

Design of Resource Recommendation Algorithm and System Based on Heterogeneous Network Representation Learning

Jiana Bi^{1*}

¹ School of Software and Big Data, Changzhou College of Information Technology, Changzhou 213164, Jiangsu, China

Based on the historical behavior data of users, the recommendation system filters out valuable information from complex data and actively recommends resources for users, so that they can obtain resources quickly and accurately. This effectively solves the problem of information overload in the Internet era, and improves the efficiency of resource use. A heterogeneous information network can naturally model the complex objects and their rich relationships in the recommendation system, and explore the potential connections between users by analyzing different types of relationship paths. This paper focuses on two issues: Firstly, the recommendation algorithm is studied, including recommendation algorithm design, model training design, experimental data set selection, evaluation index design, and the superiority of the algorithm is verified through experiments. Then we study the recommendation system, design the simulation function which consists of the recommendation use subsystem and the recommendation management subsystem, and verify the validity of the software through the system test.

Keywords: heterogeneous network representation learning; resource recommendation algorithm; system design; random walk; Skip-Gram model; experiment and testing

1. INTRODUCTION

With the rapid development of mobile Internet and the rapid growth of network information, the way people obtain information has shifted to the Internet. If network users have clear goals and know their actual needs, they can use the network search engine to solve the search problem and screen out the relevant data from the massive amount of available information. However, “information overload” exists in a large amount of data on the Internet, which has become an important problem faced by users when they seek information [1]. With the help of information retrieval technology, the problem of “information overload”

can be alleviated to a certain extent. In the face of massive information, users will still lose their way and don't even know what they really need. Based on the historical behavior data of users and other user information, the recommendation system can filter out valuable information from the complicated data [2]. However, the filtered information is still very large. In order to effectively avoid wasting the user's time because of browsing a lot of irrelevant resources, personalized recommendation technology has emerged that can interpret the user's individual needs. Through data mining, personalized recommendation technology analyzes users' historical records, including behaviors such as usage, browsing and collection, builds users' interest patterns, makes timely adjustments and generates new recommendation results accordingly, and provides users with use

*Corresponding author. Email: bijiana666@163.com

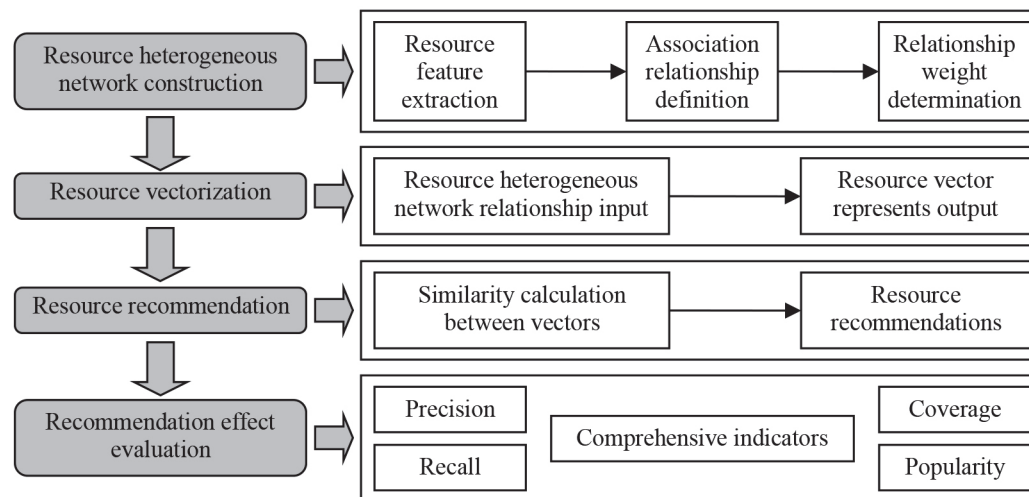


Figure 1 Resource recommendation process based on heterogeneous network representation learning.

recommendations and information services that meet their personalized preferences [3]. Heterogeneous network-based representation learning is a typical personalized recommendation technology. This paper studies the recommendation algorithm based on heterogeneous network representation learning, and uses heterogeneous network representation learning methods to learn rich structural information and semantic information in heterogeneous information networks so as to improve the performance of personalized resource recommendation.

