



ORIGINAL ARTICLE

Hierarchical User Intention–Preference for Sequential Recommendation with Relation-Aware Heterogeneous Information Network Embedding

Fan Yang,^{1,*} Gangmin Li,² and Yong Yue¹

Abstract

Existing recommender systems usually make recommendations by exploiting the binary relationship between users and items, and assume that users only have flat preferences for items. They ignore the users' intentions as an origin and driving force for users' performance. Cognitive science tells us that users' preference comes from an explicit intention. They first have an intention to possess a particular (type of) item(s) and then their preferences emerge when facing multiple available options. Most of the data used in recommender systems are composed of heterogeneous information contained in a complicated network's structure. Learning effective representations from these heterogeneous information networks (HINs) can help capture the user's intention and preferences, therefore, improving recommendation performance. We propose a hierarchical user's intention and preferences modeling for sequential recommendation based on relation-aware HIN embedding (HIP-RHINE). We first construct a multirelational semantic space of heterogeneous information networks to learn node embedding based on specific relations. We then model user's intention and preferences using hierarchical trees. Finally, we leverage the structured decision patterns to learn user's preferences and thereafter make recommendations. To demonstrate the effectiveness of our proposed model, we also report on the conducted experiments on three real data sets. The results demonstrated that our model achieves significant improvements in Recall and Mean Reciprocal Rank metrics compared with other baselines.

Keywords: recommender system; sequential recommendation; heterogeneous information networks; user intention modeling

Introduction

One of the critical tasks in the recommender system is to help users find the items that they are interested in from many items, and this will improve the user experience. Traditional recommendation algorithms usually use a binary relation between the users and items to learn users' preferences for recommendation, such as collaborative filtering^{1,2} recommending items to users based on users or items similarity, matrix factorization^{1,3} decomposing scoring matrix into latent feature expression of users and items, and then recommending items of interest to each user. However, they all have problems such as sparse matrix, cold start, flattening preference, and limiting the model's performance.

There is a natural interaction process in the actual user's purchase behavior⁴: first, the user intends to buy a specific type of item (e.g., a jacket), and then driven by this intention, he/she selects a particular item (jacket of a specific brand or a specific color) based on his/her preference and availability. This purchase behavior coincides with cognitive studies^{5,6} wherein preference only emerges one has an intention and that intention can be fulfilled with multiple options. The traditional recommendation algorithms use the user–item binary interaction relationship, ignoring the origin and the driving of the preference that is user's intention. This is because modeling user's intention and preference is challenging.

¹Department of Computer Science, School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China.

²Department of Computer Science, School of Computer Science and Technology, University of Bedfordshire, Luton, United Kingdom.

*Address correspondence to: Fan Yang, Department of Computer Science, School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou 215000, China, E-mail: yangfan@liverpool.ac.uk

The existing recommender systems contain a wealth of different types of information, which constitutes a heterogeneous information network (HIN).⁷ HINs generally have nodes and links in the form of nodes and links, which reflect different semantic perspectives on user preference.⁸ The model in Sun et al.⁹ uses matrix decomposition and factorization machine to learn the feature expression of users and items in different meta paths. It can only learn better for specific meta paths because the model has different learning abilities depending on the meta paths.

Chang et al.¹⁰ construct a HIN model, such as defining a network model on Yelp data set through node types user, review, word, etc. A proposal was then to define the semantic information association between two nodes located on two different meta paths using the PathSim algorithm. Prabhu et al.¹¹ propose a method to learn the feature representation of various types of nodes by deep heterogeneous network embedding. The model uses a convolutional neural network and fully connected layer to learn the embedding of images and text.

However, the mentioned methods have four shortcomings:

1. When modeling user preferences using a binary relationship between a user and an item, the assumption is that the user's preferences are flat, ignoring the hierarchical relationship between user intention and preference.
2. Identifying semantic heterogeneity between various types of nodes and relationships is difficult when modeling them in a shared feature space.
3. Fine-grained learning of node representation based on particular relationships does not quite exist.
4. Distinct link relationships may correlate with different features of node properties.

In this article, based on relation-aware HIN embedding, we propose hierarchical intention and preference modeling for sequential recommendation. We make the nodes that hold the relationship close to each other and the nodes that weakly hold or do not hold the relationship far away by projecting each relationship and corresponding node in the HIN into the relationship-specific semantic space rather than the public space. To integrate disparate information, we create a relation-aware attention layer that personalizes the influence of different connections on node representation learning.

We model hierarchical user intention and preference based on multirelational node embedding learned in a HIN. We adopt high-level user-category decision making to understand user's category intention and specific preferences within the intention. The model ranks and recommends items depending on their learned preference degree that is explainable.

Our contributions mainly include the following four aspects:

1. We apply relation-aware HIN embedding to generate distinct node embedding that has diverse relationships among user-item-category.
2. We propose a relation-aware attention mechanism to learn the varied effects of different relationships on the representation of distinct node features.
3. We construct a hierarchical tree of user intention and infer the possible user intentions and preferences.
4. We evaluate our method on three real-world data sets, and the results demonstrate that the proposed model outperforms the baseline methods.

This article is organized as follows: Related Work section reviews related studies that lead to our proposed model in Methodology section; Experiments section details experiments and discussions followed by conclusions.

Related Work

HIN Embedding-based recommendation

As opposed to homogeneous networks, HINs have multiple types of nodes and edges. Several attempts with HIN embedding have yielded promising results in various tasks.^{12–15} The recommender system based on HIN successfully solves the problem of how to model different kinds of heterogeneous auxiliary information and user interaction behavior. It effectively alleviates the problem of data sparsity and cold start in the recommendation system and can significantly improve the interpretability of the recommender system.

The fundamental of a recommender system based on a HIN is to model the user-item interaction and all auxiliary information into the HIN, and then design a recommendation model suitable for the HIN.¹⁶ Sem-Rec¹⁷ takes into account the attribute values of links, learns the weight mechanism of different meta paths, combines these similarities, and approximates the scoring matrix.

HeteRec¹⁸ uses a meta path to calculate the item-item similarity, then makes an inner product with a user scoring matrix to generate a user preference diffusion matrix, and uses a non-negative matrix on the diffusion matrix to learn potential characteristics of users and items. HIN2Vec¹³ learns HIN embeddings by performing several prediction training tasks concurrently. HERec¹⁵ filters node sequences with type restrictions, capturing the semantics of HINs.

Sequential recommendation

In contrast to traditional recommendation approaches such as collaborative filtering,^{19–21} or matrix factorization,^{3,22} sequential recommendation aims to capture the temporal shifting patterns of user preferences. The majority of classical approaches are based on Markov Chains (MCs), which explore how to extract sequential patterns to learn users' following preferences using probabilistic decision-tree models.^{23–26} Nevertheless, MC-based approaches can only represent local sequential patterns between neighboring interactions and cannot address the whole series. Then successive recommendation algorithms based on factorization machines are applied.

For instance, Rendle et al.²³ present FPMC, which combines matrix factorization and the Markov model to simulate individualized transition probability. Cheng et al.²⁷ expand FPMC to PFMC-LR and use a Markov model to provide geographical limits to the user's movement range. The enormous success of deep neural networks also has spurred the use of deep models in sequential recommendation.^{25,28,29} For example, Wang et al.³⁰ integrate auxiliary and identity information to develop e-commerce recommendations to prevent the recommender system's cold start. Wang et al.³¹ introduce HRM—hierarchical representation model, which can extract interest representations more effectively from user behavior sequences.

