Application of Collaborative Filtering Optimization Algorithm Based on Semantic Relationships in Interior Design

Kai Zhao[1], Lei Wang[2]*
College of Fine Arts and Art Design, Henan Vocational University of Science and Technology, Zhoukou, 466000, China[1]
School of Residential Environment and Design, Shijiazhuang Institute of Technology, Shijiazhuang, 050228, China[2]

***Abstract*—Due to the diversity of interior design, it is difficult for users to mine target data, so personalized recommendation systems for users are particularly important. Therefore, an optimized collaborative filtering recommendation system is proposed. Firstly, a random walk recommendation model based on category combination space is constructed, abandoning the traditional flat relationship connection and using Hasse diagram to achieve one-to-one mapping between items and types. The semantic relationship and distance are defined. Finally, a basic recommendation framework for random walks is established based on data such as jump behavior. Next, the potential semantic relationships between entities are explored, and a lightweight knowledge graph is proposed to define the social and explicit relationships between entities. Finally, the short-term features of the project are obtained using deep collaborative filtering technology, and a deep collaborative filtering temporal model based on semantic relationships is constructed. In subsequent validation, these experiments confirmed that under the vector dimension of 10, the average HR@K and NDCG@K were 6.9% and 12.9% higher than the other models. Therefore, the collaborative filtering recommendation model based on semantic relationships proposed in the study is reliable.**

***Keywords*—Semantic relationships; category combination space; random walks; collaborative filtering; temporal recommendations**

I. Introduction

The Internet has brought great influence to human society, which is not only a simple query tool, but also a method to tap the potential interests of users. This change is to adapt to the gradual increase in the amount of big data currently available, making it difficult for users to choose their preferences among the vast amount of information. At this time, the algorithm can build a recommendation model for users using previous preferences, and recommend relevant preference data and potential interest preference data for users [1]. Collaborative Filtering (CF) is the most commonly used recommendation algorithm at present. The key to this technology is searching for neighboring users and calculating similarity. However, the cold start and high computational dimension limit the improvement of recommendation performance. More researches focus on the introduced information. Project type is an important additional data, which can alleviate the sparse data. However, the limitation of its flattened organization is also one of the problems to be solved [2]. As a semantic network, knowledge mapping can construct semantic relations to connect entities, which effectively solves the neglected explicit and implicit interactions between information in the traditional model [3]. In recent years, recommendation systems have made significant progress, from early CF algorithms to advanced technologies such as deep learning and graph neural networks. The accuracy and efficiency of recommendation systems continue to improve. However, traditional recommendation models often rely on flat relational connections and fail to fully explore the complex semantic relationships between items and types, resulting in insufficient recommendation accuracy and semantic understanding. Most studies often lack temporal considerations when dealing with dynamically changing user needs, making it difficult to adapt to real-time changes in user interests. Therefore, a deep CF recommendation system based on semantic relationship is proposed. It is expected to make significant contributions to the development of recommendation systems and provide more efficient and accurate tools for solving information overload problems. The innovation of this study lies in proposing a random walk recommendation model based on type combination space, which abandons the traditional flat relationship connection and uses Hasse diagram to achieve one-to-one mapping between items and types, providing a new perspective and idea for recommendation systems.

The research includes six sections. Firstly, the research status of recommendation system is introduced. Secondly, the designed CF recommendation system based on semantic relationship is described in Section III. Section IV verifies the recommendation model. Results and discussion is presented in Section V and finally, the paper is concluded in Section VI.

II. Related Works

In various current online platforms, personalized recommendation systems have gradually become a mainstream and necessary part. Gou L et al. believed that the marketing model in the communication field shifted towards socialization. Therefore, a recommendation model based on social analysis was proposed. Initially, a two-layer communication network for users and platforms was constructed through historical information, followed by a network of similar users and platforms using CF. Finally, a bipartite graph weighting method was used to achieve project recommendation, and the feasibility of the model was experimentally verified [4]. Wu et al. believed that knowledge maps had a good auxiliary effect on recommending a large amount of data content. Therefore, a context aware algorithm based on Knowledge Graph (KG) was proposed, while gaining the advantages of both propagation and path-based technologies. The user preferences were represented through rules, and an automatic rule mining model was used in entity interaction. Finally, the performance of the model was further optimized through the local features in the neighborhood [5]. Yan et al. believed that the data density was the key factor affecting the CF recommendation performance. Therefore, a neighboring clustering method based on granular computing technology was introduced, and a CF optimization scheme based on coverage rough granular computing was proposed. The basic framework was built on user project scoring data, and local rough granularity sets were established through user preference thresholds, solving the data sparsity. The accuracy improvement in simulation experiments verified the reliability of the optimization model [6]. Miao et al. believed that the subjectivity of social networks has driven the improvement of their platforms. A recommendation algorithm based on user profiles was proposed and the user profile application model on short video platforms was described. These experiments confirmed that in four different platforms, the user satisfaction rate with recommended content was greater than 75% [7].

Liang et al. believed that the recommendation balance and trust of educational resources were generally poor. In response to this phenomenon, a trust relationship recommendation model was proposed. Firstly, support vector machines were used to classify educational resources and remove duplicate and useless data. Based on the remaining data, resource features were extracted. Finally, Kalman filtering was introduced to denoise and reduce the dimensionality of the features, constructing the recommendation model. These experimental data confirmed that the equilibrium degree reached 96 [8]. Qi et al. proposed to apply graph neural networks to recommendation algorithms, using user historical data and second-order social data to learn user project feature representation. Multiple graph attention network modules were utilized to construct the model. Finally, simulation experiments were conducted to verify the effectiveness of the model [9]. Wang et al. believed that intelligent English has solved a large number of educational challenges. The Internet of Things serves as the technical foundation in this field and can effectively collect and manage information. An attention mechanism module was introduced to propose an educational resource recommendation algorithm, and a deep collaborative recommendation model was constructed. The effectiveness of the method was verified through experiments [10]. The summary of the research on recommendation models mentioned above is shown in Table I.

