2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) | 978-1-7281-8028-1/20/$31.00 ©2021 IEEE | DOI: 10.1109/IAEAC50856.2021.9390614

Person-Job Fit model based on sentence-level representation and theme-word graph

Jiang Zhenhong[1] ,Peng Lingxi[1] ,Shi Lei[2]
1.College of Electronics and Communication Engineering, Guangzhou University, Guangzhou 510000, China
www.923888178@qq.com,ww.flyingday@139.com,ww.1017275370@qq.com

***Abstract*—Previous studies on Person-Job Fit mainly focused on the use of large amounts of data to train word-level embedding to model with the Joint Feature Extractor. Therefore, in order to reduce the computational load and text truncation problems caused by word-level representation and better integrate the idea of Click-through Rate prediction, a Person-Job Fit model based on sentence vector and theme-words knowledge graph (KG-DPJF) is proposed. The model uses a Bi-directional Long Short-term Memory Neural Network to extract theme-words from long text of recruitment resume and requirements and construct a knowledge graph. The model uses the Bert pre-training model to code the input sentences, and uses a multi-level attention mechanism to calculate the correlation among candidate resumes, historical resumes and recruitment requirements features, then inputs the correlation and weighted output to the classifier to predict the matching degree. Finally, the experiment is tested in the industry data set, and the KG-DPJF method achieves nearly 9% and 5% performance improvement compared with method based on logistic regression and single convolution neural network feature extraction method, and shortens training time by about four times compared with word-level embedding method.**

Key words: Person-Job Fit; Knowledge graph; Attention mechanism; Sentence-level ; Bert; Bi-directional long-short term memory neural network; Theme-words

I. INTRODUCTION

In the early stages of research on Person-Job Fit, such as [1-2], researchers generally focused on the relationship between people and organization (P-O) and the relationship between people and environment (P-E). Based on this, some job-matching models were developed, such as the recruitment model based on People-Organization matching [3] and the research on weight distribution in the combination matching of PO fit and person and job (PJ) fit [4]. In addition, some researchers have raised concerns about personality and job characteristics, which have an important impact on predicting person-job matching [5]. With the development of machine learning and deep learning, more and more researchers in machine learning have focused on Person-Job Fit. Some researchers use Back Propagation (BP) neural networks to design end-to-end matching models, such as [6]. Input manual features measure matching by Support Vector Machine (SVM), such as [7]'s work, which relies heavily on manual construction of features and requires the involvement of a large number of domain experts. However, even with the Word2vec embedding layer, the resume text is mapped to a set of vectors and the Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) is used as the advanced working method of the feature extractor [8-9], it still need to truncate sentences and control of the overall length of the text to avoid large input of feature dimensions. Besides, a large amount of data is needed to adequately train the embedded representation model using Skip-Gram method. In addition, some researchers use the Click-through Rate (CTR) idea of recommendation system to fully consider the matching of user preference information with projects for talent recommendation, such as [10-11]. The work [12] that incorporates the best historical data into the feature measurement calculation so that embedding the characteristics of a prior historical resume can enrich the information. The work of the bilateral recommendation approach, such as [13-14], are an important branch in the field of personalized recommendation system. The bilateral talent recommendation system matches the preferences of both job seekers and recruiters. This win-win strategy improves the matching accuracy. And [15] proposes a similarity calculation method that combines the explicit and implicit preference information of both users.

In addition, researchers have proposed to incorporate a large amount of prior expert information. They rely on the prior knowledge of expert groups to build a complete knowledge system for more accurate recommendations and more accurate targeted recommendations for some professionals, such as [16-17]. However, their reliance on a large amount of expert knowledge leads to high application costs and lack of automated construction methods.

In addition, related tasks include the task of predicting the direction of talent flow [18], and the work of exploring the network graph of talent skills [19], which fully taps the importance of individual skills of talent. CTR prediction tasks such as [20-21] also have an important correlation between the expression of resumes and recruitment requirements.

Based on these backgrounds, in order to alleviate the disadvantage of computational load caused by word-level embedding model, this paper presents an end-to-end Person-Job Fit model using sentence vector representation and incorporating knowledge graph information. The model fully considers the information interaction between candidate resume features and recruitment requirements features through a multi-level attention mechanism. The model embeds theme-word information and knowledge graph information to fully explore the relationship between text descriptors among candidates. By using the Translating Embedding[23] (Transe) method, the resume entity information is represented as a low-dimensional vector . Considering actual production applications, try to use a better pre-training encoding model: Bi-directional Encoder Representation from Transformers[24] (Bert).