Recently, Recurrent Neural Networks have been devised to model variable-length sequential data with the goal of encoding previous user behaviors into latent representations. Hidasi et al.,³² particularly, use gated recurrent units to collect user behavior sequences for session-based recommendations, and they subsequently suggest an enhanced version³³ with a different loss function. Liu et al.³⁴ and others^{35,36} investigate the challenge of sequential recommendation given contextual information. Furthermore, unidirectional²⁸ and bi-directional²⁹ self-attention techniques are used to collect sequential patterns of user activities, resulting in state-of-the-art performance.

Nevertheless, these approaches only focus on modeling the relationships between the target user's prior behaviors and their upcoming behavior, leaving out the capacity to capture user intents buried in the behaviors. As a result, conventional techniques are unable to comprehend why the target user makes her following action.

Intention-aware recommendation

In recent years, diverse intention-aware recommendation has drawn great attention. It takes into account users' intents in behavior modeling. Zhu et al.³⁷ propose a key-array memory network (KA-MemNN) that portrays intents directly using items' categories in users' behaviors. This approach is straightforward and provides an obvious way to define user intents. Chen et al.³⁸ employ an attention mechanism to capture users' category-wise intentions, represented by a pair of action types and item categories. Wang et al.³⁹ propose a neural intention-driven method for modeling the heterogeneous intentions underlying users' complex behaviors.

Li et al.⁴⁰ present an intention-aware method to capture each user's underlying intentions that may lead to her following consumption behavior and improving recommendation performance. Wang et al.⁴¹ aggregate the history sequence into relation-specific embeddings to model dynamic impacts of historical relational interactions on user intention. In contrast, they give less attention to simulating user intentions, particularly when users' behaviors are melting. They also disregard organized user intent transition, resulting in a solid inductive bias for sequential recommendation.

Attention mechanism-based recommendation

Deep learning's attention process³¹ is comparable with humans' selective visual attention mechanism. Its purpose is to swiftly find more relevant information to the task goal among a significant volume of information. It is frequently used in text translation, sequence modeling, image recognition, video description, etc. Hidasi et al.³² pioneered the attention mechanism for machine translation within the encoder-decoder architecture. It can discover the shortest path between any two points, regardless of their distance or order. Deep Interest Network (DIN)⁴² model calculates the correlation between users' previous shopping histories and potential items using the attention mechanism.

In contrast, the DIN model does not take into account the time of user behavior and assumes that

user behavior is independent of each other. Deep Interest Evolution Network (DIEN)⁴³ holds that user interest is dynamic and shifts over time. A user interest extraction layer and a user interest evolution layer are presented based on DIN. Local activation is incorporated in each stage of Gated Recurrent Unit to boost the representation of relevant interests and mimic the movement of interests indicated by users in the behavior sequence.

Deep Session Interest Network (DSIN)⁴⁴ argues that the user behavior sequence has a hierarchical structure. User behavior in a single session is similar, and user behavior in subsequent sessions is considerably different. With high interpretability, the attention mechanism may distinguish the value of user behavior and screen out behaviors that are strongly related but irrelevant to objectives.

As we can see from above, the drawbacks associated with the traditional recommendation approach stimulate various of efforts in different directions. HIN Embedding-based recommendation tries to overcome problems with homogeneous networks; sequential recommendation aims to capture the temporal shifting patterns of user preferences. Realizing the root of the preference comes by the user's intention, many efforts have been conducted to capture the user's intention. Intention-aware recommendation simply tries to directly link user intents with behavior that ignores the behaviors conflict and intents transitions. Recent development on machine learning and deep learning shed new lights on the problem, attention mechanism-based recommendation is a brave attempt.

DIEN and DSIN are examples. However, neither explicitly represents users' intention and preference in a hierarchical structure. To address the issues identified, we propose a hierarchical user intention and preference framework for sequential recommendation based on relation-aware HIN embedding as described in the following section.

Methodology

In this section, we first introduce the problem formalization. Then we describe the proposed model framework in detail. After that, we talk about the different modules of our model. Finally, we discuss the model training.

Problem definition

Definition 1: Heterogeneous information network. A HIN is defined as a graph $G=(V, E, R, \phi, \varphi)$, in which V , E , and R are the sets of nodes, edges, and

edge types, respectively. V contains the set of users U , the set of items I , and the set of categories C .

Definition 2: Node and relation. We defined three types of nodes in HIN as follows: user nodes $u \in U$, item nodes $i \in I$, and category nodes $c \in C$. Besides, three types of relations are defined as follows: user-item ($u-i$), item-category ($i-c$), and user-user ($u-u$). A node relation triple, $u, r, i \in P$, describes that two nodes u and v are connected by a relation $r \in R$. Here, P represents the set of all node-relation triples.

Definition 3: HIN embedding. Given a HIN $G=(V, E, R, \phi, \varphi)$, HIN embedding aims to develop a mapping function $f: V \rightarrow \mathbb{R}^d$ that projects each node $v \in V$ to a low-dimensional vector in \mathbb{R}^d , where $d \ll |V|$.

Model framework

The framework of our approach is shown in Figure 1. It consists of three modules as follows:

1. **Relation-aware node embedding:** We generate distinct node embedding in HINs that have diverse relationships among the user-item-category. The user-item relationship represents the interaction between the user and item. Meanwhile, the item-category relationship represents which category the item belongs to. Relation-aware node embedding is to develop mapping functions that project nodes of diverse relationships to low-dimensional vectors.
2. **Relation-aware attention layer:** As the core of the attention model, the relational attention layer can capture the dependencies between nodes. To capture the effects of different relations on different node embeddings, we create the user-specific representation of categories as a sum of the node embeddings weighted.
3. **Hierarchical user intention and preference for sequential recommendation:** We construct a hierarchical tree of user intention and infer the possible user intentions and preferences the next time. We extract information about user intent from the relational attention layer and represent their hierarchical structure from fine to coarse. The users' intentions are learned to anticipate the interactions between users and items. We elaborate on the details of the three modules in the following subsections.

