

Article

Recommendation Algorithm Based on Heterogeneous Information Network and Attention Mechanism

Li Li ^{1,2} , Xiangquan Gui ^{1,*}  and Rui Lv ³¹ School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China; lili2021@lzu.edu.cn² School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China³ Department of Management, Lanzhou Institute of Technology, Lanzhou 730050, China; ylr0319@163.com

* Correspondence: xqgui@lut.edu.cn

Abstract: Heterogeneous information networks (HINs) contain a rich network structure and semantic information, which makes them commonly used in recommendation systems. However, most of the existing HIN-based recommendation systems rely on meta-paths for information extraction, lack meta-path information supplements, and rarely learn complex structure information in heterogeneous graphs. To address these issues, we develop a novel recommendation algorithm that integrates the attention mechanism, meta-paths, and neighbor node information (AMNRec). In the heterogeneous information network, the missing information of the meta-path is supplemented by extracting the information of users and items' neighbor nodes. The rich interactions between nodes are captured through convolution, and the embedded representation of nodes and meta-paths is obtained through the attention mechanism. TOP-N recommendation is completed by combining users, items, neighbor nodes, and meta-paths. Experiments on three public datasets show that AMNRec not only has the best recommendation performance but also has good interpretability of the recommendation results compared with the six recommendation benchmark algorithms.

Keywords: heterogeneous information network; meta-path; neighbor information; attention mechanism; recommendation system; convolutional neural network

**Citation:** Li, L.; Gui, X.; Lv, R.Recommendation Algorithm Based on Heterogeneous Information Network and Attention Mechanism. *Appl. Sci.* **2024**, *14*, 353. <https://doi.org/10.3390/app14010353>

Academic Editors: Xiaoke Ma and Jingjing Deng

Received: 28 November 2023

Revised: 25 December 2023

Accepted: 26 December 2023

Published: 30 December 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the rapid development of the Internet, the problem of information overload [1] has become more and more serious, and recommendation systems [2] play a pivotal role in a variety of online services, which can help users quickly and efficiently select the information they need in a huge amount of information, such as product recommendations on e-commerce platforms, movie recommendations on video websites, and natural language processing (NLP)-related tasks [3–6].

Early models of recommender systems, such as Collaborative Filtering (CF) [7], mainly used historical user–item interaction records to model user preferences for items. However, collaborative filtering-based recommendation algorithms usually suffer from the cold-start problem (both for the prediction of new items or new users), and many research studies [8,9] have attempted to use auxiliary information to improve the accuracy of recommender systems. For instance, Ling et al. [10] considered both rating and review information and proposed a unified model combining content information and collaborative filtering algorithm. Ali et al. [11] addressed the paper cold-start problem by fully considering the attribute information of papers (such as authors, conferences, tags, and topics), constructed multiple weighted bipartite graphs based on the attribute information, and jointly learned the second-order similarity between nodes of each bipartite graph to generate a paper recommendation. Covington et al. [12] implemented a two-stage recommendation framework for a YouTube audio recommendation system supplemented by audio information.

Guo et al. [13] applied multi-attribute features other than a user ID and project ID, including user age and device category, and added an attention mechanism to distinguish the weights of paper titles and abstracts.

With the popularity of deep learning (DL), collaborative filtering algorithms have also begun to use various DL models to obtain the representation of users and items, to make more efficient use of the historical interaction information between them, and to improve the performance and generalization ability of the recommendation system [14–16]. For example, He et al. [17] proposed a combination of matrix decomposition and the multi-layer perceptron algorithm. Ali et al. [18] proposed a GAN-based network embedding model to address the network sparsity problem. Liu et al. [19] proposed the MMGRec model to learn nodes through graph attention networks of embedding features for recommendation. Chen et al. [20] introduced an attention mechanism to effectively utilize review information.

Heterogeneous information networks (HINs), as an emerging direction, can naturally model complex objects and their rich relationships in recommender systems [21]. Therefore, some researchers have started to realize the importance and necessity of HIN-based recommendations. Feng et al. [22] proposed the OptRank approach to alleviating the cold-start problem by exploiting the heterogeneous information contained in social tags. Further, meta-path is widely used in learning recommendation algorithms based on heterogeneous information network representation, as it can extract rich semantic information. For example, Shi et al. [23] proposed a heterogeneous information network embedding method called HERec, which only considers meta-paths whose starting type is user type or item type. Hu et al. [24] proposed a contextual collaborative attention recommendation model, MCRec, based on rich meta-paths considering the influence between meta-paths and user-item pairs. Although meta-paths can extract semantics between nodes' information and feature information between nodes, they cannot fully extract information in the network. Therefore, Zhao et al. [25] proposed FMG, a meta-graph-based recommendation fusion framework on heterogeneous information networks. Jin et al. [26] proposed an innovative convolutional neighborhood-based HIN recommendation interaction model, NIREc, which can capture and aggregate rich interaction information at the node and path levels. Fang et al. [27] proposed a contrastive meta-learning framework on HINs named CM-HIN, which addresses the cold-start issue at both the data level and model level.

Although the above methods have achieved performance improvements for recommendations to some extent, the following problems still exist: how to supplement the missing information of meta-paths; how to learn the display representation of paths or meta-paths in recommendation tasks; how to fuse the direct and indirect information in heterogeneous information networks' assisted recommendations. To solve the above problems, this paper proposes a recommendation algorithm that integrates the attention mechanism, meta-paths, and neighbor node information (AMNRec). It uses the neighbor node information of users and items to mine and supplement the missing information of meta-paths. By introducing the attention mechanism in DL, the representation vectors of users and items under different meta-paths are fused, and the feature vector representation capability of the user and item node sequence is enhanced to generate the final representation of users and items. The algorithm captures and fuses indirect information in the HIN at the node level and path level to achieve more accurate recommendations.

2. Problem Definition

2.1. Heterogeneous Information Network

A heterogeneous information network (HIN) is an information network, given a network $G = \{V, E\}$, where V is the set of objects and E is the set of links, if there exists a mapping function $\varphi: V \rightarrow A$ of node types and a mapping function $\psi: E \rightarrow R$ of edge types, where nodes and edges are mapped to a specific type. An information network is said to be heterogeneous if the number of object types $|A| > 1$ or the number of link types $|R| > 1$ in an information network. An example of music recommendation based on a

heterogeneous information network is given in Figure 1, from which it can be seen that HIN contains multiple types of entities, which are connected by different types of relationships.

