Christopher G Harris. 2017. Finding the Best Job Applicants for a Job Posting: A Comparison of Human Resources Search Strategies. In *International Conference on Data Mining Workshops*. 189–194.

[8] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *International Conference on World Wide Web*. 173–182.

[9] Prerna Khurana, Puneet Agarwal, Gautam Shroff, Lovekesh Vig, and Ashwin Srinivasan. 2017. Hybrid bilstm-siamese network for faq assistance. In *ACM Conference on Information and Knowledge Management*. 537–545.

[10] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *International Conference on Empirical Methods in Natural Language Processing*. 1480–1489.

[11] Diederik P Kingma and Max Welling. 2014. Auto-encoding variational bayes. *International Conference on Learning Representations*.

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

[13] Yong Luo, Huaizheng Zhang, Yongjie Wang, Yonggang Wen, and Xinwen Zhang. 2018. ResumeNet: A learning-based framework for automatic resume quality assessment. In *International Conference on Data Mining*. 307–316.

[14] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian Goodfellow. 2016. Adversarial autoencoders. In *International Conference on Learning Representations Workshop Track*.

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

[16] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Advances in Neural Information Processing Systems*. 6294–6305.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*. 3111–3119.

[18] Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the conference on Empirical Methods in Natural Language Processing*. 1532–1543.

[19] Matthew E Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. *arXiv preprint arXiv:1705.00108*

(2017).

[20] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Association for Computational Linguistics: Human Language Technologies*. 2227–2237.

[21] David Martin Powers. 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *International Journal of Machine Learning Technology* 2, 1 (2011), 37–63.

[22] 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 *ACM SIGIR Conference on Research & Development in Information Retrieval*. 25–34.

[23] Rohan Ramanath, Hakan Inan, Gungor Polatkan, Bo Hu, Qi Guo, Cagri Ozcaglar, Xianren Wu, Krishnaram Kenthapadi, and Sahin Cem Geyik. 2018. Towards Deep and Representation Learning for Talent Search at LinkedIn. In *ACM International Conference on Information and Knowledge Management*. 2253–2261.

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

[25] Zheng Siting, Hong Wenxing, Zhang Ning, and Yang Fan. 2012. Job recommender systems: a survey. In *International Conference on Computer Science & Education (ICCSE)*. IEEE, 920–924.

[26] Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Annual Meeting of the Association for Computational Linguistics*. 384–394.

[27] Jeff Weiner. 2012. The future of LinkedIn and the Economic Graph. *LinkedIn Pulse* (2012).

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

[29] Jiaolong Yang, Peiran Ren, Dongqing Zhang, Dong Chen, Fang Wen, Hongdong Li, and Gang Hua. 2017. Neural aggregation network for video face recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 4362–4371.

[30] Yiming Yang and Jan O Pedersen. 1997. A comparative study on feature selection in text categorization. In *Icml*, Vol. 97. 35.

[31] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Association for Computational Linguistics: Human Language Technologies*. 1480–1489.

[32] XianXing Zhang, Yitong Zhou, Yiming Ma, Bee-Chung Chen, Liang Zhang, and Deepak Agarwal. 2016. Glmix: Generalized linear mixed models for large-scale response prediction. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 363–372.

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

[34] 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), 12:1–17.