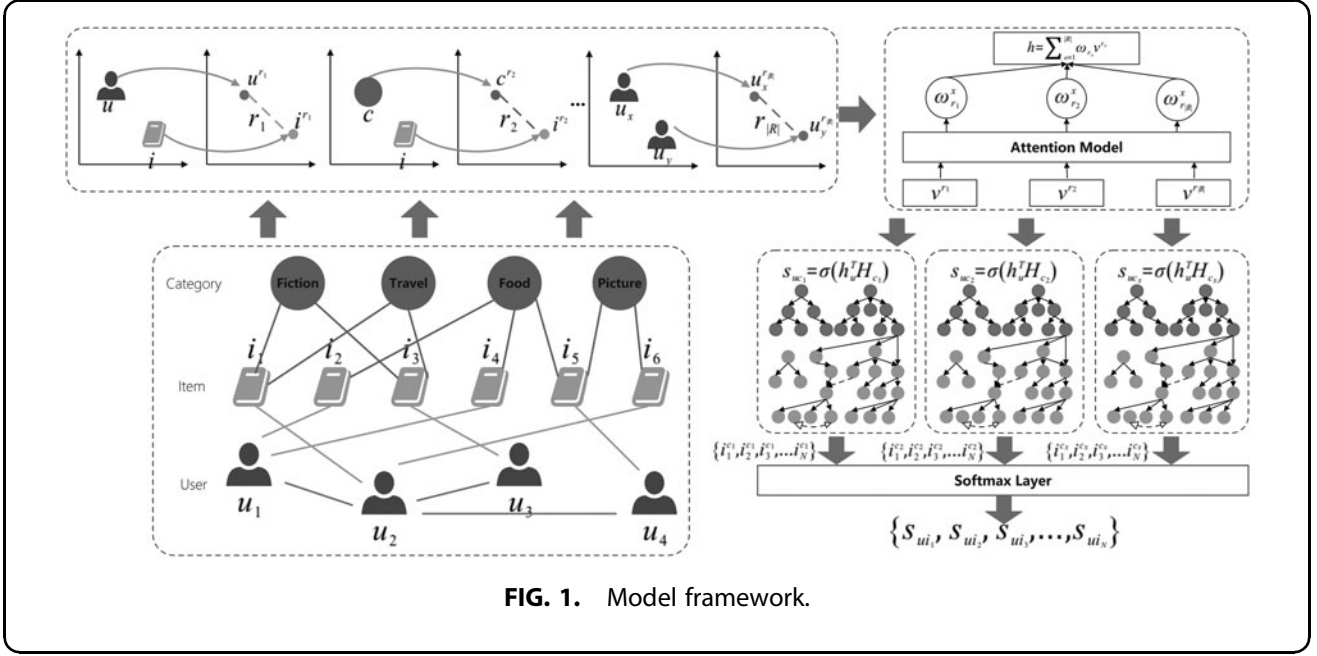


FIG. 1. Model framework.

Relation-aware node embedding

The observable node V is embedded through the embedding layer $W_v \in \mathbb{R}^{d \times |V|}$ to obtain low-dimensional embedding $v \in \mathbb{R}^d$. For observable triples $u, r, i \in P$, it represents that there is an edge r connecting between node u and node i , and edge r can also be called relation. We project it into the corresponding relation r semantic space. In the relation r semantic space, node u and node i are represented as $u^r = uM_r \in \mathbb{R}^{d_r}$, $i^r = iM_r \in \mathbb{R}^{d_r}$ after matrix $M_r \in \mathbb{R}^{d \times d_r}$ mapping, where d_r represents the embedding dimension in relation r semantic space.

The correlation of two nodes is measured by Euclidean distance. Euclidean distance satisfies the triangular inequality, naturally maintaining the first-order and second-order correlation. This specific relation projection can keep the related nodes closely connected with each other or keep the unconnected nodes away. The distance between node u and node i in relation r space is:

$$\text{dist}(u, i, r) = \|u^r + r - i^r\|_2^2, \quad (1)$$

where $r = rW_r \in \mathbb{R}^{d_r}$ represents the embedding vector of relation r , and $r \in \mathbb{R}^{|R|}$ is on-hot vector of relation r . $W_r \in \mathbb{R}^{d_r \times |R|}$ is the learnable parameter in the model. If $\text{dist}(u, i, r)$ is small, the relation r between node u and node i is strong. On the contrary, the relation r between node u and node i is weak or there is no relation r .

Relation-aware attention layer

Different relations have different semantic information. That is, they represent different aspects of nodes. This section wants to capture the effects of different relations on different node embeddings. We propose a relation-aware attention layer to learn to assign different attention weights to capture the relationships among the nodes. We input node embedding $v \in V$ into the attention layer, one layer can be formulated as follows:

$$h = \sum_r \omega_r v^r, \quad \omega_r = \frac{\exp(q^T \sigma(W_a v^r + b_a))}{\sum_k \exp(q^T \sigma(W_a v^k + b_a))}, \quad (2)$$

where ω_r represents the attention weight of relation r embedded in nodes $v \in V$, and $W_a \in \mathbb{R}^{d \times d_r}$, $q \in \mathbb{R}^d$, and $b_a \in \mathbb{R}^d$ are learnable parameters in the model. Finally, we get the final feature representation h of node v , which combines node embedding $v \in V$ based on multirelation semantics. Specifically, user type node u , item type node i , and category type node c correspond to h_u , h_i , and h_c , respectively.

Hierarchical user intention and preference for sequential recommendation

Inspired by Prabhu et al.¹¹ and Zhu et al.,⁴⁵ we build a hierarchical tree according to the characteristic that the category-item relation has a hierarchical index in the recommender system. The retrieval process of each

hierarchy is called hierarchical user intention and preference. To facilitate construction, at each hierarchy of nonleaf nodes in the tree, we first randomly sort the category information and place the items together that belong to the same category. If an item belongs to multiple categories, it will be randomly assigned to one of them.

Then we use the learned node embedding vector to recluster into a new tree. The nonleaf node is a coarse-grained category concept used as the index of items in the tree. The leaf node is the items in the corpus, which finely represents users' specific preferences under their intention. We predict the user's category intention and preference as follows:

$$s_{uc} = \sigma(h_u^T H_c), \quad (3)$$

where $H_c = [h_{c_x}]_{x=1}^{|C|} \in R^{d \times |C|}$ is category feature representation, and s_{uc} can also be written as $s_{uc} = [s_{uc_x}]_{x=1}^{|C|}$. Here, the value of s_{uc_x} reflects the user u 's preference for category c_x , and σ is *sigmoid* activation function.

We take the items $\{i_1, i_2, i_3, \dots, i_N\}$ that are the first K categories according to users' interest as the candidate set. The feature of this candidate set is represented as $H_i = [h_{i_x}]_{x=1}^N \in R^{d \times N}$, and then the user's preference for these candidate items is calculated based on user's category intention as follows:

$$s_{ui} = \text{softmax}(h_u^T H_i), \quad (4)$$

where $\text{softmax}()$ is a normalization function. $s_{ui} = [s_{ui_x}]_{x=1}^N$, where s_{ui_x} represents the probability that user u likes item i_x . We rank the probabilities in s_{ui} and recommend the top k items to user u .

Model training

We use Bayesian personalized ranking objective⁴⁶ to optimize our model. The key idea of Bayesian personalized ranking optimization is to make the items that users are really interested in ranking ahead of the items that users are not interested, that is, the positive sample probability is greater than the negative sample probability. So, we take a negative sample i_x for each positive sample $(u, i_x) \in D_{train}^+$. When optimizing node embedding in a relation semantic space, for the observable triple $u, r, i \in P \setminus D_{test}^+$, we take a negative sample i' , indicating no relation r between i' and u . We hope that u is closer to positive sample i and farther away from negative sample i' . Our optimization objectives are:

$$\begin{aligned} \mathcal{L} = \text{argmin} - & \sum_{(u, i_x) \in D_{train}^+, i_x'} \ln \sigma(s_{ui_x} - s_{ui_x'}) - \sum_{u, r, i \in P \setminus D_{test}^+, i'} \\ & \ln \sigma(\|uM_r + r - iM_r\|_2^2 - \|uM_r + r - i'M_r\|_2^2) \\ & + \lambda_1 \|W_*\|_2^2 + \lambda_2 q_2^2 + \lambda_3 \sum_{r \in R} \|M_r\|_2^2, \end{aligned} \quad (5)$$

where λ_1, λ_2 , and λ_3 are the regularization parameters. We use the Adam method⁴⁷ to optimize our model. $W_* = \{W_v, W_r, W_a\}$ and q are learnable parameters in our model.

