IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 26, 2018, accepted June 29, 2018, date of publication July 11, 2018, date of current version August 7, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2854887

Feature Regularization and Deep Learning for Human Resource Recommendation

HAOXIANG WANG, GUIHUANG LIANG, AND XINGMING ZHANG
School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China
Corresponding author: Xingming Zhang (cszxm@scut.edu.cn)

This work was supported by the Guangdong Provincial Science and Technology Plan under Grant 2017B010111003.

ABSTRACT A novel recommender system is proposed in this paper. It has been implemented for human resource recommendation and achieved improvement on different evaluation metrics. The algorithm leverages both gradient boosting tree model and a convolutional network-based deep learning model for feature regularization and recommendation. The optimizations of activation function and pooling strategy in the proposed network model have been investigated for mitigating the problems of the gradient disappearance and the feature loss in pooling and for the improvement of recommendation quality. Human resource datasets are fetched by using a cloud-based distributed data collecting framework. Using the datasets, experiments on the proposed recommender system have been done and analyzed. Our proposed algorithm shows better recall rate and F1-score than some other recommender algorithms.

INDEX TERMS Recommender systems, decision trees, artificial neural networks, cloud computing.

I. INTRODUCTION

The employment issue now has been a social problem that needs to be solved urgently. With the rapid growth of data, employees and employers are both facing challenge of efficiently finding useful information in the massive amount of human resource data. This paper concentrates on the recommendation algorithms for job seekers. As for job seekers, the problem is that: searching for potentially interesting information about job vacancies usually take a lot of time and effort, therefore, recommender systems are required for improving the accuracy and efficiency of job searching.

Recommender algorithms have been researched for long time. The classical collaborative filtering recommendation can be tracked back to 1994 [1]. More recently, the booming growth of Deep Learning brings a new opportunity for conducting optimization for recommendation algorithms and systems. Covington *et al.* [2] described how deep neutral networks can be used for YouTube video recommendation. Cheng *et al.* [3] proposed a wide & deep model for recommendation and evaluated it on Google Play. A RNN (Recurrent Neural Networks) based approached is presented in [4], and has been applied on Yahoo news recommendation. Zhang *et al.* [5] describe a survey of researches which exploit different deep learning based approaches for recommendation, including Bayesian based models, different neural networks, clustering and classification techniques, etc. Georgieva and Nakv [6] proposed a framework for collaborative filtering based on Restricted Boltzmann Machines (RBM). Deep Semantic Similarity Model (DSSM) is a deep neural network widely used in information retrieval area. DSSM projects different entities into a common low-dimensional space, and computes their similarities with cosine function. It is supremely suitable for top-n recommendations [7]. Convolution Neural Network (CNN) has been discovered as a powerful tool for recommender systems. Most of the CNN based recommender systems utilize CNN for feature extraction. There are different way of utilization CNN for recommender systems, such as [8]–[10]. For the recommendation scenario with the temporal dynamics of ratings and sequential features, Recurrent Neural Network (RNN) is considered as a suitable deep learning technique for recommender system [11]–[13].

This paper proposes a novel recommender system by utilizing a hybrid-convolutional network model with GBDT (Gradient Boost Decision Tree) for feature regularization. The algorithm exploits information from both job seeker and job posts to explore the potential correlations between them, for achieving high-quality personalized human resource recommendation. The main contributions of this paper include: 1) the design and implementation of recommender system with a gradient boosting tree model and a hybrid CNN model for feature regularization and recommendation;

2) the optimization of activation function and pooling strategy for the mitigation of the gradient disappearance and the feature loss in pooling, thereby to improve the personalized recommendation quality. The recommender system is evaluated on a human resource dataset. Compared with some other recommendation algorithms, the experimental results show that our algorithm achieves better recall rate and F1-Score than the competitive algorithms.

This paper is organized as follows: A brief overview of proposed system is presented in Section2. Section 3 describes a GBDT based model for feature regularization. Section4 details a CNN-based network model for generating recommendation. Section 5 shows experimental results and analysis. Finally, Section 6 presents our conclusions and lessons learned.

II. OVERVIEW OF THE SYSTEM

The design of a recommender system is strongly related with specific application scenario. Our recommendation system concentrates on improving human resource recommendation quality. This section depicts a brief explanation of the workflow of our recommendation system, including data colleting and pre-processing, recommender system model and top-k recommendation. The system overview is shown as Figure 1.