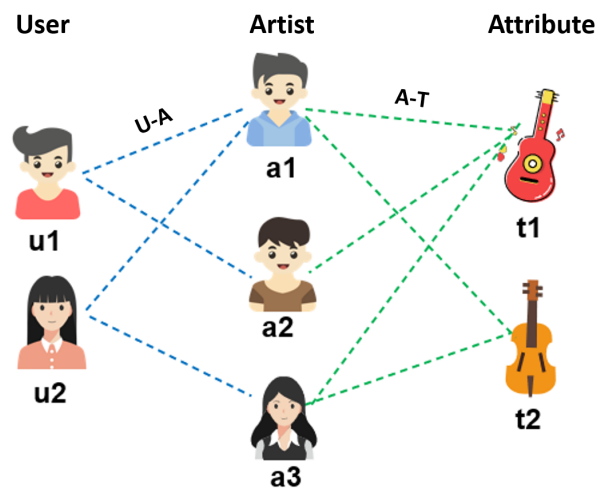


Figure 1. Example of heterogeneous information network that contains three types of nodes (user, artist, attribute) and two types of relations, user–artist (U–A) and artist–attribute (A–T).

2.2. Meta-Path

The meta-path P is defined as $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \dots \xrightarrow{R_l} A_{l+1}$ (abbreviated as $A_1 A_2 \dots A_{l+1}$), which represents the composite relation $R_1 \circ R_2 \circ \dots \circ R_l$ between objects A_1, A_2, \dots, A_{l+1} , and \circ denotes the synthesis operator on the relation [12]. There exist multiple specific paths under the meta-path P , called path instances, denoted as p . As shown in Figure 1, the user u_1 and artist a_3 can be connected by multiple meta-paths, such as $u_1 - a_1 - u_2 - a_3$ (UAUA) and $u_1 - a_1 - t_1 - a_3$ (UATA). For the interaction between u_1 and a_3 , different meta-paths convey different semantic information, e.g., a path instance $u_1 - a_1 - u_2 - a_3$ under the meta-path UAUA indicates that a user u_2 who has the same hobby as user u_1 also likes artist a_3 , and path instances under different meta-paths can provide more information for the recommendation.

2.3. First-Order Similarity

First-order similarity [28] refers to the similarity of two directly connected nodes; when there are no edges connected, the similarity is 0. When connected and with large weights, the similarity is high; as in Figure 2, nodes 5 and 6 have high first-order similarity, and nodes 5 and 6 are one-hop neighbors.

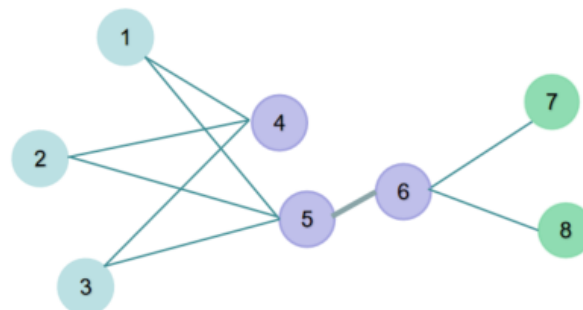


Figure 2. Example of node similarity. Nodes 5 and 6 are directly connected through a strong tie, so they have a high similarity. Nodes 4 and 5 do not have directly connected edges, but their similarity is also high due to the presence of common neighbor nodes 1, 2, and 3.

2.4. Second-Order Similarity

Second-order similarity refers to the similarity of neighbor nodes, i.e., nodes with the same neighbors are considered to be similar; if there are many common neighbor nodes and the weight is large, the similarity is high. As shown in Figure 2, there are no directly connected edges between nodes 4 and 5, but because there are common neighbors (nodes 1, 2, 3), nodes 4 and 5 are two-hop neighbors.

2.5. Attention Mechanism

The attention mechanism is an important technique in deep learning that is used to weight the input data of a model so that the model can pay more attention to important information. It has become very popular in the fields of computer vision and NLP, and is also widely used in recommendation systems [29]. In graph data, the connection relationship between nodes forms the graph structure, and the relationship and weight between nodes are usually very complex [30–33]. The graph attention mechanism can help models deal with this complexity effectively, assigning different attention weights to nodes by calculating the correlation between each node and its neighbors to better capture the dependencies between nodes.

3. Main Model

The general framework of the proposed recommendation algorithm model (NMRec) in this paper is shown in Figure 3. First, a meta-path containing $\langle \text{user}, \text{item}, \text{one-hop}, \text{and two-hop neighbor information of user and item} \rangle$ is utilized as input; then, convolution operations are introduced to generate potential interactions between meta-paths; after that, key interactions are captured and information is aggregated through the attention mechanism; finally, the model provides the final prediction.