II. PERSON-JOB FIT MODEL

A. Theme-word Graph Construction

Theme-word Graph G relies on information extraction models to extract theme words from non-structured text. To better identify theme words, use the Bilstm+CRF[22] model to capture potential theme-words in sentences. Bilstm is decomposed into a splicing of the output of a long Short-Term Memory (LSTM) sequence calculated with forward and backward time steps.

Definition 1.The LSTM[26] cell can be represented in the following formula:

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t = i_t \odot \tilde{C}_t + f_t \odot C_{t-1} \\ o_t = \sigma(W_o \cdot [x_t, h_{t-1}] + b_o) \\ h_t = o_t \odot tanh(C_t) \end{cases} \tag{1}$$

In formula (1), f_t is denoted as forget gate, it denote the input gate. The input sequence is $X = \{x_1, x_2, \ldots, x_t\}$ and t denote the input vector at time-step t. And W_f, W_i, W_C, W_o, b_f, b_i, b_C, b_o are the initialization parameters as weight matrices and biases, $\odot$ represents element-wise multiplication, σ is the sigmoid activation function, and $\{h_1, h_2, \ldots, h_t\}$ represents the output of a sequence of semantic features. Furthermore, the output h_t of a LSTM can be denoted as accepting x_t and outputting h_{t-1} from the previous LSTM cell.

Definition 2.The output h_t can be represented in the following formula:

$$h_t = LSTM(x_t, h_{t\text{-}1}) \tag{2}$$

As shown in Fig. 1, the input is a long text sequence data X of a resume, which is converted from the text description of work experience through word level index sequence. After a single time step input x_t is represented by a word level embedded layer, $\vec{h}_t$ is calculated by the input to the forward LSTM cell, $\overleftarrow{h}_t$ is calculated by the input to the backward LSTM cell, and the output h'_t of the hidden layer is the concatenation of $\vec{h}_t$ and $\overleftarrow{h}_t$. Bilstm can be represented as:

Definition 3.The $\vec{h}_t$, $\overleftarrow{h}_t$ and h'_t can be represented in the following formula:

$$\begin{cases} \vec{h}_t = LSTM(x_t, \vec{h}_{t-1}) \\ \overleftarrow{h}_t = LSTM(x_t, \overleftarrow{h}_{t+1}) \\ h'_t = [\vec{h}_t; \overleftarrow{h}_t] \end{cases} \tag{3}$$

The $h_1, \ldots, h_{t-2}, h_{t-1}$, h_t of the output layer enters the linear chain CRF layer. Given the input $\tilde{X}$ and label set $y = l_1, l_2, l_3, l_4, \ldots, l_t$.

Definition 4.The probability of sequence tag Y is derived:

$$\begin{cases} P(y \mid s) = \dfrac{1}{Z'} exp(\sum_t (W^{l_t} h_t + b^{(l_{t-1}, l_t)})) \\ Z' = \sum_{y'} exp(\sum_t (W^{l_t} h_t + b^{(l'_{t-1}, l'_t)})) \end{cases} \tag{4}$$

Where Z' is the normalization factor. y' represents any tag sequence, and W^{l_t} is the l_t-exclusive model parameter matrix, and $b^{(l_{t-1}, l_t)}$ is the l_t-exclusive model bias parameter matrix. Use the optimization function training model to learn the parameter matrices. For decoding, given the conditional probability P(y|s) of Conditional Random Field and an observation sequence $\tilde{X}'$, the sequence $\tilde{y}$ with the highest score is required. The Viterbi algorithm is used to compute the giving the transition parameter matrix and the sequence $\tilde{X}'$, resulting in the Viterbi result sequence and the Viterbi result value. Finally, the position of the theme-words is obtained from the Viterbi result sequence.

The triple is output by fixed ID-Item-value rule. All the triples form the knowledge graph *G*.

Definition 5. The h and G can be represented in the following formula:

$$\begin{cases} h = L(\mathrm{i}_s, \mathrm{k}_1) \\ G = (h, r, t) \end{cases} \tag{5}$$

After the graph *G* is constructed, the entities and relationships are mapped to a low dimensional continuous vector space using the Transe method. Suppose that h , r and t are the vectors corresponding to head, relation and tail respectively. If (h,r,t) has a triple relationship, it is assumed that $h+r \approx t$.