Experiments

We provide empirical results to demonstrate the effectiveness of our proposed model. The experiments are designed to answer the following research questions:

- RQ1: How does our proposed model perform compared with other state-of-the-art sequential recommendation models and user intention modeling-based methods?
- RQ2: How does each module (i.e., multirelation HIN embedding, relation-aware attention layer, and hierarchical user intention) affect the performance of our model?
- RQ3: How do the influences of different parameters affect our proposed model?

Experiments settings

To answer the first research question (RQ1), we use three actual and available data sets and make comparisons with existing models on Recall and Mean Reciprocal Rank (MRR).

Data sets. To evaluate our proposed model, we conducted extensive experiments on the three real data sets. The statistics of the data sets are summarized in Table 1.

MovieLens. This data set is about movie ratings and has been widely used to evaluate recommendation algorithms. We use MovieLens-1m containing 1 million rating records, respectively. We extract interaction records from rating data, items from "movie name," and users from "user id."

Douban-Book. This data set is about book ratings collected from Douban website. We use friend relationship, rating data, and genres of books in the data set as category. It is worth noting that although our

Table 1. Statistics of the data sets

Data set	No. of users	No. of items	No. of interactions (m)	No. of genres	Average genres/item	Density (%)
MovieLens	6040	3416	1.0	18	144.4	4.79
Douban-Book	129,490	58,541	16.8	381	3.1	0.63
Last-FM	23,566	48,123	3.0	1946	8.7	0.08

model only illustrates three types of nodes, our model can be extended to more types of nodes and correspond more types of relationships.

Last-FM. This data set is about music that users listen to on the online music website Last.fm. The data set includes friend relationship, user listening to artist, user label to artist, and artist label. To unify category nodes, we take the artist's label as category.

Evaluation metrics. To evaluate the recommendation performance of our proposed model, we use two evaluation metrics Recall@K and MRR@K for short. The first metric evaluates the fraction of ground truth items that are retrieved over the total amount of ground truth items, whereas the second metric is the mean of reciprocal of the rank at which the ground-truth item is retrieved. The larger the values of both Recall and MRR metrics, the better the performance.

$$\text{Recall@K} = \frac{1}{|D_{\text{test}}^+|} \sum_{(u, i_x, r) \in D_{\text{test}}^+} \mathbf{I}(p_{u,x} \leq k), \quad (6)$$

$$\text{MRR@K} = \frac{1}{|D_{\text{test}}^+|} \sum_{(u, i_x, r) \in D_{\text{test}}^+} \frac{1}{R_{u,x}} \mathbf{I}(p_{u,x} \leq k), \quad (7)$$

where $p_{u,x}$ represents the ranking of the positive sample i_x among the top k items recommendation for user u . $\mathbf{I}(\cdot)$ indicates that if the positive sample i_x is in the top k items, it returns 1; otherwise it returns 0. D_{test}^+ is the test set.

Baselines. We compare our model with the following baseline algorithms, including HIN embedding methods, session-based recommendation, and hierarchical representation approaches.

Deep heterogeneous autoencoders. This article proposes a deep heterogeneous self-encoder to model heterogeneous auxiliary information to solve the data sparsity problem of the collaborative filtering algorithm.⁴⁸ We set the number of hidden layers of deep heterogeneous autoencoders (DHA) self-encoder $L=4$. We also sort the input data of DHA according to the data format re-

quirements in this article. The input data include user, item, category, and interaction.

BPR-MF + TransE. This method combines BPR-MF and TransE. BPR-MF combines Bayesian personalized ranking with matrix factorization model and learns personalized ranking from implicit feedback.⁴⁹ TransE models the node embedding of HIN. Because we do not use image data, we remove the image (visual knowledge) processing module in BPR-MF + TransE.

FPMC. This method models user preferences by combining MF, which captures users' general preferences and a first-order MC to predict the user's next action.²³

PageRank with Priors. This method integrates the user-item relationship and other heterogeneous auxiliary information into a unified isomorphism diagram.⁵⁰ PageRank outputs a personalized initial probability distribution. Similarly, we remove the image (visual knowledge) processing module in PageRank with Priors (PRP).

FOSSIL. This method integrates factored item similarity with MC to model a user's long- and short-term preferences.²⁶ We set μ_u and μ as single scalar since the length of each session is variable.

Hierarchical representation model. This method generates a hierarchical user representation to capture sequential information and general tastes.³¹ We use max pooling as the aggregation operation because this achieves the best result.

SHAN. This model employs two attention networks to mine users' long- and short-term preferences.⁵¹

Key-array memory network. This article proposes a KA-MemNN to hierarchize user intention preference for sequence recommendation based on the ternary relationship of user-intention-item.³⁷

Parameter settings. To facilitate the experiments, we filter out users and items for which interactive data

are <5 . For each user, we randomly select 80% of the interactive data as the training set D_{train}^+ and the remaining 20% of the interactive data as the test set D_{test}^+ . In the training set, we randomly selected 20% of the interactive data as the development set D_{valid}^+ to adjust our model parameters and comparison methods. In addition, models are tuned for best performance through tuning of parameters, such as learning rate $\alpha \in \{0.1, 0.01, 0.001, 0.0001\}$, regularization parameters $\lambda_* \in \{0.1, 0.01, 0.001, 0.0001\}$, $Dropout \in \{0.2, 0.4, 0.5\}$, and dimensionality $d = d_r \in \{60, 80, 100, 120, 140, 150\}$.

Performance comparison

We begin with the comparison with respect to Recall@20, Recall@50, MRR@20, and MRR@50. Table 2 gives the empirical results, with percent Imp. denoting the relative improvements of the top performing technique (bold) over the strongest baselines (underlined). We find the following:

1. Our model consistently outperforms all baselines across the three data sets in terms of all measures. More specifically, it achieves significant improvements over the strongest baselines with respect to MRR@20 by 7.25%, 25.7%, and 15.87% in MovieLens, Douban-Book, and Last-FM, respectively. Our model's logic and efficacy are demonstrated in this way. These gains can be attributed to our model's relational modeling: (1) By investigating user intentions, we can better define the links between users and objects, resulting in more effective user and item representations. Some baselines, in contrast, ignore hidden user intents; (2) our model learns node embeddings in HINs

based on user-intention-item relationships; (3) our model fuses node feature representations in multirelational semantic spaces using relation-aware attentional layers.

2. We can see that the sequential methods (e.g., FPMC, HRM, and KA-MemNN) outperform the nonsequential methods (e.g., BPR-MF, PRP, and FOSSIL) in general. The methods that only consider user actions without the sequential order do not make full use of the sequence information and report the worse performance. Specifically, compared with BPR-MF, the main advantage of FPMC comes from modeling historical user actions with first-order Markov chains, namely considering the sequence order, so that FPMC reports better results than BPR-MF. This can verify that sequential pattern is essential for improving the predictive ability for sequential recommendations.
3. BPRMF + TransE and PRP outperform DHA, indicating that HIN embedding can more reasonably capture heterogeneous information semantic features to improve recommendation quality rather than directly encoding structural information in a feature engineering manner. KA-MemNN outperforms both BPR-MF + TransE and FPMC on all the data sets, indicating that hierarchical user intent and preference are better than flat user preference of the learning approach. Compared with BPRMF + TransE and KA-MemNN, we model the heterogeneity of relationships in HINs based on specific relation semantics and personalize the fusion of node feature representations in each semantic space, and in addition, we model hierarchical user intentions and preferences according to the natural user interaction process.