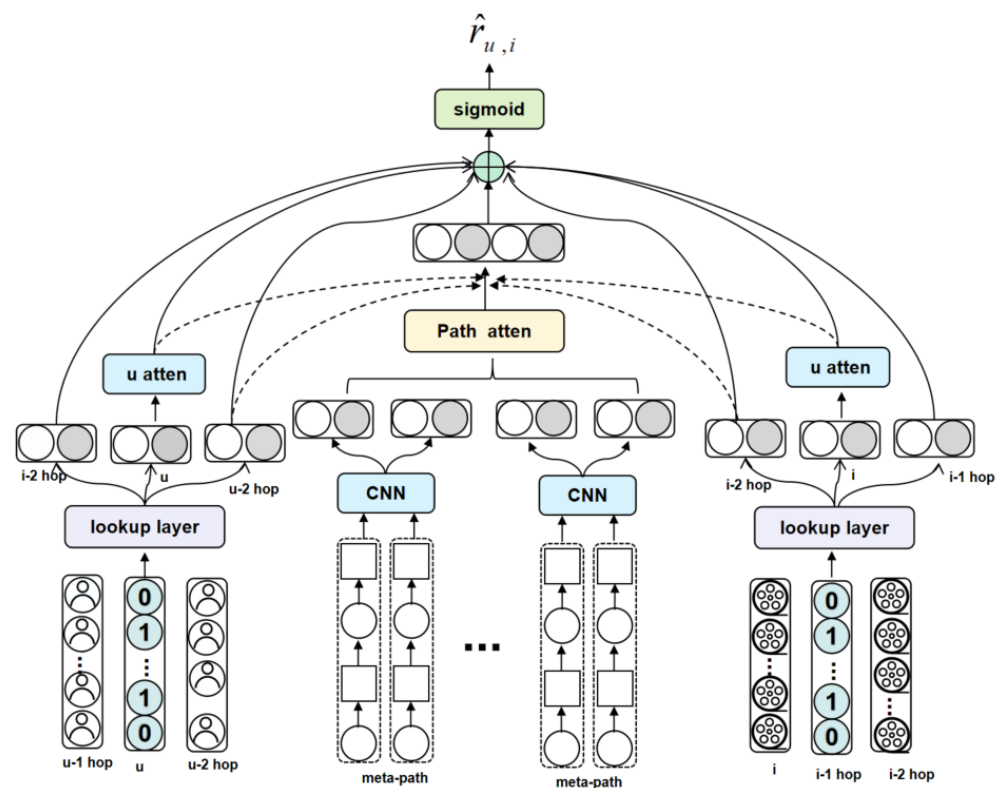


Figure 3. The overall framework of NMRec model.

3.1. Node and Neighbor Information Embedding

To learn the attribute information of users and items in the heterogeneous information network, this model uses HIN2vec [34] to represent the nodes in the heterogeneous

information network as low u dense vectors, and x_u, y_i are the learned characteristic vector embedding of users u and items i .

After obtaining the user and item node embedding, we need to study how to extract the missing information of users and items more efficiently. We consider that in practical recommendation tasks, people who become friends often have similar interests, and each user is influenced by the preferences of family, friends, and colleagues. Therefore, we extract the nodes' one-hop neighbor information to supplement the nodes' missing information. In this paper, we calculate the cosine similarity of vectors between nodes and establish the connected edges between nodes to aggregate the user and item one-hop neighbor information according to the similarity.

Since the one-hop neighbor links only account for a small percentage, they are not enough to represent the global information of users and items. Considering that in social networks, two people have more identical friends, then the probability of becoming friends between them is greater. Therefore, this paper introduces the two-hop neighbor information to supplement the richness of the recommendation algorithm and mine the users' real interest. This cannot be satisfied only by the one-hop neighbor information; that is, the more similar users are not necessarily the better, and complementary users can sometimes provide more useful information. Let $P_u = (w_{u^1}, \dots, w_{u^{|v|}})$ denote the first-order proximity of node u to other vertices. Then, the second-order proximity between u and v is determined by the cosine similarity between p_u and p_v . The connected edges between nodes are built based on the similarity to aggregate the second-hop neighbor information of users and items.

In real life, the friends and family that have the greatest impact on users are concentrated in the scope of second-hop neighbors, and beyond the second-hop neighbors, there is almost no influence on the node. In addition, considering the amount of data, processing speed, and other factors, users and items are only aggregated to the second-hop neighbor information. In this paper, referring to the network node aggregation method of HAN [17], we first convert the neighbor information into feature vectors with the same dimensionality. After that, considering that different nodes have different importance to different users and items, we use the attention mechanism to learn the weight size of each node. Then, *Softmax* normalizes the user and item information with a function that imposes the attention mechanism to obtain weight coefficients of each feature vector. Finally, we obtain the node aggregation representation, calculated as:

$$U = \sigma \left(\sum_{i \in N'(u)} e_i M_u \right) \quad (1)$$

$$I = \sigma \left(\sum_{j \in N'(i)} e_j M_i \right) \quad (2)$$

where M_u and M_i are learnable parameters, $N'(u)$ denotes the one-hop and two-hop neighbor information of user and user, and $N'(i)$ denotes the one-hop and two-hop neighbor information of item and item. The aggregated one-hop neighbor information of the user and item are U_1 and I_1 , and the aggregated two-hop neighbor information of the user and item are U_2 and I_2 .

3.2. Path Instance Embedding

Meta-paths provide special scene information, which includes link information between nodes and node attribute information. Learning the display representation of meta-paths can capture the scene information provided by meta-paths and can be used to explore the correlation between meta-paths for better extraction of information in heterogeneous networks for a recommendation. How to obtain high-quality path instances based on meta-paths is the key to efficient extraction of the meta-path information. The existing

meta-path information extraction based on heterogeneous information networks mainly uses a random wandering strategy to generate path instances, which are sampled with equal probability to the output nodes. However, the path instances sampled by such simple random wandering are generally of poor quality and are not suitable for a recommendation. Therefore, the priority is measured by calculating the similarity of feature vectors between different nodes, then the average value of the calculated similarity between nodes is ranked, and finally, the K path instances with the highest average similarity are obtained and input to the recommendation model.

The path instance of a meta-path is a sequence of entity nodes, and a convolutional neural network (CNN) is used to convert the path instance into a low-dimensional feature vector. Given a path instance $x_m^1 - x_m^2, \dots, x_m^n$ where $x_m^i \in R^{d \times 1}$ ($1 \leq i \leq n$) is the dimensional embedding of each node in the path instance, $[x_m^1 - x_m^2, \dots, x_m^n]^T$ denotes the representation matrix stitched together from the node sequence, Θ denotes all relevant parameters in the CNN, and the process of learning the path instance representation is as follows:

$$x_m = \text{CNN}([x_m^1 - x_m^2, \dots, x_m^n]^T, \Theta) \quad (3)$$

The meta-path can generate multiple path instances, so the embedding of the meta-path is generated by the maximum pooling operation and lets the embedding of K path instances obtained under the meta-path P be x_1, x_2, \dots, x_k , resulting in the meta-path embedding as

$$c_p = \text{max-pooling}(x_1, x_2, \dots, x_k) \quad (4)$$

3.3. Obtain Meta-Path Embedding Based on the Attention Mechanism

After obtaining a single meta-path embedding, the traditional meta-path aggregation method is to average the pooling operation to obtain the context representation of the meta-path c . This simple average pooling operation does not take into account the impact of the users and items involved on the meta-path, and the lack of meta-path has no consideration of semantic information in the interaction.