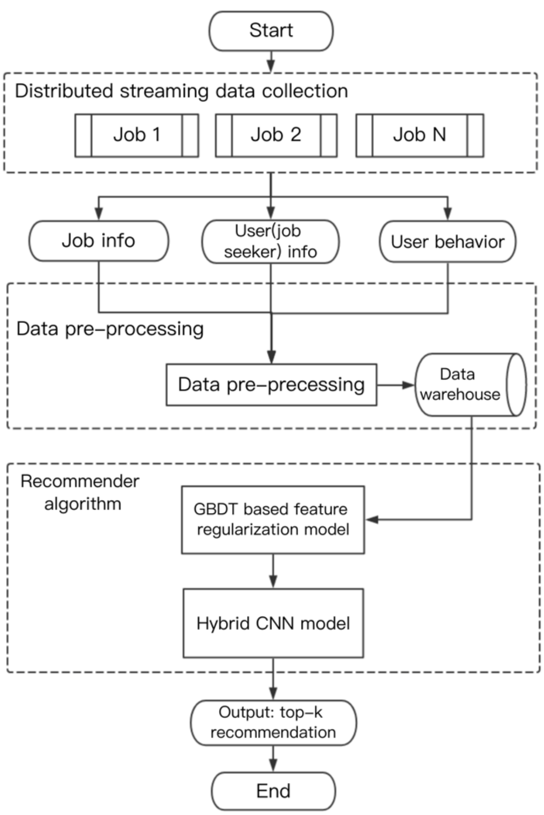

FIGURE 1. Recommender system overview. The system consists of a cloud-based distributed data collection framework, data pre-processing module and recommender networks.

The human resource data is collected by using a cloud-based distributed data collection framework. A number of collecting agents are deployed on cloud cluster and fetch human resource data from different online sources. The data collected includes personal information of job seekers, information of job vacancies, and behaviors of job seekers. The behaviors reflect user interest on jobs, such as viewing the job description, saving the job as favorite, and applying for the job. In the data pre-processing stage, raw data is purified and encoded so that the dataset can be presented as matrix for subsequent processing.

The core of recommender system consists of two cascaded models: a feature regularization model and a deep neutral network model. The feature regularization model uses GBDT and Embedding technique [4]. The deep neutral network model uses a combination of multiple convolution channels and local connections. Details of the two core models are explained in the following sections.

On the completion of model trains, the network can be used for recommendation for un-labeled data. In this paper, for the evaluation of the recommender system, test dataset is used on the recommender model for experimental analysis and comparison.

III. FEATURE REGULARIZATION MODEL

Unlike graphical data for computer vision, it is more difficult to discover implicit correlations amongst the features of human resource data. Therefore, we design a feature regularization model, trying to extract some implicit correlations in the data, as the first processing stage in the recommender algorithm.

The feature regularization model contains two components: Embedding based dimensionality reduction and GBDT based feature regularization, as show in Figure 2.

A. EMBEDDING

In the collected human resource data, non-numerical data are converted to numerical data by using one-hot encoding. However, one-hot encoding results in high dimensionality and sparseness in matrix. Both of them with harm the performance of GBDT. Embedding is a popular dimensionality reduction approach, which mapping data from high dimension space to low dimension. Mathematically, the operation of Embedding can be represented as (1), in which z presents the index for feature $x_z \in \mathbb{R}, W_z$ stands for matrix $m_z \times n_z$, b is the bias, y_z is the vector with reduced dimensionality, and $\mathcal{F}(\cdot)$ is nonlinear activation function.

$$y_z = \mathcal{F}(W_z x_z + b_z) \tag{1}$$

The Embedding result is merged with numerical features. This step is represented as symbol $\sum$ in Figure 2. The merged matrix is then inputted into the GBDT model for feature regularization.