In summary, although recommendation models have received widespread attention from researchers, current recommendation models still have problems such as not delving into the complex semantic relationships between items and types, and requiring a large amount of data and computing resources. Therefore, this study introduces Hasse diagram, lightweight KG, and deep CF techniques to comprehensively consider the semantic relationships between items and types, as well as the social and explicit implicit relationships between entities. A semantic-based optimized deep CF algorithm is proposed to improve the accuracy and robustness of the recommendation system. This study has significant advantages over previous research in terms of the depth of recommendation models, semantic relationship mining, temporal considerations, and generalization ability.

TABLE I. SUMMARY TABLE

Researcher	Method	Insufficient
Gou L	Recommendation Model Based on Social Analysis	Not fully considering the semantic relationship and distance between projects and types
Wu C	Context Aware Algorithm Based on Knowledge Graph	Not delving deeply into the social and explicit implicit relationships between entities
Yan HC	Collaborative Filtering Optimization Based on Covering Rough Particle Computation	Not considering the complex semantic relationship between projects and types
Miao R	Recommendation Algorithm Based on User Profile	Lack of in-depth exploration of the complex semantic relationships between projects and types
Liang X	Trust Relationship Recommendation Model	Requires a large amount of data and computing resources
Qi W	Recommendation Algorithm Based on Graph Neural Network	Strong dependence on secondary social data
Wang F	Deep Collaboration Recommendation Model	Failed to fully explore the complex relationship between users and projects

III. APPLICATION OF CF OPTIMIZATION ALGORITHM INTEGRATING SEMANTIC RELATIONSHIPS IN INTERIOR DESIGN

The interior design data are relatively rich, making it difficult for users to accurately search for more types of preferences and to mine potential preferences. Therefore, it is necessary to optimize it to achieve more accurate user recommendations. The recommendation algorithm based on project types is a common personalized strategy. Traditional model building methods based on type similarity and preference types often overlook the relationship structure between types, which hinders the recommendation accuracy [11-12]. This study proposes a random walk based recommendation model based on Category Combination Space (CCS), and introduces a temporal model based on semantic relationships to achieve final recommendations through deep CF.

A. *A Random Walk Recommendation Algorithm Defined by Semantic Relationship*

CCS is a collection of types contained in multiple projects, which Hasse diagrams to map projects and types one by one, including various relational structures such as up and down, same layer, and skip. Eq. (1) represents each element in space.

$$\begin{cases} U = \{u_1, u_2, ..., u_m\} \\ I = \{i_1, i_2, ..., i_m\} \\ C = \{c_1, c_2, ..., c_m\} \end{cases} \quad (1)$$

In Eq. (1), $U / I / C$ represent a collection of users, projects, and types, respectively. The relationship between projects and types is one-to-many. CCS considers all types corresponding to each project as a whole and obtains a one-to-one correspondence diagram, as shown in Fig. 1.

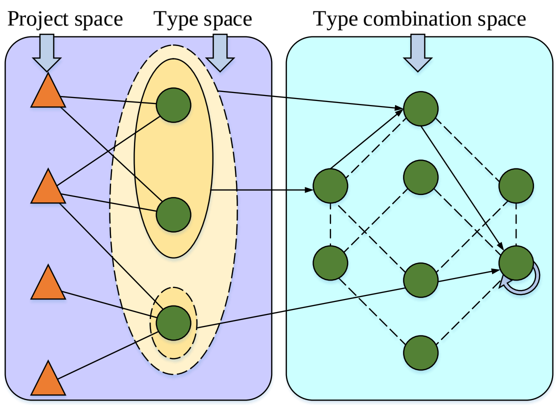


Fig. 1. Category combination space transformation structure.

CCS is essentially a partially ordered set of project types, belonging to an efficient Hasse graph. In actual operation, only some elements in the space are applied, and their project set is a subset of the partially ordered set. CCS achieves user preference mining through the jump sequence formed by user browsing behavior on nodes and the semantic relationship between jump nodes. Among them, the semantic relationships of browsing behavior include four categories, corresponding to the reduction, expansion, stability, and jump of user interests [13]. The qualitative description of relationships lays a solid foundation for mining dynamic preferences, while semantic distance further describes the preference change. This can be analyzed from two aspects: the true distance on the Hasse diagram and the changes in the number of basic and common elements [14]. Eq. (2) is the minimum distance between Hasse graph types.

$$d_L(\tilde{c}_i, \tilde{c}_j) = 2\left|\tilde{c}_i \cup \tilde{c}_j\right| - \left|\tilde{c}_i\right| - \left|\tilde{c}_j\right| \quad (2)$$

In Eq. (2), d_L is the link distance. This distance value represents the hierarchical span between types. If the span is small, the corresponding interests are stable. $(\tilde{c}_i, \tilde{c}_j)$ represents different types of combinations. Eq. (3) is the preference distance.

$$d_p(\tilde{c}_i / \tilde{c}_j) = \begin{cases} 0 & \tilde{c}_i = \tilde{c}_j \\ \dfrac{1}{|C|} & \tilde{c}_i \succ \tilde{c}_j \\ \left|\tilde{c}_j - \tilde{c}_i\right| & \tilde{c}_i \prec \tilde{c}_j \\ |C| + d_L(\tilde{c}_i / \tilde{c}_j) & other \end{cases} \quad (3)$$

The preference distance represents the changes in type combinations. There are four situations: the combination remains unchanged, the combination becomes larger, the combination becomes smaller, and the combination is located in different projects [15]. The last one has the highest preference distance and the most significant change in interest. User browsing graph $G = \left(\tilde{C}, E, A^E\right)$ is introduced to collect semantic relationships, distance sizes, and temporal features between types. $\tilde{C}, E, A^E$ represent the vertex, edge, and edge feature matrix of the graph, respectively. PageRank based on random walk was introduced to construct an interest preference model. The core idea of this algorithm is to evaluate the importance of a page based on its quantity and quality of links in Fig. 2.