After obtaining a single meta-path embedding, the traditional meta-path aggregation method is an averaging pooling operation to obtain the contextual representation $c_{u \rightarrow i}$ of the meta-paths. This simple averaging pooling operation does not consider the influence of the involved users and items on the meta-paths and lacks the consideration that meta-paths have different semantic information in the interaction. In practical recommendation tasks, different meta-paths have different semantics among the same set of users and items, and different users have different preferences for meta-paths. Thus, our model learns the attention weights of users and items on meta-paths in response to interactions and given an initial representation of x_u users and an initial representation y_i of items, a two-layer architecture is used to implement this attention mechanism as:

$$\alpha_{u,i,\rho}^{(1)} = f(W_u^{(1)} x_u + W_i^{(1)} y_i + W_\rho^{(1)} c_\rho + b^{(1)}) \quad (5)$$

$$\alpha_{u,i,\rho}^{(2)} = f(W^{(2)} \alpha_{u,i,\rho}^{(1)} + b^{(2)}) \quad (6)$$

where $W_u^{(1)}, W_i^{(1)}, W_\rho^{(1)}$ denote the first layer weight matrix, $b^{(1)}$ is the first layer bias vector, $W^{(2)}$ and $b^{(2)}$ denote the second layer weight vector and bias vector, respectively, and $f(\cdot)$ is the *ReLU* activation function. After that, using the *Softmax* function to normalize the attention scores of all the above meta-paths, the final meta-path attention weights are obtained as

$$\alpha_{u,i,\rho} = \frac{\exp(\alpha_{u,i,\rho}^{(2)})}{\sum_{\rho' \in M} \exp(\alpha_{u,i,\rho'}^{(2)})} \quad (7)$$

The meta-path attention weight can be interpreted as the contribution of the meta-path to the interaction between u and i . After obtaining the attention value $\alpha_{u,i,\rho}$ of the

meta-path, the enhanced meta-path-based context is calculated according to the following weighted summation:

$$c_{u \rightarrow i} = \sum_{\rho \in P} \alpha_{u,i,\rho} \cdot c_{\rho} \quad (8)$$

where c_{ρ} is the representation of a single meta-path obtained from Equation (4), $\alpha_{u,i,\rho}$ is the attention weight generated by each interaction, and P is the set of meta-paths.

3.4. Obtain User and Item Embedding Based on the Attention Mechanism

Given a user and an item, the meta-path connecting them provides important semantic information that may affect the original representations of the user and the item. Given the original representations of the user x_u , item y_i , and the representation $c_{u \rightarrow i}$ of the meta-path context, a single-layer neural network is used to compute the attention vectors β_u and β_i of the user u and item i , respectively, as:

$$\beta_u = f(W_u x_u + W_{u \rightarrow i} c_{u \rightarrow i} + b_u) \quad (9)$$

$$\beta_i = f(W'_i y_i + W'_{u \rightarrow i} c_{u \rightarrow i} + b'_i) \quad (10)$$

where W_* and W'_* denote the weight matrix, b_u and b'_i denote the bias vector of the user attention layer, respectively, and $f(\cdot)$ is the *ReLU* activation function.

The attention vectors β_u and β_i are used to augment the original representations of users and goods, respectively, and the dot product of the attention vectors is used to obtain the user and goods representations as:

$$\tilde{x}_u = \beta_u \odot x_u \quad (11)$$

$$\tilde{y}_i = \beta_i \odot y_i \quad (12)$$

3.5. The Complete Model

The model obtains an enhanced representation \tilde{x}_u for a user u , a representation \tilde{y}_i for item i , a representation $c_{u \rightarrow i}$ based on the meta-path context, one-hop neighbor aggregations U_1 and I_1 for users and items, respectively, and two-hop neighbor aggregations U_2 and I_2 for users and items, respectively. The two-hop neighbor information enhances the generalization capability of the model by splicing the above six representation vectors as follows:

$$x_{u,i} = \tilde{x}_u \oplus U_2 \oplus c_{u \rightarrow i} \oplus I_2 \oplus \tilde{y}_i \quad (13)$$

$$F_{u,i} = U_1 \oplus I_1 \quad (14)$$

$$S_{u,i} = U_2 \oplus I_2 \quad (15)$$

The model first inputs $x_{u,i}$, $F_{u,i}$, and $S_{u,i}$ into the hidden layer after splicing, outputs $\tilde{x}_{u,i}$, $\tilde{F}_{u,i}$, and $\tilde{S}_{u,i}$ with *ReLU* as the activation function, then performs the summation operation of mutual complementary information, and finally, the output layer is a *sigmoid* function with the formula

$$\hat{r}_{u,i} = \text{Sigmod}(\tilde{x}_{u,i} + \tilde{F}_{u,i} + \tilde{S}_{u,i}) \quad (16)$$

Only implicit feedback is available in the recommendation task. Therefore, the optimization objective of learning model parameters by negative sampling techniques for interaction $\langle u, i \rangle$ is

$$L_{u,i} = -\log(\hat{r}_{u,i}) - E_{j \sim P_{neg}} [\log(1 - \hat{r}_{u,j})] \quad (17)$$

where the first term $\hat{r}_{u,i}$ models the observed interactions and the second term models the negative samples obtained by sampling from the noisy distribution P_{neg} . In this paper, the P_{neg} distribution is set to a uniform distribution.

3.6. The Model Discussion

In summary, the model has the following three main advantages.

- (1) **Effective fusion of multi-source information using second-hop neighbor information:** The introduction of second-hop neighbor information in this model provides the model with richer context and auxiliary information, which helps to better understand the relationship between users and items and realizes the effective fusion of multiple information of users, items, and meta-paths. In this way, the model can understand user preferences and item characteristics more comprehensively, thus making up for the limitations of a single data source, further enhancing the generalization ability of the model, and improving the accuracy and diversity of recommendations.
- (2) **Reinforcement learning mechanism:** The model uses a negative sampling technique to learn model parameters, similar to the “reward” mechanism in reinforcement learning. Through negative sampling technology, the model can effectively learn under limited feedback, which helps the model to better adapt to the data distribution and improve the prediction ability of unobserved interactions.
- (3) **Interpretability and efficiency:** The model can provide users with an explicable reasoning process and enhance the trust and acceptance of recommendation results. For example, through the interpretation of meta-path and neighbor information, users can better understand how recommendations are made. The model structure is flexible and efficient, enabling fast training and reasoning on large datasets. This high efficiency enables the model to provide a personalized recommendation service to users more quickly and provides an effective solution to solve the problem of data sparsity and cold starts.