Definition 6. Transe evaluation function is defined as:

$$f_r(h,t) = \| h + r - t \|_2^2 \tag{6}$$

The smaller the evaluation function, the more reliable the triple relation (h,r,t) in the graph.

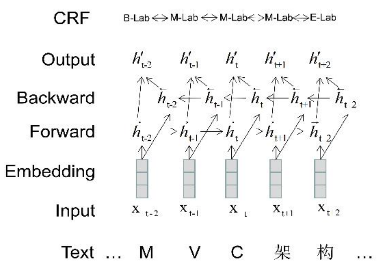

Fig. 1. Extracts information from subject words

B. Feature Representation Of Sentence-level

Thanks to Bert's powerful information encoding ability, sentence vectors rich in semantics can be generated without the need to learn embedded layer parameters from word level sequences using a large amount of data and Ship-Gram methods.

Definition 7. Bert encoding is represented as follows:

$$C = bert([s_1, s_2, s_3, ..., s_t]) \tag{7}$$

Where the Bert refers to Bert encoder, which is a large pre-training model trained by a Chinese dataset, Bert, an encoder with Bi-directional Transformer. The pre-training model uses a 768-dimensional Chinese pre-training model. The $[s_1, s_2, s_3, ..., s_t]$ is a sentence list after a text slice, where t represents a total sentence. $C \in \mathbb{R}^{dt \times de}$, where $d_e = 768$.

C. Feature Interaction Layer

As shown in Fig 2, the design of the model is similar to the traditional deep learning model, including data preprocessing, feature representation, multi-layer perceptron and classifier.

Feature representation: the input features of the model include three modules: the first is the current resume feature j , including gender, working years, education background, work experience slice, project experience slice, theme-words set, etc. Next is the best resume feature p_n , n is total of input of history resume features. Among them, features include gender, working life, academic qualifications, work experience slices, project experience slices, theme-words, etc. Finally, recruitment requirements feature q , including academic qualifications, work life requirements, professional requirements, specific requirements slices, theme-words, etc.

Definition 8. Define the current candidate resume j , which can be represented as:

$$j = [v_1, v_2, v_3, ..., v_d] \in \mathbb{R}^{d_j \times d_v} \tag{8}$$

Where v is the matrix after each item is represented by the embedded layer. The d is the number of items. Note that d_v is the vector dimension encoded by Bert, and the number of items is generally limited, that is, the value of the weight of the embedded layer is loaded into the model when the model graph is run and fine-tuned during subsequent model training.

Definition 9. The q and p_n can be represented as:

$$\begin{cases} q = [v_1, v_2, v_3, ..., v_d] \in \mathbb{R}^{d_q \times d_{v'}} \\ p_n = [v_1, v_2, v_3, ..., v_d]_n \quad \mathbb{R}^{d_{p_n} \times d_v}, \forall n \quad [1, ..., N] \end{cases} \tag{9}$$

Feature Interaction Layer: Its basic structure is an interactive computing layer based on feature extraction and Attention[25] mechanism, which uses implicit methods to obtain the correlation between resume and recruitment requirements. As shown in Fig 2, the feature matrix s' and e' are stitched together through sentence embedding and entity embedding layers, where entity feature e' matrix contains entity and entity context information. The stitched matrix is extracted by the convolution neural network K-Convolutional Neural Networks (KCNN). The fixed dimension resume representation vector j' is output, while the n history resumes p and the current recruitment requirements q are also embedded layer representation and then input into the convolution neural network for information extraction to generate the history resume representation p' and the recruitment requirements representation q', respectively.

The relationship between the current resume and the historical resume is modeled using a weighted attention calculation, and the result expresses the correlation weight $\hat{a}_{jk}$ between the current resume j' and the historical resume p'. This is considered a kind of feature interaction. Here the n historical resume features are averaged as equal weights. The same attention algorithm is also used to represent the relative weights $\hat{a}_{qk}$ of the recruitment requirements and the historical resume. Where $\hat{a}_{jk}$ and $\hat{a}_{qk}$ can be represented as follows:

Definition 10. The $\hat{a}_{jk}$ can be represented as:

$$\begin{cases} e_{jk} = z^{\top} tanh(W_j j + W_p p + b) \\ \hat{a}_{jk} = \dfrac{exp(e_{jk})}{\sum_{i=1}^{T} exp(a_{ji})} \end{cases} \tag{10}$$