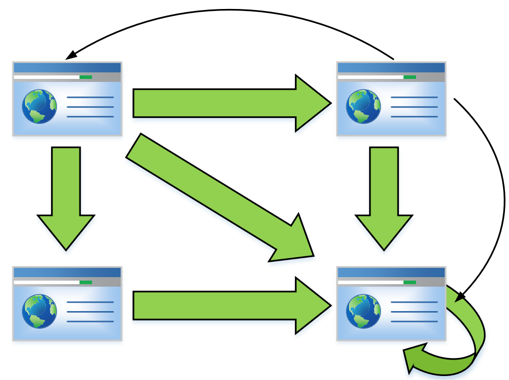


Fig. 2. Evaluation chart of the number of webpage links.

The number of times a webpage is linked is proportional to its importance, and its corresponding PageRank will also be improved. This transition probability design is the key to this method. The edge features of user browsing graphs contain a large amount of data such as node semantic relationships and preference distances [16]. The transfer matrix needs to transfer these data into probability, as shown in Eq. (4).

$$p_{ij}(\omega) = \frac{\omega^T a_{ij}^E}{\sum_j \omega^T a_{ij}^E} = \frac{\sum_k \omega_k a_{ijk}^E}{\sum_j \sum_k \omega_k a_{ijk}^E} \quad (4)$$

In Eq. (4), $p_{ij}(\omega)$ represents each element in the transition matrix. a_{ij}^E represents the edge feature vector. ω represents the weight of each edge. k represents the number of neighbors. The preference vector is the key to personalized random walk, as shown in Eq. (5).

$$q_u(i) = \frac{b_u(i)}{\sum_{k=1}^{|\tilde{C}|} b_u(k)} \quad (5)$$

In Eq. (5), b_u represents the browsing vector of user u. The objective function can be obtained through the iterative process, as shown in Eq. (6).

$$\min_{\omega,\pi}(\omega^{(s)}, \pi^{(s)}) = \left\|\pi^{(s+1)} - \pi^{(s)}\right\|^2 \quad (6)$$

In Eq. (6), $\pi = (\pi_1, \pi_2, ..., \pi_n)^T$ represents the scoring vector of the type combination. $\pi^{(s+1)}$ represents the iteration of a random walk. $\left\| \pi^{(s+1)} - \pi^{(s)} \right\|^2$ represents the loss function, and $\left\| \pi^{(s)} \right\|_1 = 1$. The study introduces gradient descent method to minimize the objective function, while using cosine similarity to compare user similarity. Fig. 3 shows the overall CCS-based random walk algorithm.

According to Fig. 3, this model initializes the user preference vector through information such as rating matrix, user browsing graph, and weight parameters. Personalized preferences for each user are calculated based on their browsing data. Subsequently, the transfer matrix, objective function, and weight values are updated. When the scoring vector matches the iterative function, the final scoring prediction matrix is output.

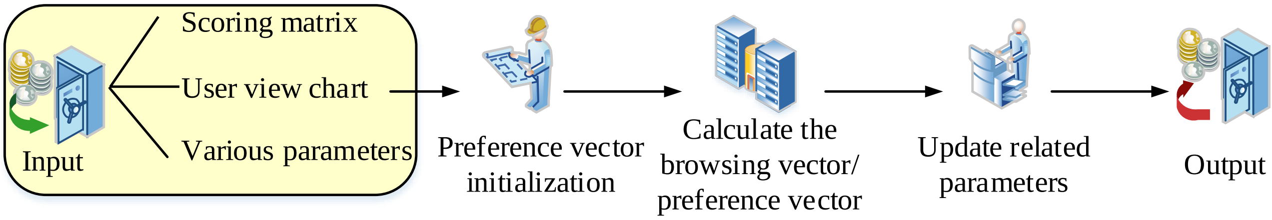


Fig. 3. Random walk algorithm model based on type combination space.

B. Construction of Semantic Relationship Temporal Recommendation Model Based on Deep CF

KG based on semantic relationships can extend project data to mine deep relationships between project types. However, this also means that a large number of databases are required as support. Due to the limitations of mature and publicly available databases and the large volume of training features, the feasibility of actual operation is low. Moreover, using only semantic relationships as predictive information cannot mine the interactions between user items [17]. Therefore, a timing lightweight KG is proposed in Fig. 4.