4. Experiment and Analysis

4.1. Dataset

In this paper, we use three widely used datasets: the MovieLens (<https://grouplens.org/datasets/movielens>, accessed on 27 November 2023) movie dataset, the LastFM (<https://www.last.fm>, accessed on 27 November 2023) music dataset, and the Yelp (<https://www.yelp.com/dataset>, accessed on 27 November 2023) business dataset. The MovieLens dataset contains information about users, movies, and users’ ratings of movies. The LastFM dataset contains information about users, artists, and the number of times users listen to the artists’ songs. The Yelp dataset contains information about the properties of companies and users’ ratings of them. The details of the dataset are shown in Table 1. Also, the last column of Table 1 is the meta-path used for the experiment, and a maximum of four short meta-path steps are chosen to avoid introducing noise.

Table 1. Statistical information of the two datasets.

Datasets	Relations (A–B)	#A	#B	#A–B	Meta-Paths
MovieLens	User–Movie	943	1682	100,000	UMUM
	User–User	943	943	47,150	UMGM
	Movie–Movie	1682	1682	82,789	UUUM
	Movie–Genre	1682	18	2861	UMMM
LastFM	User–Artist	1892	17,632	92,834	UATA
	User–User	1892	1892	18,802	UAUA
	Artist–Artist	17,632	17,632	153,399	UUUA
	Artist–Tag	17,632	11,945	184,941	UUA
Yelp	User–Business	16,239	14,284	198,397	UBUB
	User–User	16,239	16,239	158,590	UUB
	Business–City (Ci)	14,267	47	14,267	UBCiB
	Business–Category (Ca)	14,181	511	40,009	UBCaB

4.2. Benchmark Model

To demonstrate the validity of the proposed model, two types of representative recommendation algorithms are selected as comparison models in this experiment. They are collaborative filtering-based methods (ItemKNN, BPR, MF, NeuMF) that only consider implicit feedback and HIN-based methods (FMG, MCRec). The comparison models are given below:

- ItemKNN [35], a classical collaborative filtering recommendation algorithm that makes recommendations based on the historical interaction behavior of users and items.
- BPR [36], a Bayesian-based personalized ranking model, which is a typical pairwise learning personalized ranking method based on implicit feedback.
- MF [37], a standard matrix decomposition algorithm that uses the cross-entropy loss function instead of the original loss function for TOP-N recommendation.
- NeuMF [17], a neural network-based ranking recommendation algorithm consisting of matrix decomposition and a multilayer perceptron.
- FMG [25], an advanced recommendation algorithm for heterogeneous information networks. It uses a two-step process: information extraction (matrix decomposition of individual meta-path connection matrices) and information utilization (recommendation using FM).
- MCRec [24], a deep network model with a cooperative attention mechanism for TOP-N recommendation in multi-source information networks.

4.3. Parameter Setting

The model is implemented based on the Pytorch deep learning framework. Similar to [24], using the common setting method, considering the convergence speed, memory consumption, and other aspects, we set the model parameters as follows: the model parameters are randomly initialized using a Gaussian distribution, the model is optimized using Adaptive Moment Estimation (Adam), the learning rate is set to 0.001, the regularization parameter λ is set to 0.0001, the CNN convolutional kernel size is set to 3, the embedding dimension of users and items is 128, the output dimension is 32, the batch size is set to 256, the epoch is set to 50, and five items with no interaction record with the target user is randomly selected as a negative sample. For other comparison methods, this experiment uses 10% of the training data as the validation set to optimize its parameters. All experiments are conducted on a machine with four GPUs (NVIDIA GTX-1080 \times 4), one CPU (Xeon W-1350 @ 3.30 GHz), and 48 GB memory.

4.4. Evaluation Indicators

To measure the effectiveness of each model in Top-N recommendation, this experiment uses two widely used evaluation metrics: the top K recall (Recall@K) and the normalized discounted cumulative gain (NDCG@K), where recall is used to test the ability of the model to discriminate whether users are interested in an item or not, and the normalized discounted cumulative gain focuses on the position of the user's preferred item in the recommendation list.

4.5. Ablation Experiments

To verify the effectiveness of the model in supplementing the multi-hop neighbor information and attention mechanism, three variants of the model were prepared as follows, and the results of the ablation experiments are shown in Table 2.

- AMNRec 2-hop: This is a variant of AMNRec that removes the first-order neighbor information and retains the second-order neighbor information and meta-path embedding information.
- AMNRec 1-hop: This is a variant of AMNRec that removes the second-order neighbor information and retains the first-order neighbor information and meta-path embedding information.

- AMNRec-atten: This is another variant of AMNRec that keeps only the attention components of meta-paths and removes the attention components of user- and item-directed one-hop and two-hop neighbors.
- AMNRec-total: This is the complete model.

Table 2. Results of ablation experiments performed on MovieLens, LastFM, and Yelp.

Model	MovieLens		LastFM		Yelp	
	Recall@10	NDCG@10	Recall@10	NDCG@10	Recall@10	NDCG@10
AMNRec 2-hop	0.3533	0.6889	0.6067	0.8476	0.7862	0.6446
AMNRec 1-hop	0.3687	0.7056	0.6123	0.8525	0.7972	0.6523
AMNRec-atten	0.3606	0.7094	0.5956	0.7978	0.7938	0.6465
AMNRec-total	0.3762	0.7125	0.6234	0.8639	0.8127	0.6675

4.6. Model Evaluation and Analysis

The user–item interaction records in the datasets were randomly divided into training and test sets, with 80% as the training set and 20% as the test set. AMNRec was compared with six benchmark algorithms, and the experimental results are shown in Table 3. The following can be seen from the tables:

- (1) It is observed that the complete AMNRec model in this paper consistently outperforms the other comparable models on the three datasets, verifying the effectiveness of AMNRec on the ranking recommendation task and indicating the effectiveness of AMNRec in proposing the use of neighbor information to supplement the missing information of meta-paths.
- (2) Considering the three variants of AMNRec, it can be found that the overall performance is as follows: AMNRec-total > AMNRec 1-hop > AMNRec 2-hop > AMNRec-atten. The results show that although AMNRec 1-hop is competitive with other methods, it is still worse than the complete AMNRec-total, and it can be seen that the one-hop neighbor information supplementation plays a crucial role. AMNRec-atten proves the effectiveness of introducing an attention mechanism to extract multi-hop neighbor information.
- (3) Among these compared methods, the HIN-based methods (FMG, MCRec) are superior to the collaborative filtering methods (ItemKNN, BPR, MF, and NeuMF) in most cases, indicating the usefulness of heterogeneous information embedding. In addition, NeuMF also achieves competitive performance due to its use of neural networks, but its performance is still inferior to MCRec due to the lack of heterogeneous information. Among the benchmark methods for multi-source information networks, MCRec performs the best, indicating that collaborative attention can make good use of meta-path-based contexts for a recommendation, thus improving the recommendation efficiency.

Table 3. Results of effectiveness experiments on MovieLens, LastFM, and Yelp.

Model	MovieLens		LastFM		Yelp	
	Recall@10	NDCG@10	Recall@10	NDCG@10	Recall@10	NDCG@10
ItemKNN	0.1536	0.5161	0.4513	0.7981	0.5421	0.5378
BRP	0.1946	0.6459	0.4492	0.8099	0.5504	0.5549
MF	0.2053	0.6511	0.4634	0.7210	0.5350	0.5322
NeuMF	0.2090	0.6587	0.4678	0.8104	0.5857	0.5713
FMG	0.2165	0.6682	0.4916	0.8263	0.5951	0.5861
MCRec	0.2256	0.6900	0.5068	0.8526	0.6326	0.6301
AMNRec-total	0.4062	0.7275	0.6234	0.8639	0.8127	0.6675

4.7. Parameter Analysis

To analyze the effect of different meta-paths on the recommendation performance, meta-paths are gradually added to the proposed model. From Figure 4, it can be observed that the overall performance of the model gradually improves with the addition of meta-paths. Also, different meta-paths have different weights on the recommendation performance improvement: adding UUUM and UMMM is more important for MovieLens.

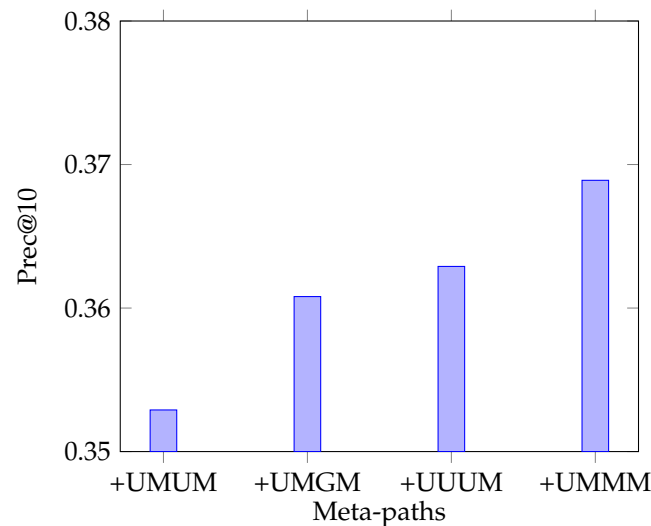


Figure 4. Stepwise addition of meta-paths affects model performance for MovieLens.

The meta-paths have different weights for different users and items. In this paper, the MovieLens dataset is taken as an example, and it can be found in Figure 5 that UMGM contributes the most to the interaction. By looking at the dataset, it is found that the movie genre of “The Eighth Day” is opera, which happens to be the favorite movie genre of user “u782”. This partly explains why the meta-path UMGM contributes the most to the interaction and proves the interpretability of the model.

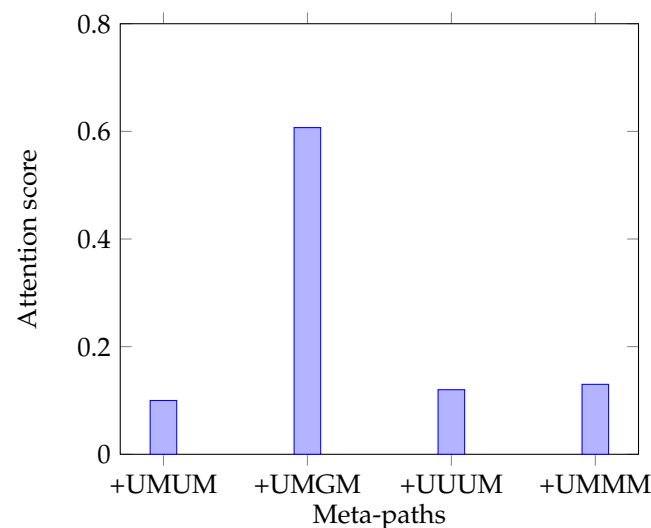


Figure 5. Attention scores for different meta-paths for MovieLens.

This experiment investigates the output layer dimension size, setting different output layer dimensions of 8, 16, 32, 64, as shown in Figure 6. The performance gradually improves as the output layer dimension grows, reaching the best around dimension 32, and then starts to decline. The reason may be that the model needs a suitable dimension to encode the semantic information, and larger dimensions may introduce additional redundancy.

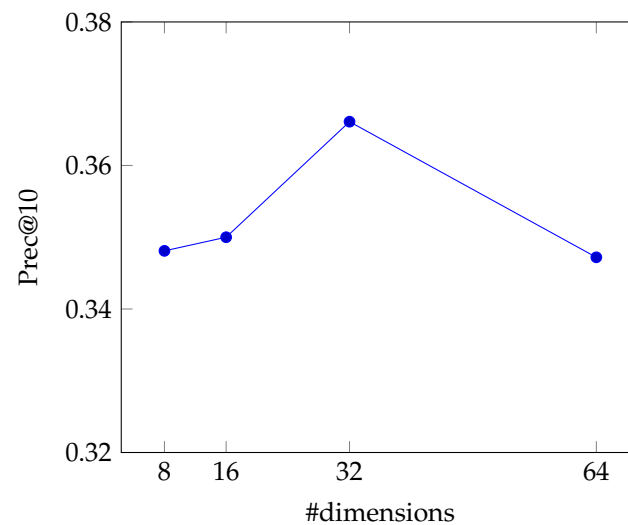


Figure 6. The effect of dimension size of the output layer on model performance.