Definition 11. The $\hat{a}_{qk}$ can be represented as:

$$\begin{cases} e_{qk} = z'^{\top} tanh(W_q q + W_p p + b) \\ \hat{a}_{qk} = \dfrac{exp(e_{qk})}{\sum_{i=1}^{T} exp(\tilde{a}_{qi})} \end{cases} \tag{11}$$

Where $W_j \in \mathbb{R}^{d_h \times d_{j'}}$, $W_p \in \mathbb{R}^{d_h \times d_{p'}}$, $b \in \mathbb{R}^{d_h}$, $z \in \mathbb{R}^{d_h}$, $W_q \in \mathbb{R}^{d_h \times d_{q'}}$, $W'_p \in \mathbb{R}^{d_h \times d_{p'}}$, $b' \in \mathbb{R}^{d_h}$ 和 $z' \in \mathbb{R}^{d_h}$ are the training parameter matrices. Then, output the weighted attention computed vector j'' of the current resume and the weighted recruitment representation vector q'', respectively. Here, j'' and q'' are correlated by a layer of attention, and $\hat{a}_{ck}$ is output. These layers of attention are mainly used to measure the implicit correlation among the three and to provide input for the classification layer.

Definition 12. The j'', q'' can be represented as follows:

$$\begin{cases} j = \sum_{l}^{T} \hat{a}_{jk} j' \\ q = \sum_{l}^{T} \hat{a}_{qk} q' \end{cases} \tag{12}$$

Definition 13. The $\hat{a}_{ck}$ can be represented as:

$$\begin{cases} e_{ck} = \tilde{z}^{\top} tanh(W'_j j + W'_q q + \tilde{b}) \\ \hat{a}_{ck} = \dfrac{exp(e_{ck})}{\sum_{i=1}^{T} exp(a_{ci})} \end{cases} \tag{13}$$

Where $W'_j \in \mathbb{R}^{d_h \times d_{j'}}$, $W'_q \in \mathbb{R}^{d_h \times d_{p'}}$, $\tilde{b} \in \mathbb{R}^{d_h}$, $\tilde{z} \in \mathbb{R}^{d_h}$. Finally, after attention calculation, all the correlation weights are concatenated together into $u = [\hat{a}_{ck}, \hat{a}_{jk}, \hat{a}_{qk}]$. Then u and Resume Representation j'' and Requirement Representation q'' are concatenated together to form the input $\hat{X} = [j'', q'', u]$ of the prediction layer.

D. Prediction Layer

In the prediction layer, these weighted vectors are connected through the full connection layer. Hidden layer is activated by Parametric Rectified Linear Unit (Prelu[27]) function. Use sigmoid cross-entropy loss function to calculate the loss between Logits and labels. Therefore, the vector of the input fully connected layer is x, and the label of the classification is $y \in \{0,1\}$. The formula for calculating the loss function is:

Definition 14. The formula for calculating the loss function L_{target} is::

$$L_{target} = -\frac{1}{N} \sum_{(x,y) \in D} (y \log(f(x)) + (1-y)\log(1-f(x))) \tag{14}$$

Where the D is the training set, with a total of N samples, and $f(x)$ is the prediction result of the model output.

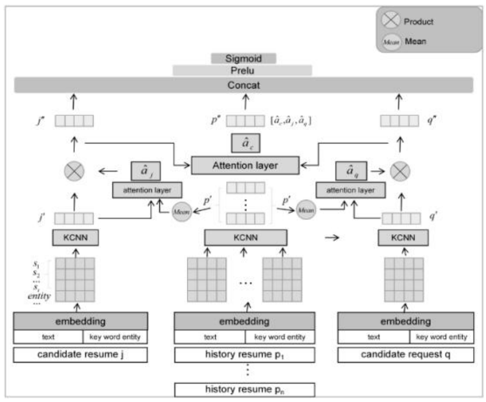

Fig. 2. KG-DPJF model

III. Experiments

The experiment uses the data set collected in the real recruitment environment and makes relevant statistical analysis. Then the performance indicators of KG-PDJF and some typical Person-Job Fit baseline methods are compared in the real data.