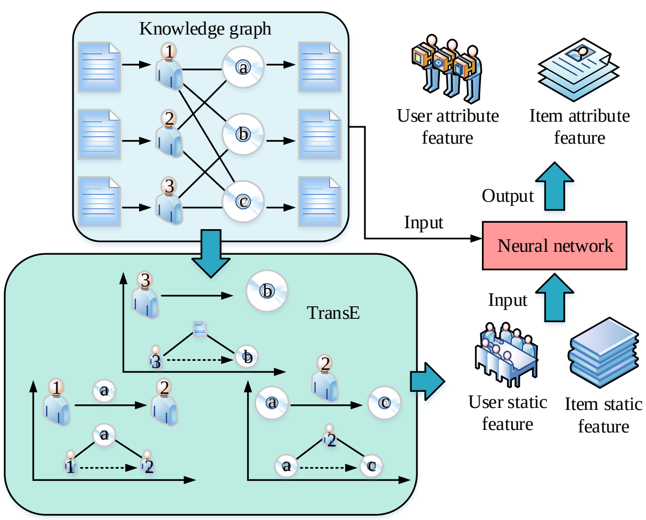


Fig. 4. Lightweight knowledge graph structure based on time series recommendation.

The model represents the explicit and implicit interactions between entities, users, and projects in KG by defining semantic relationships. The user rating of the project will create an edge between the two, and the attribute features of the two are linked to the corresponding entities [18]. The static attribute refers to the semantic feature representation vector of the two, which is obtained by associating the entity neighborhood with the entity's first and second order neighbors. The attribute triplets of both are defined, including rating behavior, user relationships, project relationships, and attribute characteristics. Eq. (7) is the learning optimization probability of the rating behavior triplet.

$$P(u,r,v) = \sum_{(u,r,v^+)\in KG(u,r,v^-)\in KG^-} \sum \sigma(g(u,r,v^+) - g(u,r,v^-)) \quad (7)$$

In Eq. (7), $\sigma(x) = 1/(1+\exp(x))$ represents the sigmoid function. $g(\cdot)$ represents the energy function. The optimization probability $P(u_i, v, u_j)$ for user relationship triplets and project relationship triplets $P(v_i, u, v_j)$ are the same as Eq. (7). Eq. (8) is the energy function.

$$g(u,r,v) = \left\| u + r - v \right\|_{L_1/L_2} + b_1 \quad (8)$$

In Eq. (8), b_1 represents the bias constant. The attribute relationship between users and projects is essentially a bandit problem. Based on the representation vector, the neural network is selected for training. The optimization probabilities of user and project attribute triplets are expressed as $P(u,a,e)/P(v,a,e)$, and their definitions are similar. Eq. (9) is the calculation of the former.

$$P(u,a,e) = \sum_{(u,a,e^+)\in KG(u,a,e^-)\in KG^-} \sum \sigma(h(u,a,e^+) - h(u,a,e^-)) \quad (9)$$

In Eq. (9), $h(\cdot)$ represents the classification function expressed by in Eq. (10).

$$h(u,a,e) = \left\| f(uW_a + b_a) - e_{ae} \right\|_{L_1/L_2} + b_2 \quad (10)$$

In Eq. (10), b_a, b_2 represent the learning parameter and bias constant, respectively. $f(\cdot)$ represents a nonlinear transformation function. e_{ae} represents the vector of attribute a. Due to the changing interests of users, popular elements can affect recommendation results, while static features do not change over time [19]. Therefore, short-term features of entities are introduced for improvement. Model features can include static and attribute features based on KG semantic relationships, as well as short-term preference features generated by short-term browsing. The latter is implemented through Long Short-Term Memory (LSTM), which extracts popular items from the current browsing set. Taking the static and attribute features of the project as input, learning effectively reduces the computational pressure of the model in the attention mechanism. Finally, the recommendation is achieved through the long-term and short-term feature connections between entities in a multi-layer perception, as displayed in Fig. 5.

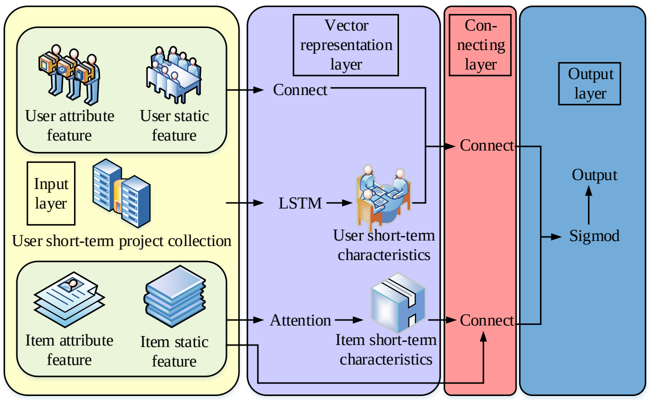


Fig. 5. LSTM recommendation model based on deep collaborative filtering.

The popularity of projects varies. A high level of popularity indicates that the project has a significant impact and requires learning through attention mechanisms. On the basis of maintaining sequential information, the attention mechanism extracts element relationships. The nearly 1-hour browsing sequence is used as input to match the entire project. The specific input data is the attributes and static features of the project. The weighted sum of each sequence is output, and Eq. (11) is the weight matrix.

$$T_i^t = z^T \tanh(W_c c_t + W_y y_i) \tag{11}$$

In Eq. (11), z^T, W_c represent vectors and matrices, respectively. W_y represents the learning parameter. c_t represents the training program at time t. y_i represents the i-th input item. The weight matrix also needs to be normalized in softmax to obtain S_i^t. The final short-term feature V_s needs to be output after iteration in Eq. (12).

$$\begin{cases} S_i^t = soft\max(T_i^t) \\ V_s^{'} = \sum S_i^t y_i \end{cases} \tag{12}$$

The key to achieving the final recommendation lies in the model-based CF recommendation algorithm. It filters similar preferences in the neighborhood and recommends them to the target user through rating interaction between entities [20]. A multi-layer perception can perform deep CF recommendation, which combines to obtain diverse features in Eq. (13).