2. RELATED WORKS

As a typical example of information-driven technology, the recommendation system has become the core service of many customer-facing systems. In order to optimize the user's information search process in the era of data flooding, various information retrieval technologies have been widely used. Although the traditional matrix decomposition algorithm has achieved success in the recommendation system, it uses only the interaction information between users and items for recommendation. When the interaction between users and items is very sparse, the accuracy of the recommendation effect will be reduced [4]. In the actual recommendation scenario, there is not only information about the user's interaction with the project, but also various auxiliary data [5]. Although the combination of matrix decomposition algorithms and this auxiliary information can improve the recommendation performance, it is still a challenge to utilize this information effectively in the recommendation system due to the heterogeneity and complexity of the auxiliary data [6].

As a promising development direction, a heterogeneous information network can naturally model complex objects and their rich relationships in the recommendation system. Different nodes in the network represent different entities, and the links between nodes represent different relationships. Due to their flexibility in modeling various heterogeneous data, heterogeneous information networks have been recommended for systems to model rich auxiliary data [7,8].

Most of the existing recommendation algorithms based on heterogeneous information networks usually abandon the node information inside the meta-path when learning the node representation, and consider only the start and end nodes of the meta-path, resulting in information loss, and affecting the accuracy of the recommendation task. The deep learning recommendation algorithm based on graph attention network uses hierarchical aggregation to aggregate all aspects of information [9]. For node-level aggregation, we design the meta-path instance coding function to encode the node information inside the meta-path, and introduce the attention mechanism to distinguish the importance of different neighbors based on meta-path. In terms of semantic level aggregation, attention mechanisms are introduced to distinguish the importance of each meta-path. Based on the random walk algorithm, the similarity between users and resources can be calculated from the perspective of graph. How to make full use of the information generated in the random walk process to improve the accuracy of recommendation results is the focus of relevant recommendation algorithm research [10,11].

3. RECOMMENDATION ALGORITHM DESIGN AND EXPERIMENT

3.1 Recommendation Algorithm Design

The resource recommendation algorithm based on heterogeneous network representation learning proposed in this paper includes four parts: resource heterogeneous network construction, resource vector quantization, resource recommendation and recommendation effect evaluation. The resource recommendation process is shown in [Figure 1](#).

The resource recommendation process shown in [Figure 1](#) is explained as follows: First, select a variety of resource features to build a resource heterogeneous network, and define the multidimensional association relationship between the features. Based on heterogeneous network representation learning method, a semantic vectorized representation of

each resource is constructed by integrating multiple resource feature information. Secondly, from the perspective of semantic association, the degree of correlation is resources is calculated, and the similarity between resources is calculated by cosine similarity to produce the recommendation of related resources. Finally, by using “Precision, Recall, Comprehensive indicators, Coverage, Popularity” and other indicators, the recommendation algorithm proposed in this paper is compared and analyzed with other conventional recommendation algorithms to verify its feasibility and effectiveness.

- (1) Vectorization of resources. Based on the method of representation learning in a heterogeneous network, the feature information of multiple resources can be fused, and a low-dimensional dense semantic vector representation can be constructed for each resource in heterogeneous network. The random walk based on meta-path is used to extract features and generate many meaningful node association sequences [12]. Different from the ordinary random walk mode, meta-path imposes certain constraints on the walking path, and the walking path distribution is calculated with:

$$P(n_{x+1} = v' | n_x = v, p) = \begin{cases} \frac{1}{|N^{T_{m+1}}(v)|}, & (v, v') \in \varepsilon \text{ and } \phi(v') \\ & = T_{m+1} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where n_x is the x -th node of the wandering path, the type of node v is T_m , and $N^{T_{m+1}}(v)$ is the first-layer neighbor of node T_{m+1} . The walk continues along the meta-path until it reaches the specified length. Given multiple meta-path paths, multiple sequences can be generated by random walks of the meta-path. For a resource node, fusion transformation is performed on the learned resource node representation using the fusion function formula:

$$g(\{e_v^{(p)}\}) = \sigma \left(\sum_{p=1}^{|P|} w_v^{(p)} \sigma \left(A^p e_v^{(p)} + b^{(p)} \right) \right) \quad (2)$$