A. Data Set Statistics And Preprocessing

This data set is collected from enterprise registered recruitment platforms and has the same structure as the resume data downloaded from the well-known 51job enterprise recruitment application page in China. The valid time for collection is 2 years, totaling 19381 resumes. During the data cleaning stage, it need to exclude the resumes that contain full English and a large amount of English from the text, which leave a large amount of blank information or important information not filled in, such as work experience, academic qualifications, etc. which have a greater impact on the experimental results, and use a large number of regular expressions to pre-process the data.785 resumes that do not meet the requirements were excluded.

Thanks to the confidentiality of the data, the data set is desensitized, including removing all sensitive personal information such as name, phone, family information, etc. The data refers to the main daily jobs of small and medium-sized technology companies. The distribution of job and resume data is shown in Fig 3:

Among them, the label [3, 16, 22, 23, 24, 25, 26, 27, 31] is the technical posts such as Java intermediate Engineer posts with serial number 3, and accounts for the most. The total number of technical posts delivered is 9295. The remaining are civilian posts with a total number of 4375. As you can see from Fig.3 and 4, there are more resumes and theme-words for technical posts than for civilian posts. This reason can be explained. Due to the communication with the Human Resources Department, the technical category of the enterprise accounts for more than 60% of the employees (excluding the impact of personnel flow). Moreover, the number of job seekers in the technical category tends to be more competitive, while the technical field tends to have more professional terms.

At the same time, the recruitment mechanism of the recruitment platform and the employing mechanism of enterprises should be considered before the experiment. This is also the basis for classification. Label 0 stands for candidates who only stop at the stage when resumes enter the talent folder, label 1 only completes the invitation interview stage, and label 2 only completes the invitation interview and is offered. Note that in the actual training data, there are only 0 and 1 labels, that is, only the data that passes the interview will be 1, and the remaining resume labels in other stages will be 0.

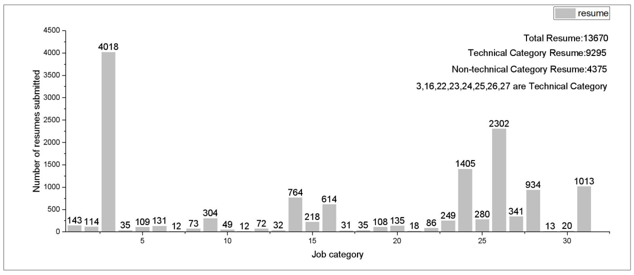

Fig. 3. Delivery statistics

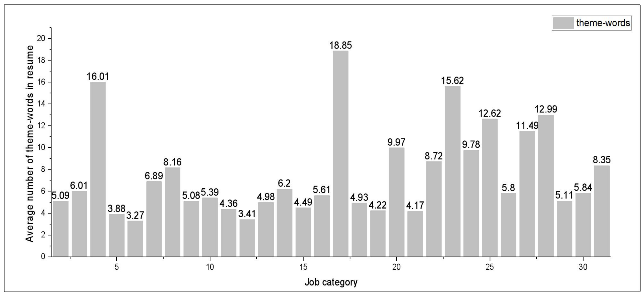

Fig. 4. Theme-word count statistics

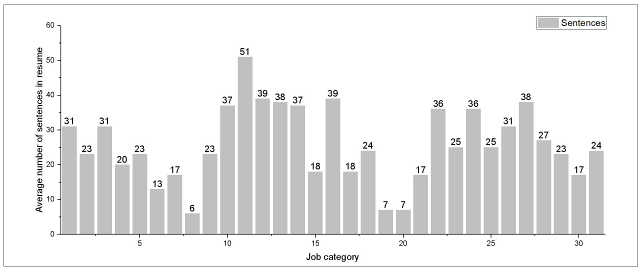

Fig. 5. Resume sliced sentence statistics

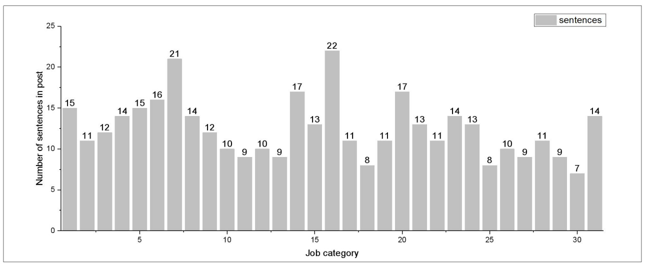

Fig. 6. Recruitment requirements sliced sentence statistics

B. Experimental Processes

In experiment, the learning rate is set to 0.001, the batch size is 8, the length of sentence embedding vector is 768, the size of entity embedding is 50, and the maximum length of resume sequence and request is 40. In addition, the number of Bilstm layers in the information extraction model is 200, the learning rate is 0.001, the batch size is set to 32, and the embedding size is set to 200.