$$\begin{cases} q_1 = \phi_1(U_{s,a,r}, V_{s,a,r}) = \begin{bmatrix} U_{s,a,r} \\ V_{s,a,r} \end{bmatrix} \\ \phi_2(q_1) = a_2(w_2^T q_1 + b_2) \\ \cdots \\ \phi_l(q_{l-1}) = a_l(w_l^T q_{l-1} + b_l) \\ \hat{y}_{uv} = \sigma(h^T \phi_l(q_{l-1})) \end{cases} \tag{13}$$

In Eq. (13), $U_{s,a,r}$ represents the combination of user dynamic preference U_s, attribute U_a, and static feature U_r. $V_{s,a,r}$ represents the union of corresponding characteristics of the project. w_x, b_x, a_x represent the weight matrix, bias vector, and ReLU activation function of the x-th layer perception, respectively. $\hat{y}_{uv}$ represents the probability of interaction between entities. The model treats y_{uv} as a label. When there is already an association between the user and the project, the value is 1. On the contrary, it is 0. The probability range of entity interaction after training is [0, 1]. Eq. (14) is the final training objective function.

$$p(y, y^- | \Theta_f) = \prod_{(u,v)\in y} \hat{y}_{uv} \prod_{(u,v)\in y^-} \hat{y}_{uv} (1 - \hat{y}_{uv}) \tag{14}$$

In Eq. (14), $p(y, y^- | \Theta_f)$ represents the objective function obtained through the probability function. Finally, a negative logarithmic likelihood loss function is obtained to minimize the optimization results in Eq. (15).

$$L = -\sum_{(u,v)\in y} \log \hat{y}_{uv} - \sum_{(u,v)\in y^-} \log(1 - \hat{y}_{uv}) \tag{15}$$

In Eq. (15), y^- represents negative instances, which are randomly generated by non-interacting items in the iteration and control the sampling probability of positive instances.

IV. Performance Testing of CF Optimization Algorithm Model Based on Semantic Relationships

To verify the comprehensive performance of the recommendation model, simulation experiments are conducted, including two parts. The first step is to verify the feasibility of the CCS-based random walk basic framework. Subsequently, further validation is conducted on deep CF model based on semantic relationship to understand its excellent recommendation performance.

A. Performance Verification of the Basic Framework of a CCS Based-Random Walk

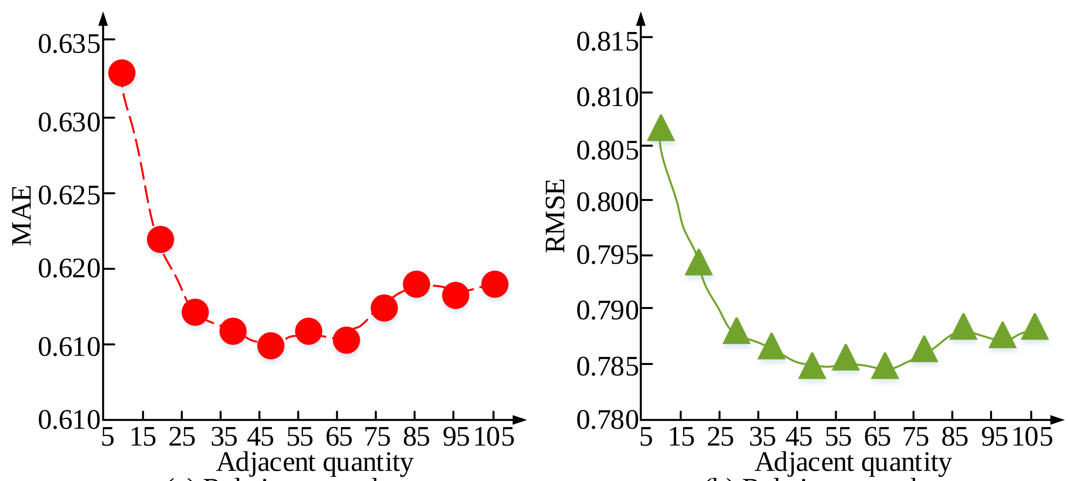
The study first validated the Optimized Random Walk (ORW) recommendation framework. Table II shows the experimental environment and parameters.

The experiment selected User-based CF (UCF), User-based CF by Genre Correlation (UBGC), Category Hierarchy Latent Factor Model (CHLF), and Genre to Classification model (GENC) for comparison with ORW. Firstly, the proximity number of UCF was analyzed using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) indicators in Fig. 6.

Fig. 6 shows the error impact of proximity number on UCF, respectively. In both MAE and RMSE, UCF tended to gradually decrease with the potential number, and there was a certain rebound phenomenon in the later stage. When the proximity number was 45, the MAE reached a minimum of 0.615. When the proximity number was 65, the RMSE reached a minimum of 0.785. Therefore, in subsequent experiments, the median value of 55 was chosen as proximity value. In further comparative experiments on various algorithms, precision, recall rate, F1 index, and Area Under Curve (AUC) index are selected for analysis. Conditions with 10 and 20 recommended lists belonging to the test set (i.e. different vector dimensions K) were selected for comparison, as shown in Table III.

TABLE II. EXPERIMENTAL ENVIRONMENT AND PARAMETER SETTINGS

Name		**Parameter/Settings**
Datasets		**MovieLens: 1M**
Data set total	User	6040
	Item	3952
	Review	1000209
Data sparsity rate		95.8%
Item type quantity		18
Number of combination types		301(Basic type: 6)
Verification method		5 fold cross verification
Training set: Test set		8:2



(a) Relation curve between proximity number and MAE

(b) Relation curve between proximity number and RMSE

Fig. 6. Relation curve between proximity number and MAE/RMSE in UCF model.