The effect of the number of negative samples on the model performance was investigated by varying the number of negative samples in the set of 1, 3, 5, 7, 9. As shown in Figure 7, the model achieves the optimal performance when the number of negative samples is five. The experiment proves that the number of negative samples should not be set too large or too small because it may lead to overfitting and underfitting.

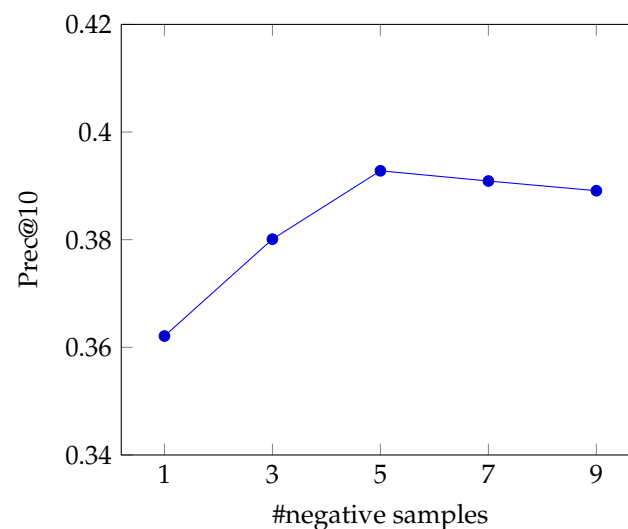


Figure 7. The effect of negative samples on model performance.

5. Conclusions

In this paper, a recommendation algorithm (AMNRec) based on neighbor nodes and meta-paths is proposed for TOP-N recommendation in heterogeneous information networks by supplementing the missing information of meta-paths with one-hop and two-hop neighbor information based on users and items, thus fully learning the information in the heterogeneous graph and obtaining a high-quality feature representation. The user, item, and meta-path information are corrected using the attention mechanism, and the effectiveness of AMNRec in recommendation performance is demonstrated experimentally.

This model provides a promising way to improve the performance of the recommendation system, and it is considered to further improve the recommendation method in the future studies to expand more practical application fields, for example, in the field of health care, exploring the association between diseases and symptoms or the interaction between drugs and diseases to provide doctors with more accurate diagnoses and treatment

recommendations; in natural language processing, semantic and contextual information in text is mined to improve the accuracy of text classification and sentiment analysis.

Author Contributions: Conceptualization, L.L.; methodology, X.G.; software, R.L.; formal analysis, L.L.; investigation, L.L.; resources, X.G.; data curation, R.L.; writing—original draft preparation, L.L.; writing—review and editing, X.G.; funding acquisition, X.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Key Research and Development Program of Gansu Province (No. 22YF7GA159), Soft Science Special Project of Gansu Basic Research Plan (No. 22JR4ZA084), and Industry Support Program of Gansu Provincial Department of Education (No. 2023CYZC-25).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: In this paper, we evaluate our model on three widely used open datasets, namely, MovieLens (<https://movielens.org>), LastFM (<https://www.last.fm>), and Yelp (<https://www.yelp.com>), all accessed on 27 November 2023, which can be easily obtained from the Internet.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- Jacoby, J. Perspectives on Information Overload. *J. Consum. Res.* **1984**, *10*, 432–435. [\[CrossRef\]](#)
- Park, D.H.; Kim, H.K.; Choi, I.Y.; Kim, J.K. A literature review and classification of recommender systems research. *Expert Syst. Appl.* **2012**, *39*, 10059–10072. [\[CrossRef\]](#)
- Dong, C.; Xie, Y.; Ding, B.; Shen, Y.; Li, Y. Collaborating Heterogeneous Natural Language Processing Tasks via Federated Learning. *arXiv* **2022**, arXiv:2212.05789
- Chifu, A.G.; Fournier, S. Sentiment Difficulty in Aspect-Based Sentiment Analysis. *Mathematics* **2023**, *11*, 4647. [\[CrossRef\]](#)
- Massaro, A.; Maritati, V.; Galiano, A. Automated Self-learning Chatbot Initially Build as a FAQs Database Information Retrieval System: Multi-level and Intelligent Universal Virtual Front-office Implementing Neural Network. *Informatica* **2018**, *42*, 515–525. [\[CrossRef\]](#)
- Yeshambel, T.; Mothe, J.; Assabie, Y. Learned Text Representation for Amharic Information Retrieval and Natural Language Processing. *Information* **2023**, *14*, 195. [\[CrossRef\]](#)
- Herlocker, J.L.; Konstan, J.A.; Terveen, L.G.; Riedl, J.T. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.* **2004**, *22*, 5–53. [\[CrossRef\]](#)
- Yin, H.; Cui, B.; Sun, Y.; Hu, Z.; Chen, L. LCARS: A Spatial Item Recommender System. *ACM Trans. Inf. Syst.* **2014**, *32*, 11. [\[CrossRef\]](#)
- Ma, X.; Sun, P.; Qin, G. Identifying Condition-Specific Modules by Clustering Multiple Networks. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2018**, *5*, 1636–1648. [\[CrossRef\]](#)
- Ling, G.; Lyu, M.R.; King, I. Ratings Meet Reviews, a Combined Approach to Recommend. In Proceedings of the RecSys '14: Proceedings of the 8th ACM Conference on Recommender Systems, Foster City, CA, USA, 6–10 October 2014; pp. 105–112. [\[CrossRef\]](#)
- Ali, Z.; Qi, G.; Muhammad, K.; Ali, B.; Abro, W.A. Paper recommendation based on heterogeneous network embedding. *Knowl. Based Syst.* **2020**, *210*, 106438. [\[CrossRef\]](#)
- Covington, P.; Adams, J.; Sargin, E. Deep Neural Networks for YouTube Recommendations. In Proceedings of the RecSys '16: 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 191–198. [\[CrossRef\]](#)
- Guo, H.; Tang, R.; Ye, Y.; Li, Z.; He, X. DeepFM: A Factorization-Machine Based Neural Network for CTR Prediction. In Proceedings of the IJCAI'17: 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; AAAI Press: Washington, DC, USA, 2017; pp. 1725–1731. [\[CrossRef\]](#)
- Lu, C.T.; He, L.; Ding, H.; Cao, B.; Yu, P.S. Learning from Multi-View Multi-Way Data via Structural Factorization Machines. In Proceedings of the WWW '18: 2018 World Wide Web Conference, Geneva, Switzerland, 23–27 April 2018; pp. 1593–1602. [\[CrossRef\]](#)
- Wang, J.; Yu, L.; Zhang, W.; Gong, Y.; Xu, Y.; Wang, B.; Zhang, P.; Zhang, D. IRGAN: A Minimax Game for Unifying Generative and Discriminative Information Retrieval Models. In Proceedings of the SIGIR '17: 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 515–524. [\[CrossRef\]](#)
- Zheng, Y.; Tang, B.; Ding, W.; Zhou, H. A Neural Autoregressive Approach to Collaborative Filtering. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; Volume 48, pp. 764–773.
- He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; Chua, T.S. Neural Collaborative Filtering. In Proceedings of the WWW '17: 26th International Conference on World Wide Web, Geneva, Switzerland, 3–7 April 2017; pp. 173–182. [\[CrossRef\]](#)
- Ali, Z.; Qi, G.; Muhammad, K.; Kefalas, P.; Khusro, S. Global citation recommendation employing generative adversarial network. *Expert Syst. Appl.* **2021**, *180*, 114888. [\[CrossRef\]](#)