- (2) Similarity calculation. After each resource in a heterogeneous network is represented as a fixed-dimensional vector, this vector not only contains the feature information of the resource itself, but also includes the correlation between resources. Therefore, the problem of related resource recommendation becomes a problem of calculating the similarity of resource vectors [13]. There are many methods for calculating the similarity between vectors, and the cosine distance is more suitable for solving the problem of outliers and data sparsity. The vector representation of the two resources is:

$$\begin{cases} \vec{x} = (x_1, x_2, x_3, \dots, x_n) \\ \vec{y} = (y_1, y_2, y_3, \dots, y_n) \end{cases} \quad (3)$$

The formula for calculating the similarity of the two vectors is:

$$\cos(\theta) = \frac{\vec{x} \times \vec{y}}{|\vec{x}| \times |\vec{y}|} = \frac{\sum_{i=1}^n (x_i \times y_i)}{\sqrt{\sum_{i=1}^n (x_i)^2} \times \sqrt{\sum_{i=1}^n (y_i)^2}} \quad (4)$$

3.2 Model Training Design

After the node sequence is obtained through a random walk, the Skip-Gram model is trained. The Skip-Gram model predicts the surrounding words by one word [14]. The result of training is that the representation maps node v to a d -dimensional vector space, f is a dimensional matrix, and every resource node is represented by a d -dimensional real value vector. Assuming that node x is the starting point of a random walk, in order to enable the node vector to retain as much network structure information as possible, the goal of model training is to maximize the logarithmic probability that the relevant node y appears on the random walk sequence starting from node x :

$$\max_f \sum_{x \in V} \log \Pr(R\tau(x) | f(x)) \quad (5)$$

Formula 5 is a random walk sampling sequence taking node x as the starting point. Nodes in the sequence are not necessarily directly connected to node x , but are only neighbors obtained by the sampling method. The generation probabilities of each node on the random walk sequence are independent of each other, so:

$$\Pr(R\tau(x) | f(x)) = \prod_{\tau_i \in R\tau(x)} \Pr(\tau_i(x) | f(x)) \quad (6)$$

In the above formula, assuming that two node projections have the same influence on each other in vector space, the formula for calculating conditional probability is as follows:

$$\Pr(\tau_i(x) | f(x)) = \frac{\exp(f(\tau_i) \bullet f(x))}{\sum_{y \in V} \exp(f(y) \bullet f(x))} \quad (7)$$

Let:

$$Z_x = \sum_{y \in V} \exp(f(y) \bullet f(x)) \quad (8)$$

Synthesizing the above formula, the final objective function is obtained:

$$\max_f \sum_{x \in V} \left(-\log Z_x + \sum_{\tau_i \in R\tau(x)} f(\tau_i) \bullet f(x) \right) \quad (9)$$

In order to reduce the time required to calculate Z_x , the negative sampling method is used to approximate the calculation, and finally the objective function is optimized by random gradient descent to solve the parameters.

3.3 Experimental Data Set Selection

The MovieLens dataset, created by the GroupLens project team at the University of Minnesota, was used for this

Table 1 Comparative analysis of experimental results.

No	Model	Precision@10	Recall@10	FI@10	Coverage@10	Popularity@10
1	Content-Based	0.2554	0.1411	0.1562	0.2671	7.7492
2	Item-Based	0.2315	0.1543	0.1736	0.2834	7.8965
3	Matrix Factorization	0.2174	0.1238	0.1852	0.3685	8.2138
4	Actor-Critic	0.2837	0.1614	0.2077	0.3261	8.0215
5	HERec	0.2681	0.1522	0.2134	0.3918	8.4174

experimentas it is widely used by researchers. The MovieLens dataset has a long history and is open source and shared, making it an indispensable resource in the field of recommendation systems. The MovieLens dataset is not only suitable for academic research, but can also be used for educational training and commercial applications to help developers achieve better results in multiple fields. The MovieLens data set contains three data files: U.TEM, U.SER and u.data [15]. It is available in two versions: small data capacity and large data capacity. The small data volume version was chosen for this study. The MovieLens data set was divided into two sets, the training set and the test set, in the proportion of 70% and 30% respectively, and the sparsity of the score data was 90.57%. It can be seen that the data sparsity in the data set is relatively high.