According to Table III, the number of recommended lists that belong to the test set affects the performance of each model. For precision, all models reached their optimal values at Top@25. The difference between ORW and CHLF was relatively small, only 2.6% lower than the latter. Compared to UCF, UBGG, and GENC, its precision has increased by 61.3%, 61.3%, and 77.3%, respectively. For the recall rate, all models reached their optimal value at Top@35. ORW performed the best, outperforming the other models by 57%, 59%, 6%, and 72.5%, respectively. F1 demonstrates a comprehensive performance of precision and recall. The performance difference between ORW and CHLF was minimal and almost negligible. Compared to UCF, UBGG, and GENC, ORW was more than 55% higher. AUC represents the probability that the classifier outputs positive samples higher than negative samples, which is a high value indicating good classification performance of the model. When it is not less than 0.5, it indicates that the classification effect is better than that of a random classifier. It also reached its optimal value at Top@35. The recommendation results of each model were superior to the random classifier, and the difference between the models was not more than 11%, indicating that these models were relatively excellent in recommendation performance. UCF, UBGG, and GENC perform relatively poorly, with UCF performing the worst because the latter two use project-based associations, while the former uses user-based associations. ORW and CHLF are better because they both organize project types and obtain richer structural information. Fig. 7 shows the MAE and RMSE of each model.

From Fig. 7, CHLF performed the best, with MAE and

RMSE of 0.2 and 0.29, respectively. ORW was second only to CHLF, but its error was still much greater than CHLF, with MAE and RMSE of 0.6 and 0.79, respectively, an increase of 66.6% and 63.3%. In the comparison between ORW and other models, its MAE and RMSE were 25.9%/25.3% lower than UCF, 24.1%/45.6% lower than UBGC, and 31.6%/24.0% lower than GENC. CHLF outperforms all other models due to the used hierarchical type structures to construct entity-based implicit semantic models. Although the proposed model has CCS, it only performs a relatively simple similarity calculation on type numbering, which further demonstrates the necessity of introducing deep CF recommendations in the future.

TABLE III. COMPARISON OF THE COMPREHENSIVE PERFORMANCE OF EACH RECOMMENDED ALGORITHM

Index		ORW	UCF	UBGG	CHLF	GENC
Precision	Top@15	0.183	0.069	0.063	0.182	0.030
	Top@25	0.150	0.058	0.058	0.154	0.034
	Top@35	0.130	0.051	0.055	0.135	0.035
Recall	Top@15	0.075	0.029	0.024	0.065	0.012
	Top@25	0.117	0.049	0.044	0.108	0.026
	Top@35	0.149	0.064	0.061	0.140	0.041
F1	Top@15	0.107	0.041	0.035	0.095	0.017
	Top@25	0.132	0.053	0.050	0.127	0.029
	Top@35	0.139	0.057	0.058	0.138	0.037
AUC	Top@15	0.799	0.654	0.652	0.784	0.615
	Top@25	0.810	0.707	0.713	0.805	0.693
	Top@35	0.820	0.733	0.745	0.818	0.750

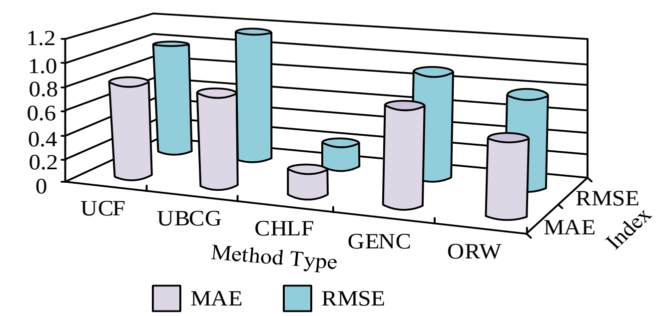


Fig. 7. Comparison of MAE/RMSE indicators of each model.

B. Performance Comparison Analysis of Semantic Relationship Temporal Recommendation Model Based on Deep CF

The study conducted performance validation analysis on the final model using a dataset. k in the triplet encoding Transe is 100. b_1, b_2 are 7 and -2, respectively. L_1 represents a regular term. The model operation framework and runtime platform are Keras and Python, respectively. To select the optimal Batch Size (S) and Learning Rate (R), this study used whether the test item was on the recommendation list and its position in the list as evaluation indicators, represented by HR and NDCG, respectively. The high value of both indicates excellent model performance. Table IV shows the experimental results.

TABLE IV. HR/NDCG CHANGES UNDER DIFFERENT R/S SETTINGS

Index		HR@10	NDCG@10
R(×10-3)	0.1	0.681	0.405
	0.5	0.690	0.438
	1.0	0.700	0.451
	1.5	0.697	0.449
S	128	0.657	0.415
	256	0.701	0.452
	384	0.689	0.436
	512	0.674	0.428

In Table IV, when R increased, both HR and NDCG showed an initial increase followed by a slow decrease, with turning points of 0.001. Compared to the initial R of 0.0001, HR and NDCG were increased by 2.7% and 10.2%, respectively. Afterwards, HR and NDCG showed a gradually decreasing trend, but the decrease was smaller compared to the increase. In the testing of S, the change was the same as that of R, with a turning point of 256. Compared to the initial 128 batches, HR and NDCG were increased by 6.3% and 8.2%, respectively. As the S and R continue to increase, the proportion of test items in the recommendation list of the model changes relatively weakly, but the change in the position of the item in the list is relatively significant. When R=0.001 and S=256, the proportion of items in the recommendation list is higher, and their positions in the list are higher. Therefore, these parameters are selected as model settings for the experiment. CoupledCF, a Wide & Deep model based on logistic regression and feedforward deep neural networks, and a multi-layer perception NCF are compared with the Semantic Relationship Timing Recommendation model based on deep CF (SRT-DCF). SRT-DCF includes three types: user, item, and NCF. Fig. 8 shows the recommended results for TOP@10.