Table 2. Overall performance comparison

Data set	Metric	DHA	BPR-MF + TransE	FPMC	PRP	FOSSIL	HRM	SHAN	KA-MemNN	Ours	%Imp
MovieLens	Recall@20	0.2712	0.3360	0.3389	0.3207	0.3159	0.3486	0.3421	0.3537	0.3715	5.03
	Recall@50	0.3423	0.4151	0.4168	0.4321	0.4098	0.4250	0.4278	<u>0.4324</u>	0.4578	5.87
	MRR@20	0.0628	0.2006	0.3021	0.1359	0.2891	0.3467	0.3097	<u>0.3628</u>	0.3891	7.25
	MRR@50	0.0677	0.2071	0.3110	0.1411	0.2928	0.3540	0.3203	<u>0.3702</u>	0.3956	6.86
Douban-Book	Recall@20	0.1298	0.1332	0.1487	0.1362	0.1442	0.1503	0.1220	0.1498	0.1687	12.24
	Recall@50	0.1726	0.1815	0.1930	0.1853	0.1902	<u>0.1977</u>	0.1703	<u>0.2011</u>	0.2203	9.55
	MRR@20	0.0352	0.0806	0.0851	0.0798	0.0814	0.0946	0.1033	<u>0.1222</u>	0.1536	25.70
	MRR@50	0.0407	0.0911	0.0934	0.0862	0.0892	0.1035	0.1092	<u>0.1301</u>	0.1610	23.75
Last-FM	Recall@20	0.0733	0.0812	0.0880	0.0863	0.0774	0.0902	0.0821	0.0743	0.1005	11.42
	Recall@50	0.1201	0.1334	0.1405	0.1391	0.1206	<u>0.1439</u>	0.1317	0.1258	0.1542	7.16
	MRR@20	0.0301	0.0330	0.0329	0.0305	0.0310	<u>0.0334</u>	0.0317	0.0309	0.0387	15.87
	MRR@50	0.0324	0.0357	0.0351	0.0343	0.0337	<u>0.0365</u>	0.0350	0.0349	0.0425	16.44

The top performing technique (bold), the strongest baselines (underlined).

%Imp, denoting the relative improvements of the top performing technique over the strongest baselines; DHA, deep heterogeneous autoencoders; HRM, hierarchical representation model; KA-MemNN, key-array memory network; PRP, PageRank with Priors.

4. As the data show, there is a discrepancy in performance between HRM and KA-MemNN. The disparity, we believe, is caused by the various degrees of user intentions. When compared with single-level user intentions, two-level intents may be thought of as an extension that separates user intents into particular and broad categories.

Impact of components

In this section, we drill deeper to answer question RQ2 the impact of each component in our proposed model, which is in relation with the overall performance based on embedding the public feature space. We also want to verify that our hierarchical user intent and preferences outperform flat user preferences on recommendation. We adopt three simplified versions of HIP-RHINE as follows.

HIP-RHINE-1: Remove the relation-aware heterogeneous information embedding module, and the replacement operation is to integrate heterogeneous relations and structured data into a unified isomorphic graph.

HIP-RHINE-2: Remove the relation-aware attention layer module, and the replacement operation is to directly add the feature expressions of nodes in each relation semantic embedding space point by point.

HIP-RHINE-3: Remove the hierarchical tree module, and the replacement operation is to directly calculate s_{ux} for the whole set of items and recommend by ranking.

We also apply Recall@N and MRR@N to evaluate the performance of these models. We show the results under the metrics of Recall@20, Recall@50, MRR@20, and MRR@50. In addition, we evaluate the score of each category as an average of the scores of its items. This way the intention-based MRR can also reflect the performance of item recommendations.

The results in Table 3 show that our method performs well on all the data sets compared with HIP-RHINE-1 because we consider the heterogeneity of relations for node embedding. Besides, our method performs well on all the data sets compared with HIP-RHINE-2 because our method captures the degree of influence of different relations on the final node embedding. The experiments demonstrate the effectiveness of our multirelational semantic embedding and relation-aware attention layer. Compared with HIP-RHINE-3, our method performs well on all data sets because our method hierarchizes user intents and predicts user preferences for items based on specific intents. The experiments show the effectiveness of hierarchical user intents and preferences.

Parameter analysis

After analyses on individual components in relation to the model's performance, we realize that the model's performance is also affected by the model's parameters.

To further investigate the influences of different parameters in our model, we calculate the values of Recall@20, Recall@50, MRR@20, and MRR@20 for HIP-RHINE across different numbers of dimensions with size d , and also explore the sensitivity of the parameter—the number of negative samples.

As shown in Figure 2a–d, the model's performance gradually improves as dimension d increases. However, the model performance decreases a little on the Last-FM data set when $d > 120$ and finally stabilizes. This trend indicates that the model can capture more complex feature embeddings as d increases. However, overincreasing d may lead to overfitting problems resulting in a degradation of model performance.

Furthermore, we study the effect of the sampling number k on the overall performance. Because the item sizes differ across the three data sets, we experiment with various k ranges. Specifically, we try $k \in \{10, 20, 30, 40, 50, 60\}$ on MovieLens, $k \in \{10, 100, 200, 300, 400,$

Table 3. Performance evaluation of variant models

Model	MovieLens				Douban-Book				Last-FM			
	R@20	R@50	M@20	M@50	R@20	R@50	M@20	M@50	R@20	R@50	M@20	M@50
HIP-RHINE-1	0.2917	0.3702	0.3058	0.3105	0.1088	0.1596	0.0862	0.0958	0.0802	0.1319	0.0191	0.0228
HIP-RHINE-2	0.3306	0.4099	0.3522	0.3567	0.1246	0.1802	0.1050	0.1133	0.0831	0.1344	0.0243	0.0310
HIP-RHINE-3	0.3158	0.3890	0.2714	0.2790	0.1412	0.1907	0.1164	0.1221	0.0932	0.1468	0.0284	0.0336
HIP-RHINE	0.3715	0.4578	0.3891	0.3956	0.1687	0.2203	0.1536	0.1610	0.1005	0.1542	0.0387	0.0425

The complete HIP-RHINE is the top performing (bold).