In the comparative experiments in Fig 7 and 8, we tested the performance of the DPJF model using Word2vec and the information extraction model to select the appropriate dimension of the embedded layer. In the subsequent models, DPJF-word2vec uses the 100-dimension embedded layer because it balances training time and classification performance. The information extraction model is only trained in the data construction stage, the 200-dimension embedded layer is selected.

The theme-words in Fig 4 statistics are quite different between technical and civilian related positions. After regular cleaning of theme-words and other structured information, construct a knowledge graph. The sample data shown in Fig 9 is downloaded from 51job, which strictly protects the privacy of personal information. All data are desensitized. The three resume data in the picture are extracted from the work experience of three resumes in the same job in the same enterprise. The theme-words contain the main meaning of the sentence and provide information support for matching tasks.The model is optimized by Adam algorithm.

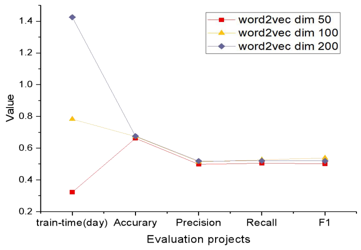

Fig. 7. Performance comparison for each embedded dimension of DPJF

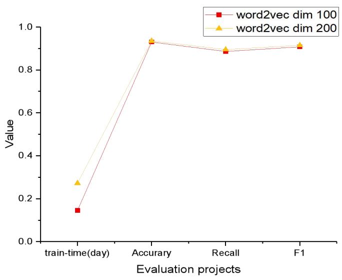

Fig. 8. Performance comparison for each embedded dimension of information extraction

Fig. 9. Sample theme-word of Chinese Resume

C. *Evaluation And Baseline Comparison*

The evaluation baseline refers to the typical method of Person-Job Fit evaluation baseline [29-30]. Since it is tested in the industry data, multiple baseline methods need to be constructed according to the input mode. Therefore, in the validation model phase, a Person-Job Fit model is built using traditional methods including Logistic Regression (LR) and Decision Tree (DT).

In addition, according to the input mode of DPJF, the experiment also specifically constructs a model using CNN structure to verify the importance of fusing multiple attention mechanisms. Compare the pre-trained word2vec embedding layer with the Bert-encoded sentence-level embedding. Finally, the experiment compares the performance of the original model after removing the knowledge graph. Note that without annotations, DPJF defaults to no graph structure information and uses Bert encoding.

In addition, methods based on CNN structure model are also built for experimental comparison. The CNN model based on word-level word2vec embedding intercepts 1480 sentences according to the total length of the text and breaks them in a fixed way. It uses word2vec embedded layer to be 100 dimensions. Also, a model of CNN structure based on Bert encoding is constructed. The performance of DPJF model using word2vec and Bert encoding is compared. Finally, a comparison is made between the models with non-KG graph information and theme-words information. The model structure constructed using the CNN method is shown in Fig 10:

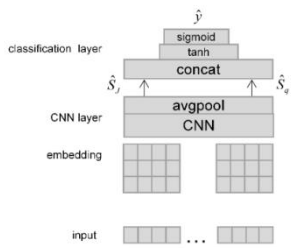

Fig. 10. Baseline model based on CNN

The convolution neural network is input to output the Resume feature representation X and the requirement feature representation X respectively. These features are input into the classifier D together, calculated by the weighted activation function layer and output to the sigmoid layer, and the X value is predicted:

$$D = \tanh(W_d[\hat{s}_j;\hat{s}_q;\hat{s}_j - \hat{s}_q] + b_d) \tag{16}$$

$$\hat{y} = sigmoid(W_y D + b_y) \tag{17}$$

Finally, the admission pass is set to 1, not at other stages is set to 0, and the results also adopted the Accuracy, Precision, Recall and F1- measure as the evaluation metrics.. Finally, 80% of the data set is selected as training set. Another 10% is used to fine tune parameters, and the last 10% is used as test data as test set.