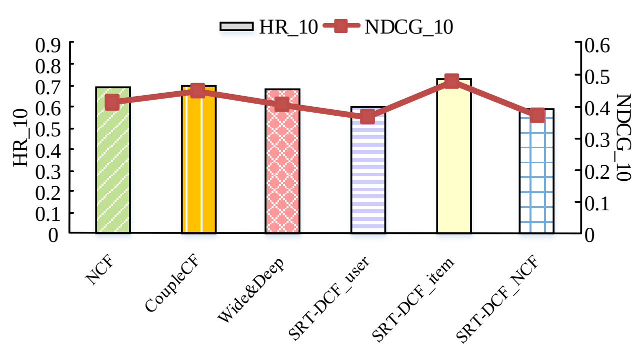
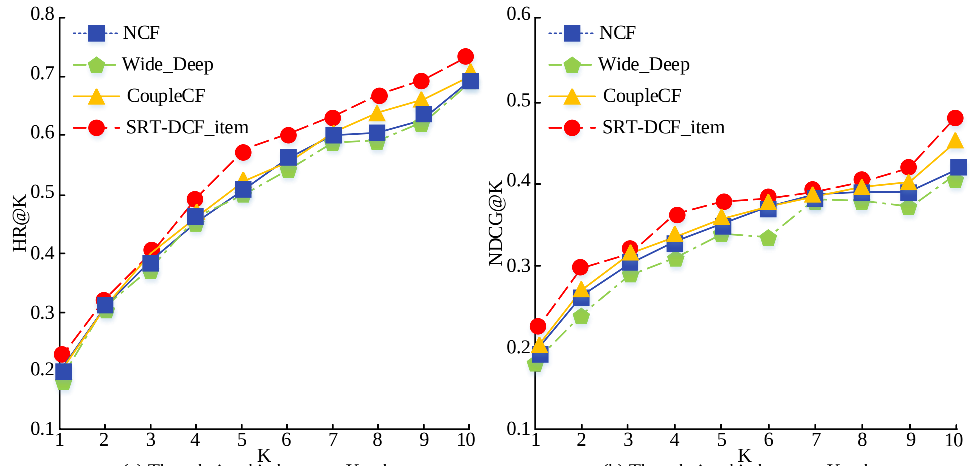


Fig. 8. Comparative analysis of HR_10/NDCG_10 performance of each model.

In Fig. 8, the difference among SRT-DCF_user, SRT-DCF_item, and SRT-DCF_NCF lies in the input of LSTM and the input of attention module. The inputs of SRT-DCF_user were the static and attribute features of recent projects learned from knowledge, as well as the onr-hot coding projects of the past hour. The inputs of SRT-DCF_item were onr-hot encoding item, as well as the static and attribute features of recent items learned from knowledge. The inputs of SRT-DCF_NCF were static and attribute features of recent projects learned from knowledge. The combined performance of CoupledCF and SRT-DCF_item was the best, while the performance of SRT-DCF_user and SRT-DCF_NCF was the worst. The average values of HR_10 and NDCG_10 were 0.583 and 0.3705, respectively. Compared to NCF and Wide&Deep models, the SRT-DCF_item had an average increase of 5.55% in HR_10 and 14.6% in NDCG_10. In summary, the research should select SRT-DCF_item as the final model. To further understand the impact of vector dimensions on various models, a comparative analysis was conducted in Fig. 9.



(a) The relationship between K value and HR@K of each model

(b) The relationship between K value and NDCG@K of each model

Fig. 9. Performance comparison of different models in different vector dimensions.

According to Fig. 9 (a), the HR@K change for each model was basically consistent, with the deviation node being K=5. At this time, the HR@K of the SRT-DCF_item model reached 0.583, which was 1.9%, 12.5%, and 1.8% higher than NCF, Wide&Deep, and CoupledCF models, respectively. When K was 10, the HR@K of other models was distributed around 0.67, while the HR@K of SRT-DCF_item reached 0.72, with an average increase of 6.9%. According to Fig. 9 (b), the NDCG@K variation was more tortuous. When K was 10, the values of each model were 0.493, 0.482, 0.409, and 0.397, respectively. Therefore, the average NDCG@K of SRT-DCF_item was 12.9% higher than that of other models. In summary, the proposed model has the best overall performance.

V. Results and Discussion

To enhance the personalized recommendation function for users, a CF model based on semantic relationships is proposed. Firstly, a CCS based on random walk basic recommendation framework was constructed, and the semantic relationships between entities were defined. Then, the final deep CF temporal recommendation model was further constructed using semantic relationships. The recommendation algorithm proposed in study [7] mainly focuses on extracting and analyzing user features, and lacks in mining complex semantic relationships between items and types. This study achieved a one-to-one mapping between projects and types by constructing the Hasse diagram and the lightweight KG, and delved into the potential semantic relationships between entities, surpassing the recommendation algorithm proposed in study [7] in terms of semantic understanding ability. The trust relationship recommendation model proposed in study [8] has some innovations in data preprocessing and trust relationship modeling, but it is slightly lacking in the depth and temporal considerations of the recommendation model. This study not only constructs a deep CF temporal model based on semantic relationships, but also fully considers the dynamic changes in user needs, thus outperforming the recommendation model proposed in study [8] in terms of recommendation accuracy and real-time performance.