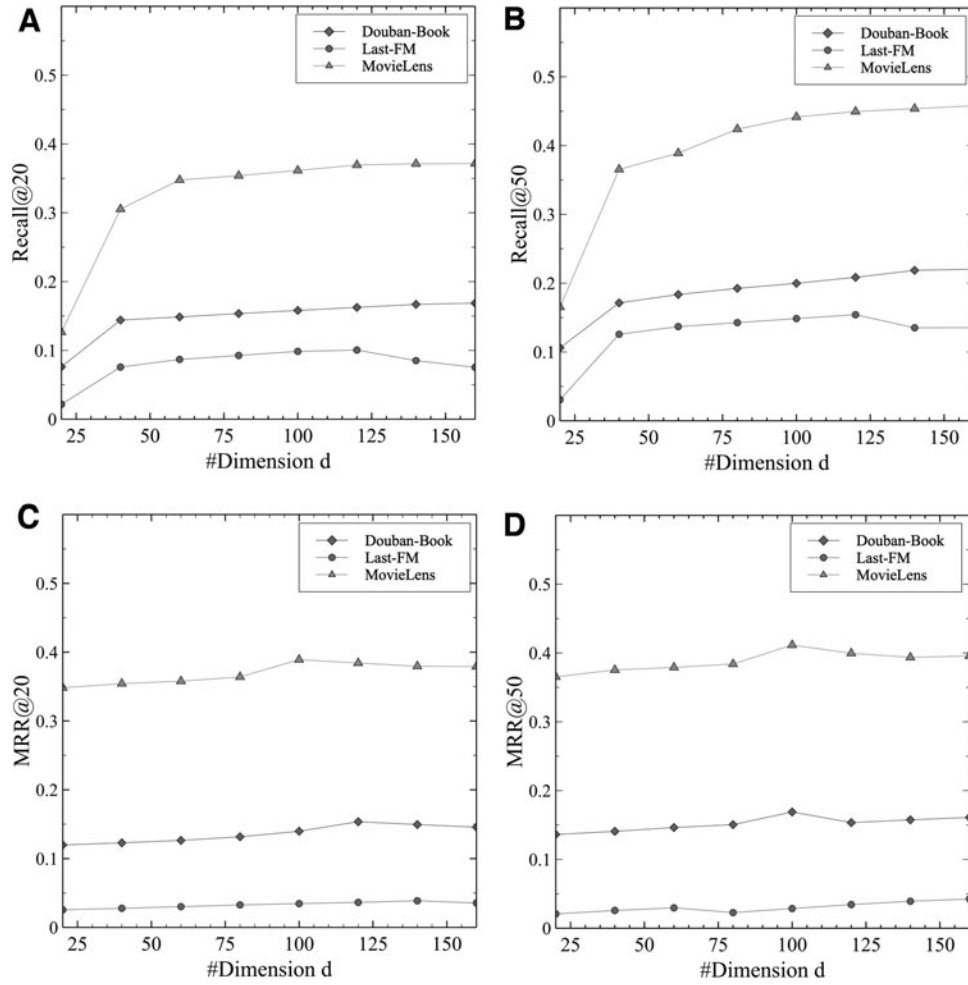


FIG. 2. (a–d) Performance in terms of Recall@20, Recall@50, MRR@20, and MRR@50 with respect to #Dimension d over three data sets. MRR, Mean Reciprocal Rank.

500} on Douban-Book, and $k \in \{10, 50, 100, 150, 200, 250\}$ on Last-FM, respectively. Here we only show the results on one dimension over each data set due to space limitation. As shown in Figure 3a–c, as the number of negative examples increases, the performance of our model first also grows. The trending is quite similar across all three data sets.

The performance gain between two successive trials, in contrast, diminishes as the sampling number k grows. It suggests that if we continue to sample more negative samples, we will see less performance progress but more computational complexity.

Case study

To investigate whether our proposed model is effective and explainable, we chose one user at random from

Douban-Book and visualize the hierarchical tree of user intention and preference.^{37,52} We extract attention between a single category and the observed objects that correspond to that category for each user.

As shown in Figure 4, there are three types of nodes. A category node is a broad term that encompasses a wide range of concepts. A concept is a collection of items that share some common attributes. Concepts, as opposed to coarse-grained categories and fine-grained entities, can assist in better representing users' interests at a semantic granularity that is appropriate. An entity is a unique item that belongs to one or more concepts. There are three sorts of edges between nodes as well. The IsA relationship denotes that the destination node is a child of the source node. The involved relationship indicates that the destination node is involved in a source node-described item.

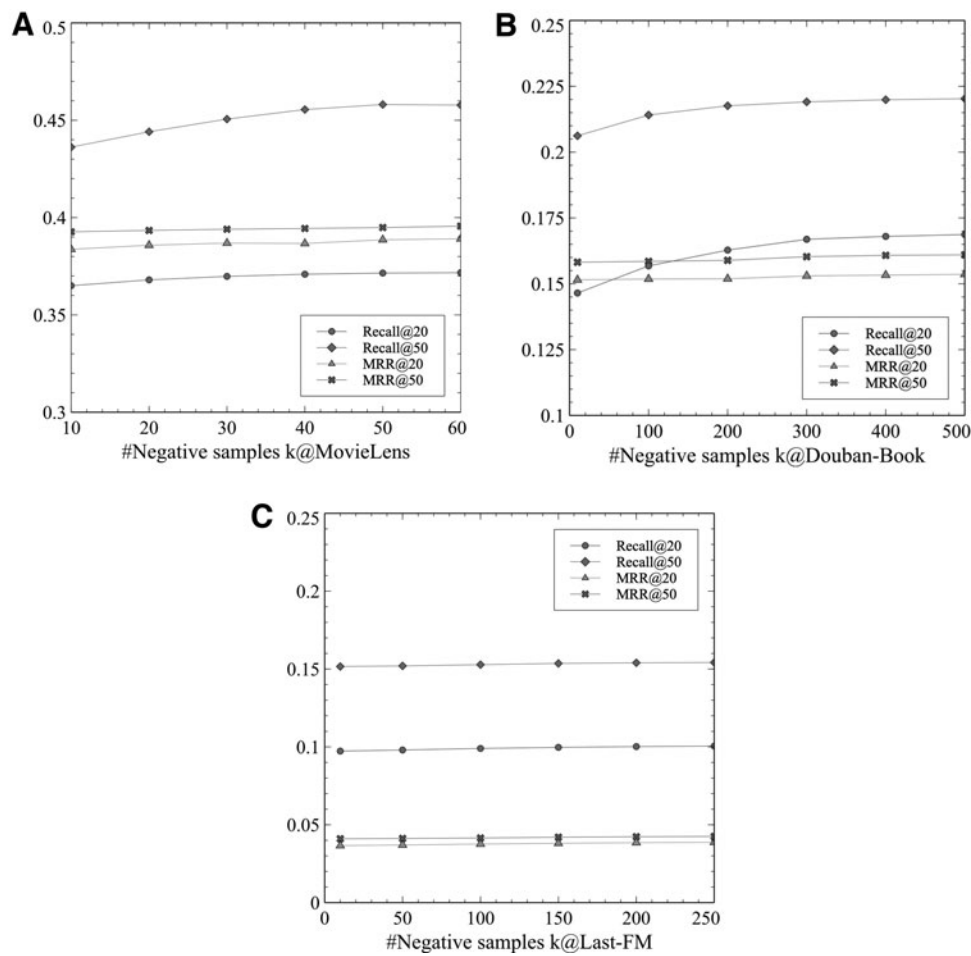


FIG. 3. (a-c) Performance in terms of Recall@20, Recall@50, MRR@20, and MRR@50 with respect to #Negative samples k over three data sets. The number of negative samples is increased from 10 to 60 on MovieLens, 10 to 500 on Douban-Book, and 10 to 250 on Last-FM.