D. Experimental Result

In summary, the task is defined as a two-category task, and the related experimental work of Person-Job Fit model is carried out. The problems of data imbalance and label class imbalance are shown in Fig 3 and Tab 1. However, due to the differences in job requirements and resume texts between civilian and technical posts, the data can be divided into two categories: civilian and technical posts. At the same time, the amount of data based on the real scene also limits some sampling methods. Therefore, based on the actual production and application, the experiment also enhances the data by cooperating with enterprises and using a part of expert knowledge. For example: Resumes for low-level technical posts may be more similar to those for medium-level technical posts, noting that about 3.72% of candidates choose to deliver multiple jobs at the same time. Based on this idea, small portions of data are sampled from each other and labeled manually in similar positions. This is to increase the number of labels that are in the process of being accepted and interviewed only.

As shown in Tab 2 and Fig 11, the designed KG-DPJF model performs better than the traditional feature classifier based model. The method of embedding theme-words information and calculating correlation using multiple layers of attention also improves the performance of the model, which is better than using a single CNN network feature extractor. In addition, the analysis of training time also shows that using Bert-based sentence encoding method has a huge advantage in training time. Word-level based method can greatly increase the amount of calculation, resulting in a relatively long training time.

Tab 1 also shows that relying on industry data, using only the pre-training model Bert for sentence coding may not be better than using the pre-training word2vec embedding layer, which may be the cause of some loss of information due to the use of pre-training models and sentence vectors. In the experiment, KG-PDJF also performs better than PDJF, which also shows that embedding theme-words can help the model learn the main idea of sentences better, express correlation, and increase matching accuracy while reducing computational load and speed up operations.

TABLE I. EXPERIMENTAL DATA ANALYSIS TABLE

Statistics	Value
The proportion of job seekers who submit two or more jobs at the same job	0.0372
Average number of words in experience text per resume for technical posts	1271
Average number of words of experience per resume for civilian posts	643
Ratio of technical post recruitment passed	0.3733
Ratio of civilian recruitment Passed	0.2642

TABLE II. EXPERIMENTAL EVALUATION TABLE

Model	Accurary	Precision	Recall	F1	AUC
LR	0.6271	0.4493	0.4473	0.4627	0.5623
DT	0.6269	0.4478	0.4615	0.4646	0.5844
CNN-word2vec	0.6528	0.4852	0.5077	0.4962	0.5725
CNN-bert	0.6477	0.4776	0.4923	0.4849	0.5871
DPJF	0.6632	0.5001	0.5385	0.5185	0.6037
DPJF-word2vec	0.6736	0.5147	0.5263	0.5381	0.5946
KG-DPJF	0.6939	0.5278	0.5846	0.5548	0.6217

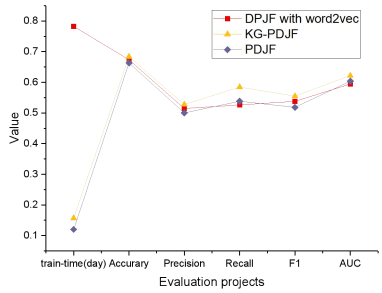

Fig. 11. Performance comparison between word 2vec and sentence level embedding

IV. CONCLUSIONS

The KG-DPJF uses multi-layer attention interaction to model the correlation among resume, historical resume and requirements, and uses a method of embedding theme-words and KG information to reduce the information impact of coding sentence vector by using large-scale pre-training model Bert. At the same time, due to the large number of text words in real data, sentence coding can reduce the problem of using word2vec encoding to some extent, which leads to the problems of text truncation and computation. The sentence level representation based on Bert model and embedding of theme-words information can reduces the amount of computation, and ensures the amount of information. Finally, industry data and validation of the model are carried out, and the new attempt of data processing method and data enhancement is proposed. Experiments show that the proposed KG-PDJF model has innovative construction method and has better performance than traditional methods.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. 61772147 and No. 61100150，University Innovation Team Construction Project of Guangdong Province under Grant No. 2015KCXTD014，Guangdong Province Philosophy and Social Science Foundation under Grant No. GD19CSH03，Open Research Fund Program of Data Recovery Key Laboratory of Sichuan Province.

REFERENCES

[1] Carless S A. Person–job fit versus person–organization fit as predictors of organizational attraction and job acceptance intentions: A longitudinal study[J]. Journal of Occupational and Organizational Psychology, 2005, 78(3): 411-429.

[2] Werbel J D, Gilliland S W. Person–environment fit in the selection process[J]. 1999.