3.4 Evaluation Index Design

For Top-N recommendations, when resources are recommended for a user, a list of n resources is generated. Let U represents the set of users, $R(u)$ represents the list of resources recommended by user u, and $T(u)$ represents the list of uses of user u in the test set. The following five indicators are generally used to evaluate the algorithm:

- (1) Precision, the proportion of resources that users are interested in. The higher the accuracy, the stronger the ability of the recommendation algorithm to align with the user's interests. It is defined as:

$$Precision@K = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|} \quad (10)$$

- (2) Recall, which refers to the ratio of the number of resources used by users in the recommended list to the number of resources used by users in the test set, is used to evaluate the algorithm's ability to filter resources. It is defined as:

$$Recall@K = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|} \quad (11)$$

- (3) Weighted average of comprehensive indicators, accuracy rate and recall rate. The higher the value, the more robust the model. It is defined as:

$$FI@K = \frac{Precision * Recall}{Precision + Recall} * 2 \quad (12)$$

- (4) Coverage, used to measure the "long tail" mining ability of the recommendation algorithm. In the recommendation scenario, the general service has a long tail distribution; that is, a small number of popular resources contribute most of the exposure. At the end of the resource, although the audience is weak, the needs of specific user groups can be met. How to mine these non-popular resources accurately and push them to specific user groups is one of the important problems that recommendation algorithms need to address. Coverage is defined as:

$$Coverage@K = \frac{|U_{u \in U} R(u)|}{|I|} \quad (13)$$

- (5) Popularity indicates the popularity of resources in the recommendation list. Under the condition that the algorithm recommendation accuracy does not decline, the lower the popularity of the recommendation list, the higher the novelty of the algorithm, and the stronger the ability to recommend resources to users. Popularity is used to measure the ability of a recommendation algorithm to mine unknown resources for users. It is defined as:

$$Popularity@K = \sum_{u \in U} \frac{\sum_{i \in R(u)} \log(1 + N(i))}{|R(u)|} \quad (14)$$

3.5 Experimental Results and Analysis

HERec is a resource recommendation algorithm based on heterogeneous network presentation learning designed in this paper. The validity of HERec, HERec and other commonly used recommendation algorithms is tested separately with MovieLens dataset. Other commonly used recommendation algorithms include "Content-Based, Item-Based, Matrix Factorization, Actor-Critic" for the Top-10 recommendation of resources. The comparative analysis of algorithms is shown in Table 1.

As can be seen from Table 1, for Precision@10 indicator, the top three ranking models are respectively "Actor-Critic, HERec, Content-Based", and HERec is in the second place. For the Recall@10 indicator, the top 3 models are "Actor-Critic, Item-Based, HERec", with HERec in the 3rd place. For the FI@10 indicator, the top 3 models are respectively "HERec, Actor-Critic, Matrix Factorization", with HERec in the first place. For the Coverage@10 indicator, the top 3 models are "HERec, Matrix Factorization, Matrix Factorization", with HERec in the first place. For the Popularity@10 indicator, the top three models are respectively

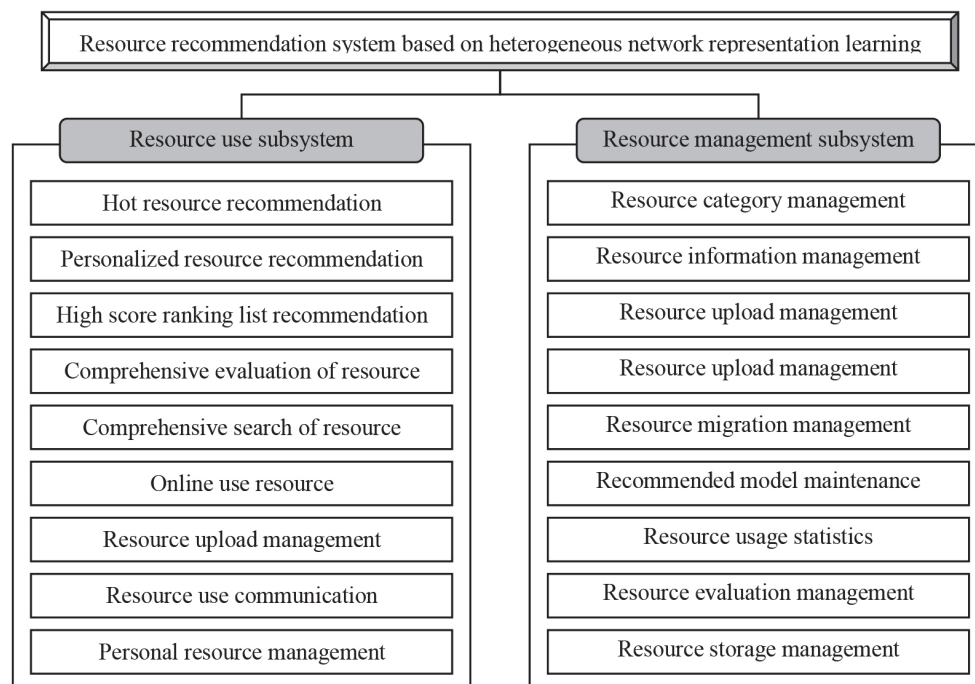


Figure 2 System functions of resource recommendation system based on heterogeneous network representation learning.