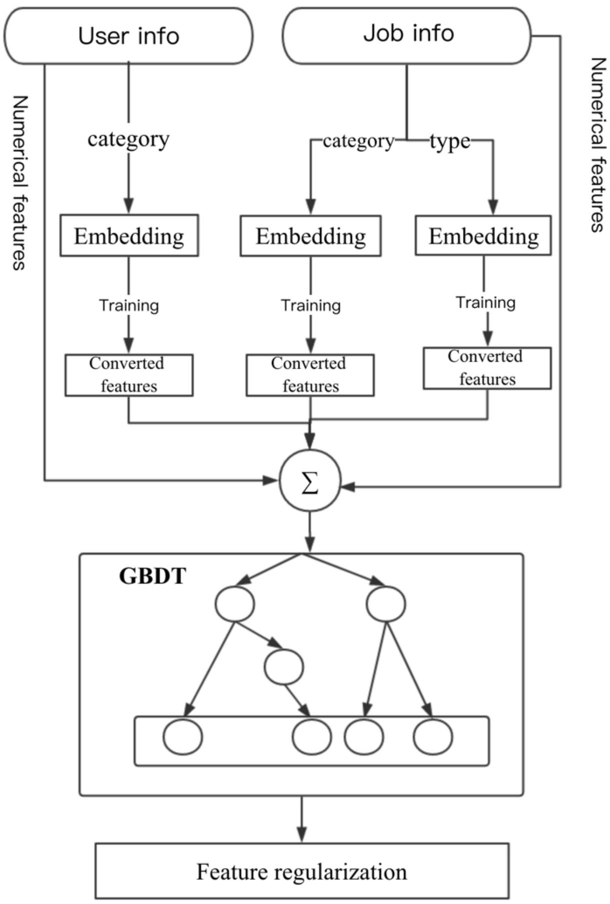

FIGURE 2. Structure of feature regularization model, which contains two components: embedding based dimensionality reduction and GBDT based feature regularization.

B. GBDT BASED FEATURE REGULARIZATION

According to [14], convolution is considered as suitable option for processing data with relevance between neighboring samples, such as pixels in graphical data. Inspired from the work described in [15], we utilize GBDT for feature regularization. Each decision tree in the model can be regarded as a feature for classification, which is a vector with 0s or 1s as its elements, corresponding to the leave nodes on that decision tree. Figure 3 demonstrates an example of a GBDT model with 3 trees, which convert input data sample as a vector.

We use XGBoost [16] as GBDT model in implementation. XGBoost is an optimized GBDT model, which runs fast and is highly scalable. Compared with traditional GBDT, it applies a novel sparsity-aware algorithm for parallel tree learning and a theoretically justified weighted quantile sketch.

In this phase, our recommender system uses decision tree to help filtering large number of features and extracting implicit correlations amongst these features. The output from feature regularization model will be inputted to our hybrid CNN model.

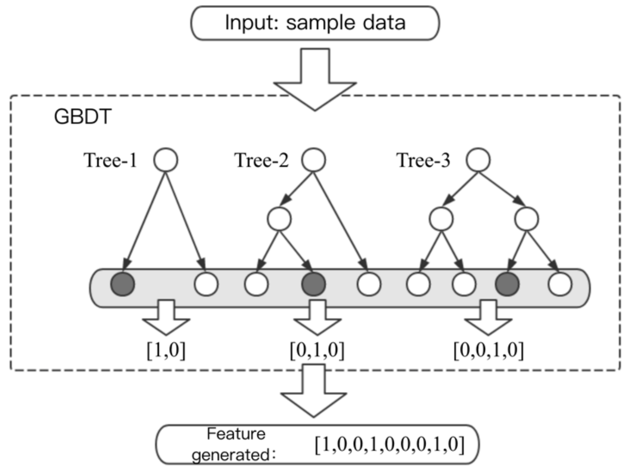

FIGURE 3. A demonstration of an example of a GBDT model with 3 trees, which convert input data sample as a vector.

IV. HYBRID CNN MODEL

This section describes the design and implementation of a hybrid CNN model. On the completion of training, this hybrid CNN model can be used for producing recommendation. The hybrid CNN model consists of two cascaded sub-models: multiple convolution channels sub-model, local connection and convolution sub-model, as shown in Figure 4

A. MULTIPLE CONVOLUTION CHANNELS SUB-MODEL

In a CNN network, each convolution kernel is of the same size and only effects on a small local regain in the matrix. The multiple convolution channels sub-model employs the idea of using different convolution kernels on each channel. Compared with unified convolution kernel, using various size convolution kernels can help extract more implicit features. Data quality can be enhanced by apply multiple convolution channel, which can improve the recommendation quality for the entire recommender system.