[3] Zhang Zhengtang, Ye Yingchun. Recruitment Model Based on People-to-Organization Matching [J]. Research on Financial Issues, 2006 (03): 94-98.

[4] Sekiguchi T, Huber V L. The use of person–organization fit and person–job fit information in making selection decisions[J]. organizational behavior & human decision processes, 2011, 116(2):203-216.

[5] Ehrhart K H. Job Characteristic Beliefs and Personality as Antecedents of Subjective Person–Job Fit[J]. Journal of Business and Psychology, 2006, 21(2):193-226.

[6] Zhenzhen Y, Postgraduate L S, Management S O, et al. Application in Person-Job Fit Evaluation of BP Neural Network in Knowledge Workers[J]. journal of wuhan university of technology(information & management engineering), 2010.

[7] Bing Y I, Fei J. Application in Person-Job Fit Evaluation of Support Vector Machine[J]. journal of central south university of forestry & technology(social sciences), 2011.

[8] Qin C, Zhu H, Xu T, et al. Enhancing person-job fit for talent recruitment: An ability-aware neural network approach[C]. The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. 2018: 25-34.

[9] Zhu C, Zhu H, Xiong H, et al. Person-Job Fit: Adapting the Right Talent for the Right Job with Joint Representation Learning[J]. ACM Transactions on Management Information Systems, 2018.

[10] Lu Y, El Helou S, Gillet D. A recommender system for job seeking and recruiting website[C]. Proceedings of the 22nd International Conference on World Wide Web. 2013: 963-966.

[11] Paparrizos I, Cambazoglu B B, Gionis A. Machine learned job recommendation[C]. Proceedings of the fifth ACM Conference on Recommender Systems. 2011: 325-328.

[12] Di W U, Li-Juan Z, Hong-Fei L. Design and Implementation of Job Recommendation System for Graduates Based on Random Walk[J]. journal of guangxi normal university(natural science edition), 2011.

[13] Malinowski J, Keim T, Wendt O, et al. Matching people and jobs: A bilateral recommendation approach[C]. Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06). IEEE, 2006, 6: 137c-137c.

[14] PIZZATO L, REJ T, CHUNG T, et al. Reciprocal recommenders[C]. The 8th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems(UMAP'2010), June 20-24, 2010, Hawaii, USA. [S.l.:s.n.], 2010:1-12.

[15] YU H, LIU C, ZHANG F. Reciprocal recommendation algorithm for the filed of recruitment[J]. Journal of Information& Computational Science, 2011, 8(16): 4061-4068.

[16] BURKE R. Knowledge-based recommender systems[J]. Encyclopedia of Library and Information Science, 2000, 69(Supplement):32.

[17] VELASQUEZ J D, PALADE V. Building a knowledge base for implementing a webbased computerized recommendation system[J]. International Journal on Artificial Intelligence Tools, 2007, 16(5): 793-828.

[18] Zhang L, Zhu H, Xu T, et al. Large-scale talent flow forecast with dynamic latent factor model?[C]The World Wide Web Conference. 2019: 2312-2322.

[19] Xu T, Zhu H, Zhu C, et al. Measuring the Popularity of Job Skills in Recruitment Market: A Multi-Criteria Approach[J]. 2017.

[20] Zhang T, Liu B, Niu D, et al. Multiresolution graph attention networks for relevance matching[C]Proceedings of the 27th ACM International Conference on Information and Knowledge Management. 2018: 933-942.

[21] Lyu Z, Dong Y, Huo C, et al. Deep Match to Rank Model for Personalized Click-Through Rate Prediction[J]. 2020.

[22] Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging[J]. arXiv preprint arXiv:1508.01991, 2015.

[23] Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data[C]. Advances in neural information processing systems. 2013: 2787-2795.

[24] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018

[25] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]. Advances in neural information processing systems. 2017: 5998-6008.

[26] Hochreiter S, Schmidhuber J. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8):1735-1780.

[27] He K, Zhang X, Ren S, et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[J]. 2015.

[28] Zhu C, Zhu H, Xiong H, et al. Recruitment Market Trend Analysis with Sequential Latent Variable Models[J]. 2017.

[29] Qin C, Zhu H, Xu T, et al. An Enhanced Neural Network Approach to Person-Job Fit in Talent Recruitment[J]. ACM Transactions on Information Systems (TOIS), 2020.