“HERec, Matrix Factorization, Actor-Critic”, with HERec in the first place. Therefore, in summary, compared with “Content-Based, Item-Based, Matrix Factorization, Actor-Critic” and other recommendation algorithms, HERec has obvious advantages.

4. RECOMMENDATION SYSTEM DESIGN AND TESTING

4.1 System Function Design

The functional design of a system involves the transformation of the requirements of a software item into an architecture that describes its top-level structure and identifies the individual software components. All requirements for the software item must be assigned to the software part and are further refined to facilitate detailed design. The functional design of the system is evaluated in terms of: traceability of software requirements, external consistency with structural design, internal consistency between software components and software units, suitability of applied design methods and standards, feasibility of testing, feasibility of operation and maintenance. The resource recommendation system function based on heterogeneous network representation learning includes not only the basic resource recommendation function, but also the related functions of use and maintenance. The design results are shown in [Figure 2](#).

- (1) Resource-use subsystem [16,17]. The resource-use subsystem is mainly oriented to end users, the core function is resource recommendation, and other functions include resource evaluation and use. Among them, Hot resource recommendation can quickly reflect the focus

of attention. Personalized resource recommendation: High score ranking list recommendation indicates the best available resources. Online use resource: Use resources directly through the tools provided by the system. Resource upload management, which can be downloaded from the network and used offline. Personal resource management: manages the resources collected by the system.

- (2) Resource management subsystem. Resource category management refers to the classification of resources for easy management. Resource information management provides basic information for resource management and use. Resource upload management: The administrator uploads the collected resources to the system. Resource upload management can be downloaded from the network and used offline. Resource migration management: migrates resources from the database to other storage media. Recommended model maintenance: the parameters of the recommended model are maintained. Resource usage statistics provides suggestions for resource creation and collection. Resource evaluation management: comprehensively manages the evaluation of information to eliminate malicious evaluation. Resource storage management: manages storage media and optimizes storage policies.

4.2 System Testing and Analysis

System testing is the best way to verify the effectiveness of software, and it is also the key to software quality assurance, which directly affects software quality evaluation. Software testing should not be regarded as an independent stage of software development, but should be carried out

Table 2 Function test of “Comprehensive search of resource” module.

1	Test case	The “Comprehensive search of resource” module consists of three parts: user input of search conditions, information retrieval from database and output of search results.
2	Test process	First, the user enters the search criteria, and in order to test adequately, a combination of various criteria needs to be entered. Then, the system connects to the database and queries the records that meet the conditions in the database. Finally, the records that meet the conditions retrieved from the database are output.
3	Expected result	The search criteria can be entered smoothly. If a search criterion is entered incorrectly, the system can give a prompt and allow re-entry. By default, the system outputs the result on the screen. After the result is output on the screen, you can select multiple modes, such as file, web page, and print.
4	Test result	No errors occurred during the test, the input and output are correct, and the retrieval from the database is fast.
5	Test conclusion	The system functions meet the expected requirements and can be delivered to users, but some development documents still need to be perfected.

during all stages of software development. All software defects detected by system testing should be rectified until the software is error-free. Software testers should master how to reduce software defects to a manageable range, determine the risks that software defects may pose to customers, and find the most appropriate test cases. After development, the software must be tested to ensure that it meets all functional requirements. The design of test cases should make every software function of the requirement specification able to be executed and confirmed. This system has many functions. Only the “Comprehensive search of resource” module is taken as an example for functional testing. The test case design and results are shown in [Table 2](#).