The sub-model is shown as Figure 5, in which *filter_k* represents for the kth convolution filter in channel n and symbol $\sum$ represents the emergence of the vectors to produce output matrix. The convolution channel employs ReLU as Nonlinear activation function, and uses max pooling as pooling strategy. Mathematically, each convolution channel can be described as (2), in which c_i represents for the ith convolution, X is the input data, K_i is the corresponding convolution kernel, b_i is the corresponding bias, and $\mathcal{F}(\cdot)$ is the activation function.

$$c_i = \mathcal{F}(K_i * X + b_i) \tag{2}$$

B. LOCAL CONNECTION AND CONVOLUTION SUB-MODEL

The output from the sub-model described in previous section, is then passed into local connection and convolution sub-model for producing recommendation. This sub-model is displayed as Figure 6. The input data is processed through a convolution network and a local connection network in parallel. Conceptually, the convolution network tries

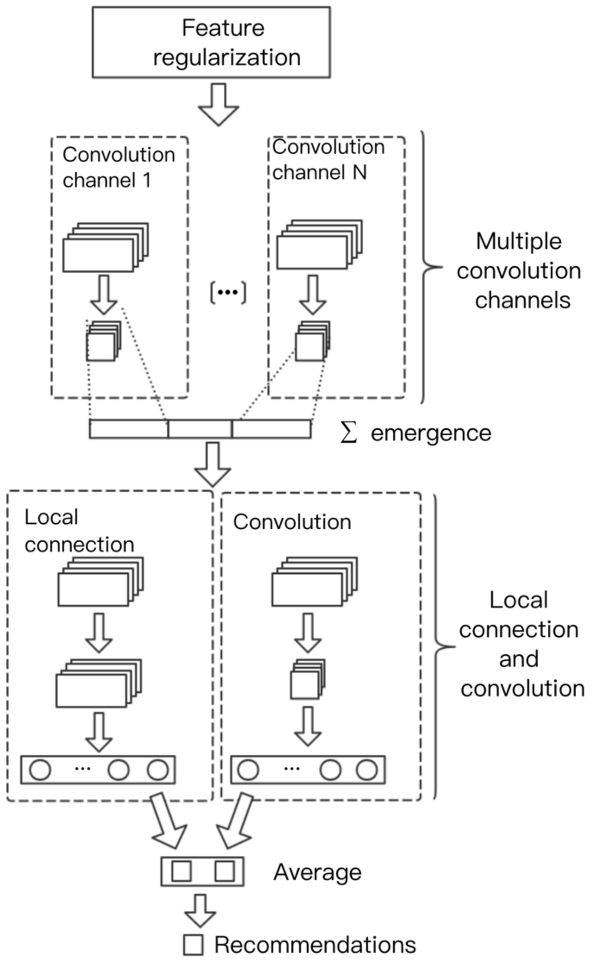

FIGURE 4. **Hybrid CNN model consisting of two cascaded sub-models: multiple convolution channels sub-model, local connection and convolution sub-model.**

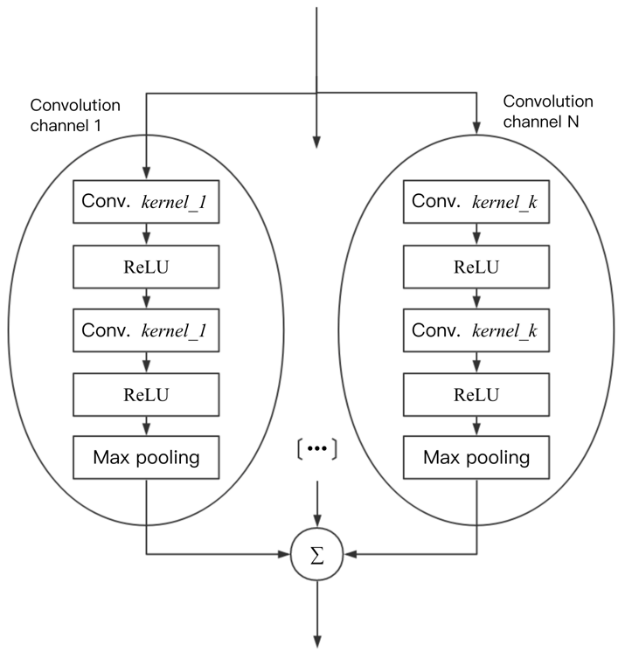

FIGURE 5. **The multiple convolution channels sub-model employs the idea of using different convolution kernels on each channel.**

to explore the global characteristics of the data, and the local connection network can be incline to discover local characteristics.