19. Liu, Z.; Zhang, Z. Top-N Recommendation Method for Graph Attention Based on Multi-level and Multi-view. *Comput. Sci.* **2021**, *48*, 104. [\[CrossRef\]](#)
20. Chen, C.; Zhang, M.; Liu, Y.; Ma, S. Neural Attentional Rating Regression with Review-level Explanations. In Proceedings of the WWW '18: 2018 World Wide Web Conference, Geneva, Switzerland, 23–27 April 2018.
21. Shi, C.; Li, Y.; Zhang, J.; Sun, Y.; Yu, P.S. A survey of heterogeneous information network analysis. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 17–37. [\[CrossRef\]](#)
22. Feng, W.; Wang, J. Incorporating Heterogeneous Information for Personalized Tag Recommendation in Social Tagging Systems. In Proceedings of the KDD '12: 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 1276–1284. [\[CrossRef\]](#)
23. Shi, C.; Hu, B.; Zhao, W.X.; Yu, P.S. Heterogeneous Information Network Embedding for Recommendation. *IEEE Trans. Knowl. Data Eng.* **2019**, *31*, 357–370. [\[CrossRef\]](#)
24. Hu, B.; Shi, C.; Zhao, W.X.; Yu, P.S. Leveraging meta-path based context for top-n recommendation with a neural co-attention model. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1531–1540.
25. Zhao, H.; Yao, Q.; Li, J.; Song, Y.; Lee, D.L. Meta-Graph Based Recommendation Fusion over Heterogeneous Information Networks. In Proceedings of the KDD '17: 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 24–27 August 2017; pp. 635–644. [\[CrossRef\]](#)
26. Jin, J.; Qin, J.; Fang, Y.; Du, K.; Zhang, W.; Yu, Y.; Zhang, Z.; Smola, A.J. An efficient neighborhood-based interaction model for recommendation on heterogeneous graph. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual, 6–10 July 2020; pp. 75–84.
27. Fang, Y.; Tan, Z.; Chen, Z.Y.; Xiao, W.D.; Zhang, L.L.; Tian, F. Contrastive Meta-learning on Heterogeneous Information Networks for Cold-start Recommendation. *J. Softw.* **2023**, *34*, 4548. [\[CrossRef\]](#)
28. Lu, P.; Zhang, Z. Critical nodes identification in complex networks via similarity coefficient. *Mod. Phys. Lett. B* **2022**, *36*, 2150620. [\[CrossRef\]](#)
29. Wen, P.; Yuan, W.; Qin, Q.; Sang, S.; Zhang, Z. Neural Attention Model for Recommendation Based on Factorization Machines. *Appl. Intell.* **2021**, *51*, 1829–1844. [\[CrossRef\]](#)
30. Ma, X.; Gao, L. Discovering Protein Complexes in Protein Interaction Networks via Exploring the Weak Ties Effect. *BMC Syst. Biol.* **2012**, *6*, 1–15. [\[CrossRef\]](#)
31. Huang, Z.; Zhong, X.; Wang, Q.; Gong, M.; Ma, X. Detecting Community in Attributed Networks by Dynamically Exploring Node Attributes and Topological Structure. *Knowl.-Based Syst.* **2020**, *196*, 1057601. [\[CrossRef\]](#)
32. Ma, X.; Tang, W.; Wang, P.; Guo, X.; Gao, L. Extracting Stage-specific and Dynamic Modules through Analyzing Multiple Networks Associated with Cancer Progression. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2018**, *15*, 647–658. [\[CrossRef\]](#)
33. Wu, W.; Ma, X. Network-Based Structural Learning Nonnegative Matrix Factorization Algorithm for Clustering of scRNA-Seq Data. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2023**, *20*, 566–575. [\[CrossRef\]](#) [\[PubMed\]](#)
34. Fu, T.Y.; Lee, W.C.; Lei, Z. HIN2Vec: Explore Meta-Paths in Heterogeneous Information Networks for Representation Learning. In Proceedings of the CIKM '17: 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 1797–1806. [\[CrossRef\]](#)
35. Sarwar, B.; Karypis, G.; Konstan, J.; Riedl, J. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web, Hong Kong, 1–5 May 2001; Association for Computing Machinery, Inc.: New York, NY, USA, 2001; pp. 285–295. [\[CrossRef\]](#)
36. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian Personalized Ranking from Implicit Feedback. In Proceedings of the UAI '09: Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Arlington, VA, USA, 18–21 June 2009; pp. 452–461.
37. Koren, Y.; Bell, R.; Volinsky, C. Matrix Factorization Techniques for Recommender Systems. *Computer* **2009**, *42*, 30–37. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.