5. CONCLUSIONS

In order to solve the problem of information overload, recommendation systems have emerged. By constructing the transition probability matrix in the random walk model, the time complexity of the random walk algorithm is reduced. Finally, the target users are recommended and the final resource recommendation list is obtained. Heterogeneous networks are used for modelling, and network representation learning is used to extract and utilize rich information on heterogeneous networks so as to improve the performance of recommendation and classification. It takes more time to calculate the cosine similarity, which affects the execution efficiency of the program and the recommendation. In future work, we hope to develop a better model to address this issue. The algorithms proposed in this paper are all based on shallow network representation learning technology for recommendation. In the future, we intend to use graph representation learning technology to mine deeper connections between data and improve the accuracy of recommendation.

REFERENCES

1. Apuke D O, Omar B, Tunca A E, et al. Information overload and misinformation sharing behaviour of social media users: Testing the moderating role of cognitive ability[J]. *Journal of Information Science*, 2024, 50(6): 1371–1381.
2. Y. Zhang, X. Liu, H. Wang. Heterogeneous Graph Neural Network-Based Resource Recommendation for Intelligent Engineering Systems[J]. *Engineering Intelligent Systems*, 2024, 22(2): 45–58.
3. Pelánek R, Effenberger T, Jarušek P. Personalized recommendations for learning activities in online environments: a modular rule-based approach[J]. *User Modeling and User-Adapted Interaction*, 2024, 34(4): 1399–1430.
4. Olubusola F I. Matrix Factorization in Recommender Systems: Algorithms, Applications, and Peculiar Challenges[J]. *IETE Journal of Research*, 2023, 69(9): 6087–6100.
5. S. Li, J. Chen, L. Zhao. Network Representation Learning for Personalized Resource Allocation in Smart Industrial Systems[J]. *Engineering Intelligent Systems*, 2025, 23(1): 12–25.
6. Yohanssen P, Angrainy R S, Dedi C, et al. Restricted Boltzmann Machine and Matrix Factorization-Alternating Square Algorithm for Development Tourist Recommendation System[J]. *Journal of Physics: Conference Series*, 2022, 2394(1): 1–11.
7. Naina Y, Sukomal P, Kumar A S. Diversified recommendation using implicit content node embedding in heterogeneous information network[J]. *Multimedia Tools and Applications*, 2023, 83(7): 20605–20635.
8. Lei T, Daofu G, Jinmao X, et al. Meta-path fusion based neural recommendation in heterogeneous information networks[J]. *Neurocomputing*, 2023, 529: 236–248.
9. Vinita V, Bargavi M S. Transforming recommender system by integrating attention-based neural network for multimodal sentiment analysis using artificial intelligence[J]. *International Journal of Computer Applications in Technology*, 2024, 74(1–2): 136–145.
10. Mozghan T, Mahdi F, Abbas A, et al. Point-of-interest recommendation using extended random walk with restart on geographical-temporal hybrid tripartite graph[J]. *Journal of Spatial Science*, 2023, 68(1): 71–89.
11. Yang Q, Wang H, Bian M, et al. Incorporating Reverse Search for Friend Recommendation with Random Walk[J]. *The International Arab Journal of Information Technology*, 2020, 17(3): 291–298.
12. Verma K A, Jadeja M, Jayaswal S. RW-HeCo: A random walk and network centrality-based graph neural network for community detection in heterogeneous networks[J]. *Multimedia Tools and Applications*, 2024, (prepublish): 1–24.

13. Sajjadi S Z, Esmacili M, Arani G M, et al. An efficient graph embedding clustering approach for heterogeneous network[J]. The Journal of Supercomputing, 2024, 80(13): 19562–19591.
14. Celik E, Omurca I S. Skip-Gram and Transformer Model for Session-Based Recommendation[J]. Applied Sciences, 2024, 14(14): 6353–6353.
15. Talha M A, Syed U, Shakil, et al. Comparative Analysis of Machine Learning based Filtering Techniques using MovieLens dataset[J]. Procedia Computer Science, 2021, 194: 210–217.
16. J. Zhang. Design of Personalized Recommendation System for Online Learning Platform Based on Data Mining[J]. Software, 2023, 44(12): 44–46.
17. J. J. Liang, G. Q. Guo, Z. H. Ren. Design of personalized teaching system based on intelligent recommendation[J]. Automation & Instrumentation, 2023, 43(10): 126–130.



Jiana Bi was born in 1979 in Jiamusi, Heilongjiang, China. She is an associate professor at the Changzhou College of Information Technology. She received her master's degree and doctor's degree from Harbin Institute of Technology. Her research interests include big data and network security.