We convert recommendation problem into classification problem. That is why a Sigmoid activation function is used as the output layer in both convolution network and local connection network. The output result is the probabilities of the input data example falling into two classes: recommending or not. The probabilities computed from two networks are averaged as the final recommendation decision.

C. MODEL OPTIMIZATIONS

ReLU function is used in both convolution network and local connection network. However, ReLU might cause the issue of gradient disappearance. Pooling might be the other potential problem in our model, as it might lose some feature of sample data. Both of these two issue might end up with slow convergence or feature loss in pooling. The optimizations include two aspects: using ELU (Exponential Linear Unit) [17] to replace ReLU and adapting more comprehensive pooling strategy.

As shown in (3), ELU saturates to a negative input values with smaller inputs, thereby it decreases the forward propagated variation and information. Therefore, ELU counts the degree of presence of particular phenomena in the input, while it does not quantitatively model the degree of their absence. In our human resource recommendation, it has been proved that ELU helps speed up the convergence of the model.

$$\mathrm{ELU}(x) = \begin{cases} x, & x > 0 \\ \alpha \cdot (\exp(x) - 1), & x \le 0 \end{cases} \tag{3}$$

For the mitigation of weakening features potentially cause by single max pooling, a more comprehensive hybrid pooling method [18] can be used for optimization. In our recommender system, we use a hybrid of max pooling and average pooling, which can be represented as (4). In the equation, $\max_{(p\times p)_{i=1,j=1}} M_{ij}$ is a $p \times p$ size max pooling on input matrix M, b is the bias, and F is the hybrid pooling output. The experiments show that the use of hybrid pooling relieves the problem of feature loss and overfitting, enhance the capacity of generalization.

$$F_{ij} = \frac{1}{2} \cdot \left(\frac{1}{p \times p} \cdot \left(\sum_{i=1}^{p} \sum_{j=1}^{p} M_{ij} \right) + \max_{(p\times p)_{i=1,j=1}} M_{ij} \right) + b \tag{4}$$

V. EXPERIMENTS AND ANALYSIS

We use collected human resource dataset as experimental data. Data are classified as positive samples, which users

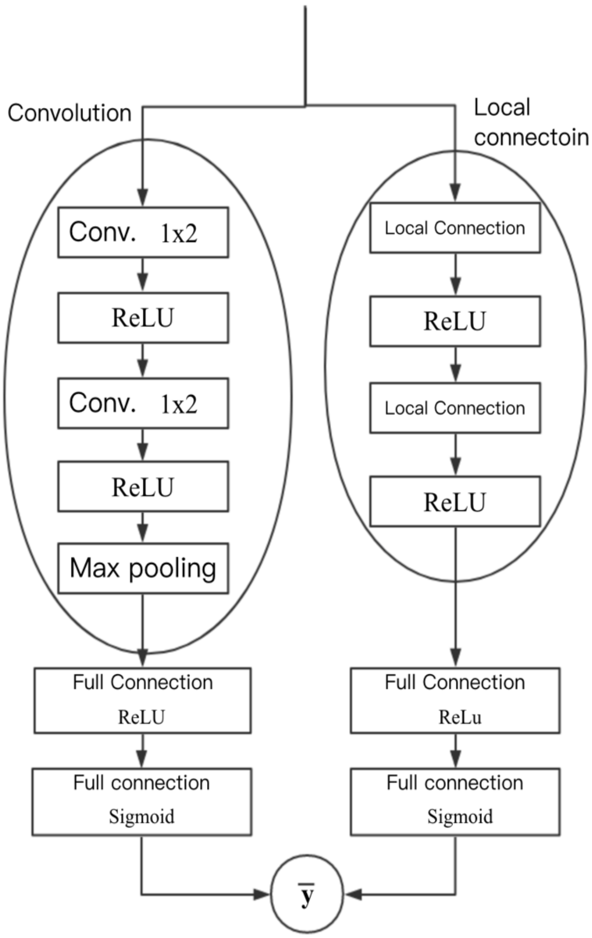

FIGURE 6. Local connection and convolution sub-model. Convolution network is used for explore global characteristics of the data and local connection network tries to discover local characteristics.