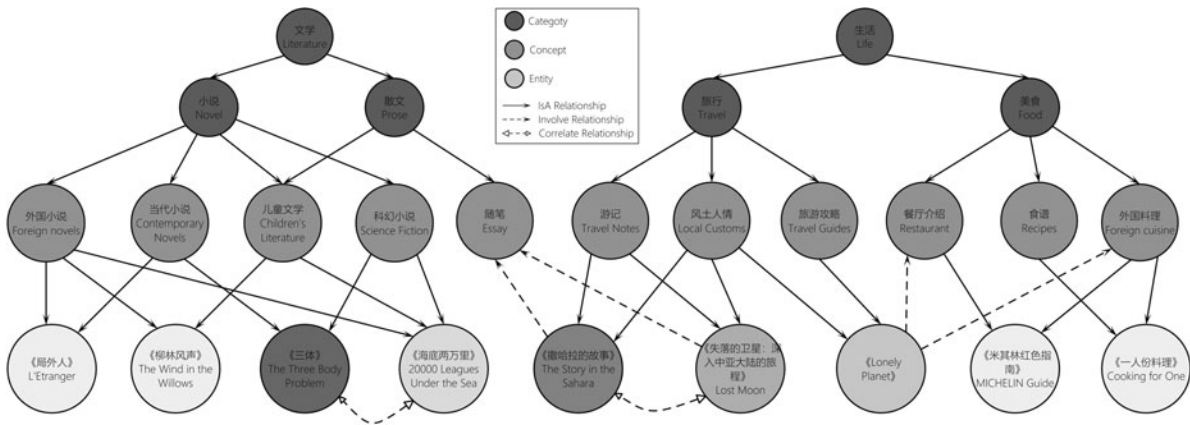


FIG. 4. An example to show the hierarchical tree of user intention and preference given by our approach.

The color scale of entity nodes (items) shows the value of the attention weights, with darker signifying a more considerable weight and lighter representing a lower weight, as illustrated in Figure 4. When generating category embeddings, we can see that the frequently visited objects are generally given a higher weight. This phenomenon might be explained because category-specific users' preferences are reflected in the most frequently viewed items in that category.

Conclusions

In this article, we propose a model for sequential recommendation based on hierarchical intentions and preferences with relation-aware HIN embedding, which can learn node representation in the HIN at a fine-grained level based on the particular relationships. To customize the merging of heterogeneous information, we adopt a relation-aware attention layer. Furthermore, we employ hierarchical trees to represent user intents and preferences hierarchically, and we use structured choice patterns of users for user preference learning to improve recommendation performance.

Extensive experiments on three real data sets are carried out to evaluate the performance of our proposed approach. In terms of Recall and MRR metrics, the findings show that our model outperforms state-of-the-art approaches by a significant margin. In the future, we will investigate multiple and variable intents or knowledge graph information combined with user intention modeling.

Authors' Contributions

F.Y. contributed to methodology (lead), writing—original draft (lead), formal analysis (lead), and writing—review and editing (equal). G.L. was involved in evaluation (lead) and writing—review and editing (equal). Y.Y. carried out conceptualization (lead) and writing—review and editing (equal).

Author Disclosure Statement

No competing financial interests exist.

Funding Information

This study was supported by Basic Public Welfare Research Project of Zhejiang, China (LGF20G020001), Key Lab of Film and TV Media Technology of Zhejiang Province (No. 2020E10015), and the AI University Research Centre (AI-URC) through the XJTLU Key Program Special Fund (KSF-A-17).

References

1. Hu YF, Koren Y, Volinsky C. Collaborative filtering for implicit feedback datasets. In: 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy: IEEE, December 15–19, 2008. pp. 263–272.
2. Linden G, Smith B, York J. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Intern Comput.* 2003;7:76–80.
3. Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. *Computer.* 2009;42:30–37.
4. Li Z, Zhao HK, Liu Q, et al. Learning from history and present: Next-item recommendation via discriminatively exploiting user behaviors. In: *Proceedings of the SIGKDD International Conference on Knowledge Discovery and Data Mining*, London, UK: ACM, July 19, 2018. pp. 1734–1743.
5. Li G, Hopgood AA, Weller MJ. Shifting matrix management: A model for multi-agent cooperation. *Eng Appl Artif Intell.* 2003;16:191–201.
6. Sugimoto T. A preference-based theory of intention. In: *Pacific Rim International Conference on Artificial Intelligence*. Berlin, Heidelberg: Springer, 2000. pp. 308–317.
7. Dias MB, Locher D, Li MQ, et al. The value of personalized recommender systems to e-business: A case study. In: *Proceedings of the 2008 ACM Conference on Recommender Systems*, Lausanne, Switzerland: ACM, October 23–25, 2008. pp. 291–294.
8. Zhao H, Yao QM, Li JD, et al. Meta-graph based recommendation fusion over heterogeneous information networks. In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Halifax, NS, Canada: ACM, August 13–17, 2017. pp. 635–644.
9. Sun YZ, Han JW, Yan XF, et al. PathSim: Meta-path-based top-k similarity search in heterogeneous information networks. *Proc VLDB Endowment.* 2011;4:992–1003.
10. Chang SY, Han W, Tang JJ, et al. Heterogeneous network embedding via deep architectures. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia: ACM, August 10–13, 2015. pp. 119–128.
11. Prabhu Y, Varma M. Fastxml: A fast, accurate and stable tree-classifier for eXtreme multi-label learning. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, New York, NY: ACM, August 24–27, 2014. pp. 263–272.
12. Tang J, Qu M, Mei Q. Pte: Predictive text embedding through large-scale heterogeneous text networks. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney, NSW, Australia: ACM, August 10–13, 2015. pp. 1165–1174.
13. Fu TY, Lee WC, Lei Z. Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. In: *Proceedings of the 2017 ACM Conference on Information and Knowledge Management*, Singapore: ACM, November 6–10, 2017. pp. 1797–1806.
14. Wang H, Zhang F, Hou M, et al. Shine: Signed heterogeneous information network embedding for sentiment link prediction. In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, Marina Del Rey, CA: ACM, February 5–9, 2018. pp. 592–600.
15. Shi C, Hu B, Zhao X, et al. Heterogeneous information network embedding for recommendation. *IEEE Trans Knowl Data Eng.* 2019;31:357–370.
16. Liu J, Shi C, Yang C, et al. Heterogeneous information network based recommender systems: A survey. *J Inform Sec.* 2021;6:16.
17. Shi C, Zhang ZQ, Luo P, et al. Semantic path based personalized recommendation on weighted heterogeneous information networks. In: *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*, Melbourne, Australia: ACM, October 18–23, 2015. pp. 453–462.
18. Yu X, Ren X, Sun YZ, et al. Recommendation in heterogeneous information networks with implicit user feedback. In: *Proceedings of the 7th ACM Conference on Recommender Systems*, Hong Kong, China: ACM, October 12–16, 2013. pp. 347–350.
19. Resnick P, Iacovou N, Suchak M, et al. Grouplens: An open architecture for collaborative filtering of netnews. In: *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, Chapel Hill, NC: ACM, October 22–26, 1994. pp. 175–186.
20. Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms. In: *Proceedings of the 10th International*