The study conducted experimental verification according to the above two parts. In the experiments on ORW, compared to UCF, UBGG, and GENC, the precision of the proposed method was increased by 61.3%, 61.3%, and 77.3%, respectively, but 2.6% lower than CHLF. The recall rate was 57%, 59%, 6%, and 72.5% higher than other models, respectively, but the average MAE and RMSE were higher than CHLF by 66.6% and 64.85%, indicating the importance of subsequent optimization. In the final model validation, R and S were first analyzed. When R=0.001 and S=256, the HR@K and NDCG@K of this model reached the maximum mean of 0.7 and 0.45. The experiment selected vector dimension 10 and analyzed each model. Compared to NCF and Wide & Deep, the SRT-DCF_item had an average increase of 5.55% in HR_10 and 14.6% in NDCG_10. In the validation of the impact on vector dimensions, HR@K and NDCG@K were on average 6.9% and 12.9% higher than other models.

VI. Conclusion

A semantic-based CF recommendation system was proposed to address the personalized user preferences in interior design, and its recommendation performance was tested. In summary, the proposed CF recommendation algorithm based on semantic relationships has excellent performance. However, this study still remains at a static level in constructing KG, failing to fully capture the dynamic growth and evolution of information in actual situations. Therefore, in future research, real-time data stream processing technology should be further introduced to capture and process new information. Meanwhile, efficient dynamic update algorithms should be developed to update entities and relationships in the KG in real time.

REFERENCES

[1] Wu Q, Cheng X, Sun E, et al. Collaborative filtering algorithm based on optimized clustering and fusion of user attribute features//The International Society for Applied Computing (ISAC), Tokyo University of Science, Japan and Cisco Networking Academy. Proceedings of 2021 4th International Conference on Data Science and Information Technology (DSIT 2021). ACM, 2021, 1(1):136-140.

[2] Wang M, Li Q. A Multi-Agent Based Model for User Interest Mining on Sina Weibo. China Communications (English), 2022, 19(2):225-234.

[3] Li M, Li Y, Xu YC, et al. Explanatory Q&A recommendation algorithm in community question answering. Data Technologies and Applications, 2020, 54(4):437-459.

[4] Gou L, Zhou L, Xiao Y. Design and Implementation of Collaborative Filtering Recommendation Algorithm for Multi-layer Networks. Proceedings of the International Computer Frontiers Conference, 2021, 000(001): 32-50.

[5] Wu C, Liu S, Zeng Z, et al. Knowledge graph-based multi-context-aware recommendation algorithm. Information Sciences, 2022, 595(1):179-194.

[6] Yan HC, Wang ZR, Niu JY, et al. Application of covering rough granular computing model in collaborative filtering recommendation algorithm optimization. Advanced Engineering Informatics, 2022, 51(1):101485-101495.

[7] Miao R, Li B. A user-portraits-based recommendation algorithm for traditional short video industry and security management of user privacy in social networks. Technological Forecasting and Social Change, 2022, 185(3): 122103-122103.

[8] Liang X, Yin J. Recommendation Algorithm for Equilibrium of Teaching Resources in Physical Education Network Based on Trust Relationship. Journal of Internet Technology, 2022, 23(1): 133-141.

[9] Qi W, Yu J, Liang Q, et al. Design of Graph Neural Network Social Recommendation Algorithm Based on Coupling Influence. International journal of pattern recognition and artificial intelligence, 2022, 36(14): 122103-122111.

[10] Wang F. IoT for smart English education: AI-based personalised learning resource recommendation algorithm. International Journal of Computer Applications in Technology, 2023, 71(3): 200-207.

[11] Zhang L, Du Y. Resilience of space information network based on combination of complex networks and hypergraphs. Computer communications, 2022, 195(11): 124-136.

[12] Zheng L, Zhao T, Han H, et al. Personalized Tag Recommendation Based on Convolution Feature and Weighted Random Walk. International Journal of Computational Intelligence Systems, 2020, 13(1):24-35.

[13] Taheri M, Farnaghi M, Alimohammadi A., et al. Point-of-interest recommendation using extended random walk with restart on geographical-temporal hybrid tripartite graph. Spatial Science, 2021, 68(1):71-89.

[14] Yang Q, Wang H, Bian M, et al. Incorporating Reverse Search for Friend Recommendation with Random Walk. The international arab journal of information technology, 2020, 17(3): 291-298.

[15] Manju G, Abhinaya P, Hemalatha MR, et al. Cold Start Problem Alleviation in a Research Paper Recommendation System Using the Random Walk Approach on a Heterogeneous User-Paper Graph. International Journal of Intelligent Information Technologies (IJIIT), 2020, 16(2): 24-48.

[16] Liu Y, Ma H, Jiang Y, et al. Learning to Recommend via Random Walk with Profile of Loan and Lender in P2P Lending. Expert Systems with Applications, 2021, 174(1): 114763-114776.

[17] Masood F, Masood J, Zahir H, et al. Novel approach to evaluate classification algorithms and feature selection filter algorithms using medical data. Journal of Computational and Cognitive Engineering, 2023, 2(1): 57-67.

[18] Fang Y, Luo B, Zhao T, et al. ST-SIGMA: Spatio-temporal semantics and interaction graph aggregation for multi-agent perception and trajectory forecasting. CAAI Transactions on Intelligence Technology, 2022, 7(4):744-757.

[19] Li G, Liu H, Li G, et al. LSTM-based argument recommendation for non-API methods. Science in China: Information Science (English), 2020, 63(9): 8-30.

[20] Dib B, Kalloubi F, Nfaoui EH. A Boulaalam. Incorporating LDA with LSTM for followee recommendation on Twitter network. International Journal of Web Information Systems, 2021, 17(3): 250-260.