(job seekers) conducting viewing, favoring or applying on a particular job post, and negative samples, which have no user behaviors related. The dataset involves 4692 users, 15000 job posts. Totally, it has 481478 samples, amongst which 288500 samples are positive and 192978 are negative.

The experiments are run on Keras 2.0.8, which hardware resource listed as follows: Intel Core(TM) i7-6700 CPU @3.40GHz, 16GB DDRR3, NVIDIA GeForce GTX 750 Ti.

The evaluation criteria used in the experiments include recall rate and F1-Score. F1-Score is a comprehensive index reflecting both recall rate and precision rate. It is computed using (5).

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

A. GBDT BASED FEATURE REGULARIZATION MODEL

Our recommender model uses a GBDT for feature regularization, which will then be used for convolutional network and local connection network. Experiments have been done to prove the effectiveness of the feature regularization model.

The experimental results are shown as Figure 7 and Figure 8. With 60 recommendations, without using GBDT for feature regularization, the recall rate is 0.7612 and F1-Score is 0.724. Under the same circumstance, the model proposed in this paper with GBDT can reach recall rate 0.8112 and F1-Score 0.74. It can verify that the use GBDT feature regularization can improve the performance of our recommendation system.

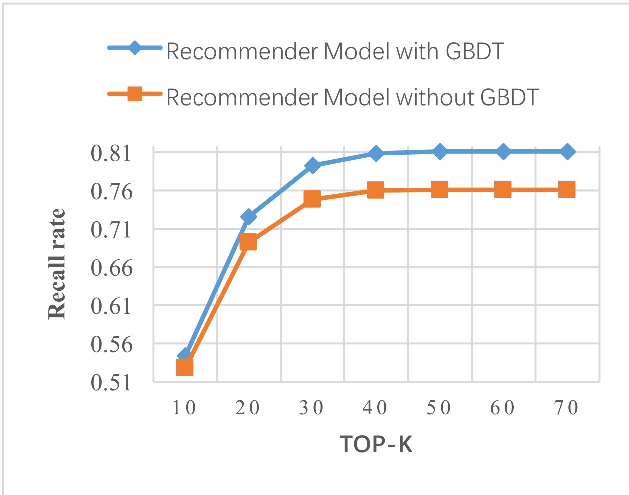

FIGURE 7. The comparison of recall rate between the recommender models with and without GBDT for feature regularization.

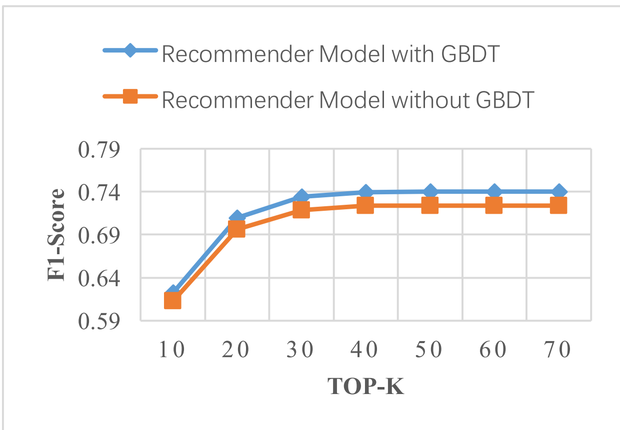

FIGURE 8. The comparison of F1-score between the recommender models with and without GBDT for feature regularization.

B. MULTIPLE CONVOLUTION CHANNEL NETWORK

In order to enhance the capacity of feature regularization, multiple convolutions with different filter sizes are applied in our recommender system. Table 1 shows the experimental results of using single convolution, dual convolutions and triple convolutions.

The experiments show that using multiple convolutions can result in better recall rate and F1-Score. However, more convolution channels mean more computational workload. Furthermore, more convolutions might extract invalid feature

TABLE 1. The experimental results of using single convolution, dual convolutions and triple convolutions.

Convolution channel configuration		Evaluation criteria	
No. of Conv.	Filter size	Recall rate	F1-Score
1	1x2	0.6936	0.6786
2	1x2、1x3	**0.7485**	**0.7032**
3	1x2、1x3、1x4	0.7128	0.6882

regularization. Through experiments, we adapt dual convolutions, as it has been proved to have highest recall rate and F1-Score.