- Conference on World Wide Web, Hong Kong: ACM, May 1–5, 2001. pp. 285–295.
21. Cai Y, Leung H, Li Q, et al. Typicality-based collaborative filtering recommendation. *IEEE Trans Knowl Data Eng.* 2013;26:766–779.
 22. Baltrunas L, Ludwig B, Ricci F. Matrix factorization techniques for context aware recommendation. In: *Proceedings of the Fifth ACM Conference on Recommender Systems*, Chicago, IL: ACM, October 23–27, 2011. pp. 301–304.
 23. Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized Markov chains for next-basket recommendation. In: *Proceedings of the 19th International Conference on World Wide Web*, Raleigh, NC: ACM, April 26–30, 2010. pp. 811–820.
 24. He R, Fang C, Wang Z, et al. Vista: A visually, socially, and temporally-aware model for artistic recommendation. In: *Proceedings of the 10th ACM Conference on Recommender Systems*, Boston, MA: ACM, September 15–19, 2016. pp. 309–316.
 25. Tang J, Wang K. Personalized top-n sequential recommendation via convolutional sequence embedding. In: *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, Marina Del Rey, CA: ACM, February 5–9, 2018. pp. 565–573.
 26. He R, McAuley J. Fusing similarity models with Markov chains for sparse sequential recommendation. In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. Barcelona, Spain: IEEE, 2016. pp. 191–200.
 27. Cheng C, Yang H, Lyu MR, et al. Where you like to go next: Successive point-of-interest recommendation. In: Rossi F (Ed.): *Proceedings of the 23th International Joint Conference on Artificial Intelligence*, Beijing, China: AAAI Press, 2013.
 28. Kang W-C, McAuley J. Self-attentive sequential recommendation. In: *2018 IEEE International Conference on Data Mining (ICDM)*. Singapore: IEEE, 2018.
 29. Sun F, Liu J, Wu J, et al. Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, Beijing, China: ACM, November 3–7, 2019. pp. 1441–1450.
 30. Wang S, Hu L, Cao L. Perceiving the next choice with comprehensive transaction embeddings for online recommendation. In: Ceci M, Hollmén J, Todorovski L, et al. (Eds.): *Joint European Conference on Machine Learning and Knowledge Discovery in Database*. Cham: Springer, 2017. pp. 285–302.
 31. Wang P, Guo J, Lan Y, et al. Learning hierarchical representation model for next basket recommendation. In: *Proceedings of the 38th International ACM SIGIR conference on Research and Development in Information Retrieval*, Santiago, Chile: ACM, August 9–13, 2015. pp. 403–412.
 32. Hidasi B, Karatzoglou A, Baltrunas L, et al. Session-based recommendations with recurrent neural networks. *arXiv:1511.06939*, 2015.
 33. Hidasi B, Karatzoglou A. Recurrent neural networks with top-k gains for session-based recommendations. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. Lingotto, Turin, Italy: ACM, 2018. pp. 843–852.
 34. Liu Q, Wu S, Wang D, et al. Context-aware sequential recommendation. In: *2016 IEEE 16th International Conference on Data Mining (ICDM)*. Barcelona, Spain: IEEE, 2016. pp. 1053–1058.
 35. Yu F, Liu Q, Wu S, et al. A dynamic recurrent model for next basket recommendation. In: *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, Pisa, Italy: ACM, July 17–21, 2016. pp. 729–732.
 36. Li J, Ren P, Chen Z, et al. Neural attentive session-based recommendation. In: *Proceedings of the 2017 ACM Conference on Information and Knowledge Management*, Singapore: ACM, November 6–10, 2017. pp. 1419–1428.
 37. Zhu N, Cao J, Liu Y, et al. Sequential modeling of hierarchical user intention and preference for next-item recommendation. In: *Proceedings of the 13th International Conference on Web Search and Data Mining*, Houston, TX: ACM, February 3–7, 2020. pp. 807–815.
 38. Chen T, Yin H, Chen H, et al. Air: Attentional intention-aware recommender systems. In: *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, Macao, China: IEEE, April 8–11, 2019. pp. 304–315.
 39. Wang S, Hu L, Wang Y, et al. Intention2basket: A neural intention-driven approach for dynamic next-basket planning. In: *IJCAI*, 2020, 1.11–13, online.
 40. Li H, Wang X, Zhang Z, et al. Intention-aware sequential recommendation with structured intent transition. *IEEE Trans Knowl Data Eng.* 2021, PP(99):1–1.
 41. Wang C, Ma W, Zhang M, et al. Toward dynamic user intention: Temporal evolutionary effects of item relations in sequential recommendation. *ACM Trans Inform Syst.* 2020;39:1–33.
 42. Zhou G, Zhu X, Song C, et al. Deep interest network for click-through rate prediction. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, London, UK: ACM, August 19–23, 2018. pp. 1059–1068.
 43. Zhou G, Mou N, Fan Y, et al. Deep interest evolution network for click-through rate prediction. *Proc AAAI Conf Artif Intell.* 2019;33: 5941–5948.
 44. Feng Y, Lv F, Shen W, et al. Deep session interest network for click-through rate prediction. *arXiv preprint arXiv:1905.06482*, 2019.
 45. Zhu H, Li X, Zhang PY, et al. Learning tree-based deep model for recommender systems. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, London, UK: ACM, August 19–23, 2018. pp. 1079–1088.
 46. Rendle S, Freudenthaler C, Gantner Z, et al. BPR: Bayesian personalized ranking from implicit feedback. In: *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, Montreal, Quebec, Canada: AUAI Press, June 18–21, 2009. pp. 452–461.
 47. Kingma DP, Ba J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
 48. Li TY, Ma YK, Xu J, et al. Deep heterogeneous autoencoders for collaborative filtering. In: *2018 IEEE International Conference on Data Mining (ICDM)*, Singapore: IEEE, November 17–20, 2018. pp. 1164–1169.
 49. Bordes A, Usunier N, Garcia-Duran A, et al. Translating embeddings for modeling multi-relational data. *Adv Neural Inform Process Syst.* 2013: 2787–2795.
 50. Nguyen P, Tomeo P, Di Noia T, et al. An evaluation of SimRank and personalized PageRank to build a recommender system for the web of data. In: *Proceedings of the 24th International Conference on World Wide Web*, Florence, Italy: ACM, May 18–22, 2015. pp. 1477–1482.
 51. Haochao Ying, Fuzhen Zhuang, Fuzheng Zhang, et al. Sequential recommender system based on hierarchical attention network. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, Stockholm, Sweden: AAAI Press, July 13–19, 2018. pp. 3926–3932.
 52. Liu B, Guo W, Niu D, et al. GIANT: Scalable creation of a web-scale ontology. In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. Portland, Oregon, USA, June 14–19, ACM, 2020. pp. 393–409.

Cite this article as: Yang F, Li G, Yue Y (2022) Hierarchical user intention-preference for sequential recommendation with relation-aware heterogeneous information network embedding. *Big Data* 10:5, 466–478, DOI: 10.1089/big.2021.0395.

Abbreviations Used

DHA = deep heterogeneous autoencoders
 DIEN = Deep Interest Evolution Network
 DIN = Deep Interest Network
 DSIN = Deep Session Interest Network
 HIN = heterogeneous information network
 HRM = hierarchical representation model
 KA-MemNN = key-array memory network
 MRR = Mean Reciprocal Rank
 PRP = PageRank with Priors