C. EXPERIMENTS ON NETWORK OPTIMIZATIONS

In experiments, different activation functions, including ReLU, LeakyReLU and ELU, have been tested to evaluate the affection on recommendation quality. Table 2 lists the experimental results, in which the recall rate and F1-Score are related with the top 70 recommendations. The improvement rates are computed against the recall rate and F1-Score of ReLU. The experimental results in Table 2 reveals the replacement of activation function ReLU with ELU can optimize the recommender system.

TABLE 2. The experimental results of the recall rate and F1-Score related with the top 70 recommendations for different activation functions.

Activation functions	Evaluation criteria			
	Recall rate	Improvement	F1-Score	Improvement
ReLU	0.7817	-	0.7305	-
LeakyReLU (a=0.01)	0.783	+0.1663%	0.7311	+0.0823%
ELU (a=0.01)	**0.7964**	**+1.881%**	**0.7356**	**+0.6981%**

The other optimization is about pooling strategy. Seven different pooling strategies have been tested in our recommender system. For top 70 recommendations, the corresponding recall rates and F1-Score values are listed in Table 3. The hybrid pooling method which integrates max-pooling and average-pooling achieves the highest recall rate and F1-Score.

D. COMPARISON WITH OTHER RECOMMENDER ALGORITHMS

The recommender system proposed and implemented in this paper has also been experimented against traditional recommender algorithms including User-based Collaborative Filter (UserCF), Item-based Collaborative Filter (ItemCF) and Content-based Filter (CBF). The experimental results are listed in Figure 9 and Figure 10. The results the recommender model described in this paper significantly exceeds

TABLE 3. The experimental results of the recall rate and F1-Score related with the top 70 recommendations for different pooling strategies.

Pooling strategies	Evaluation criteria			
	Recall rate	Improvement	F1-Score	Improvement
Average	0.7854	-	0.7334	-
Max	0.7964	+1.4%	0.7356	+0.2999%
Hybrid	**0.8112**	**+3.285%**	**0.74**	**+0.8999%**

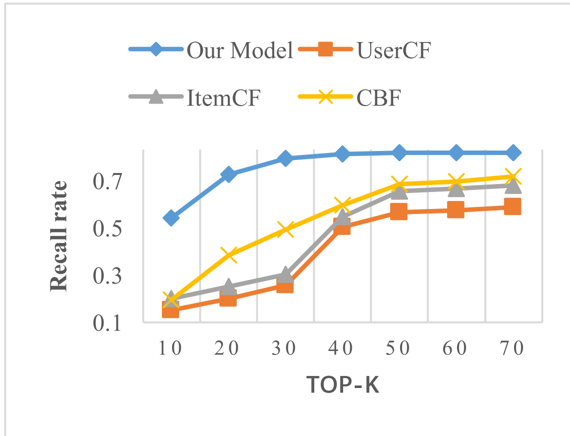

FIGURE 9. The comparison of recall rate of our model with UserCF, ItemCF, and CBF.

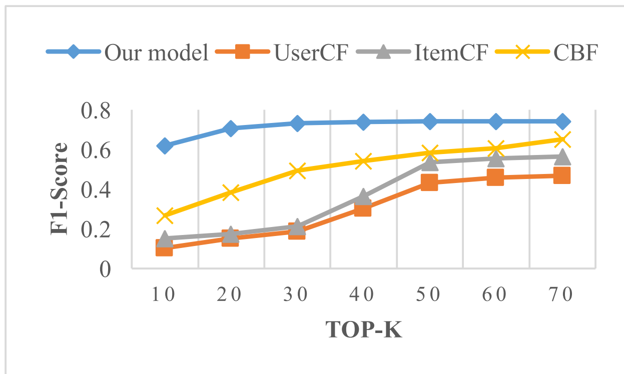

FIGURE 10. The comparison of F1-Score of our model with UserCF, ItemCF, and CBF.

UserCF, ItemCF and CBF in both the criteria of recall rate and F1-Score.

VI. CONCLUSIONS AND FUTURE WORK

This paper describes a design and implementation of recommender system for human resource recommendation. The recommender system consists of two cascaded core models: a GBDT based feature regularization model and a hybrid convolution network model. Two optimizations about activation function and pooling strategy have also been proposed in this paper. The experimental results show that our recommender system and optimizations can enhance the quality of human resource recommendation significantly.

Our current work on human resource recommendation only consider static features of the data. However, the interests or preference in job hunting might change dynamically, as the time goes. Therefore, one of the possible future work can be using networks such as RNN to accommodate sequential data. The other future work can be the investigation on facilitating parallel computing or distributed computing techniques for processing large datasets.

REFERENCES

[1] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooper. Work*, 1994, pp. 175–186.

[2] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for YouTube recommendations," in *Proc. ACM Conf. Recommender Syst.*, 2016, pp. 191–198.

[3] H. T. Cheng *et al.*, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.

[4] S. Okura, Y. Tagami, S. Ono, and A. Tajima, "Embedding-based news recommendation for millions of users," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 1933–1942.

[5] S. Zhang, L. Yao, and A. Sun. (2017). "Deep learning based recommender system: A survey and new perspectives." [Online]. Available: https://arxiv.org/abs/1707.07435

[6] K. Georgiev and P. Nakv, "A non-IID framework for collaborative filtering with restricted Boltzmann machines," in *Proc. 30th Int. Conf. Mach. Learn.* Atlanta, GA, USA, 2013, pp. 1148–1156.

[7] Z. Xu, C. Chen, T. Lukasiewicz, Y. Miao, and X. Meng, "Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, 2016, pp. 1921–1924.

[8] Y. Gong and Q. Zhang, "Hashtag recommendation using attention-based convolutional neural network," in *Proc. Int. Joint Conf. Artif. Intell.* New York, NY, USA: AAAI Press, 2016, pp. 2782–2788.

[9] H. T. H. Nguyen, M. Wistuba, J. Grabocka, L. R. Drumond, and L. Schmidt-Thieme, "Personalized deep learning for tag recommendation," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*, 2017, pp. 186–197.

[10] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, 2017, pp. 425–434. [Online]. Available: https://arxiv.org/abs/1701.04783

[11] M. Quadrana, A. Karatzoglou, B. Hidasi, and P. Cremonesi, "Personalizing session-based recommendations with hierarchical recurrent neural networks," in *Proc. 11th ACM Conf. Recommender Syst.*, 2017, pp. 130–137. [Online]. Available: https://arxiv.org/abs/1706.04148

[12] Y. K. Tan, X. Xu, and Y. Liu, "Improved recurrent neural networks for session-based recommendations," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2017, pp. 17–22.

[13] S. Wu, W. Ren, C. Yu, G. Chen, D. Zhang, and J. Zhu, "Personal recommendation using deep recurrent neural networks in NetEase," in *Proc. IEEE Int. Conf. Data Eng.*, May 2016, pp. 1218–1229.

[14] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, Dec. 1989.

[15] X. He *et al.*, "Practical lessons from predicting clicks on ads at Facebook," in *Proc. 8th Int. Workshop Data Mining Online Advertising*, 2014, pp. 1–9.

[16] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 785–794.

[17] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (ELUs)," in *Proc. Comput. Sci.*, 2015, pp. 1–14.

[18] Z. Tong, K. Aihara, and G. A. Tanaka, "A hybrid pooling method for convolutional neural networks," in *Neural Information Processing*. Kyoto, Japan: Springer, 2016, pp. 454–461.

HAOXIANG WANG received the B.S. degree in computer science and technology from the Zhejiang University of Technology, China, in 2003, and the M.S. and Ph.D. degrees from the School of Computing, University of Leeds, U.K., in 2008.

Since 2008, he has been with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His research interests include data analysis, scientific visualization, and network science.

GUIHUANG LIANG received the B.E. degree in computer science and software from the South China University of Technology, Guangzhou, China, in 2015.

He is currently pursuing the M.S. degree in computer science and technology with the South China University of Technology.

His research interests include big data analysis and application.

XINGMING ZHANG received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 1996. He is currently a Professor, a Doctoral Supervisor, and the Vice-Dean with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China.

He is a Standing Committee Member of the Technical Committee of Education and the China Computer Federation, and the Executive Director of the Computer Federation of Guangdong Province.

His research focuses on data analysis, image processing, video coding, and surveillance.