

Design and Implementation of a Learning Resource Recommendation System based on User Habits Based on GNN

Jingxuan Lu ^{1, a}, YangKwon Jeong ^{1, b, *}, Jiaqi Xue ^{2, c}

¹ Computer Science, DongShin University, Naju, Korea

² Computer Science, Beihua University, Jilin, China

^a bh_ljx@163.com, ^{b, *} 81262657@qq.com, ^c 2466879100@qq.com

* Corresponding author: YangKwon Jeong (Email: 81262657@qq.com)

Abstract: This project aims to design and implement a learning resource recommendation system based on Graph Neural Networks (GNN). The system utilizes user learning habits as a foundation to provide personalized learning resource recommendations. By collecting and preprocessing user learning history data, and constructing a user-resource relationship graph, the GNN model is used to learn the representation vectors of users and resources. Combined with user habit features, appropriate recommendation algorithms are employed to recommend learning resources that align with their interests and habits.

Keywords: Graph Neural Networks; Personalized Recommendation; Data Sparsity.

1. Introduction

With the continuous development of information technology and the increasing popularity of the Internet, it has become a very common phenomenon for people to obtain information through online learning. At the same time, it also faces the challenge of massive online resources and information that make it difficult for users to choose and find information that meets their needs. Therefore, how to provide personalized recommendations to users through intelligent methods to meet their online needs has become an urgent problem to be solved. So, we decided to design and implement a learning resource recommendation system based on user habits and interests. We use graph neural network deep learning technology to analyze the big data that users browse online, recommend to users in an intelligent way, accurately deliver resources that meet their browsing habits and personalized needs, and use advanced technologies such as face recognition to identify users and make personalized recommendations. This learning resource recommendation system based on user habits is an effective solution proposed to address the above issues.

Graph neural network (GNN) is one of the most attractive recommendation algorithms. It is a new type of neural network, which originates from the development of Convolutional neural network (CNN). GNN contains rich relational information, which can effectively extract data features in the graph field and provide powerful functions for learning graph related data. Compared to traditional deep learning methods, GNN places more emphasis on the connections between nodes, effectively avoiding the problem of overlooked associations between users and items themselves. By iterating on nodes, GNN can more accurately describe the relationships between entities in the graph model, providing a new approach for recommendation system analysis and helping to mine deeper information.

This article provides an overview of the application of GNN in recommendation systems, showcasing its advantages of scalability and efficiency, and exploring its potential in

improving recommendation system performance. This paper introduces a personalized recommendation system based on graph neural network and advanced technology, which can collect and analyze big data that users browse on the Internet. By conducting a detailed analysis of users' internet browsing records, application usage, and browsing content, the system can capture their browsing interests and habits, providing strong data support for subsequent recommendations. In this process, we consider users and browsing resources as nodes in the graph and establish corresponding graph models based on their interrelationships. By applying deep learning techniques of graph neural networks, user data is modeled and predicted to achieve accurate prediction of user needs and behaviors. In order to further improve the personalization of recommendations, we have adopted advanced technologies such as facial recognition to identify users. By recognizing users' facial features, the system can more accurately determine their usage needs and interests, and recommend online resources that better meet personalized needs for users. This personalized recommendation method enables users to obtain content and services that suit their interests more satisfactorily, thereby improving the effectiveness and user experience of the recommendation system.

2. Related Work

2.1. Research Content

A learning resource recommendation system is a system that intelligently recommends learning resources that meet users' interests and needs. The system aims to provide more personalized and accurate recommendation of learning resources, improving the effectiveness and efficiency of network usage. The system should be able to recommend usage resources that meet users' needs based on factors such as their historical usage records, interests, and usage goals. Continuously optimize recommendation algorithms through algorithm and data analysis to improve the accuracy and precision of recommendations. And recommend various types of learning resources, including videos, articles, books,

courses, etc., to meet the needs of different users. Finally, the system should ensure the security and privacy of user data to avoid the leakage or abuse of user information.

2.2. Research Background

With the continuous development of information technology and the increasing popularity of the Internet, it has become a very common phenomenon for people to obtain information through online learning. At the same time, it also faces the challenge of massive online resources and information that make it difficult for users to choose and find information that meets their needs. Therefore, how to provide personalized recommendations to users through intelligent methods to meet their online needs has become an urgent problem to be solved. So, we decided to design and implement a learning resource recommendation system based on user habits and interests. We use graph neural network deep learning technology to analyze the big data that users browse online, recommend to users in an intelligent way, accurately deliver resources that meet their browsing habits and personalized needs, and use advanced technologies such as face recognition to identify users and make personalized recommendations.

3. GNN Algorithm

In random initialization, we only need to determine the range and distribution of initialization, and common methods include using uniform or Gaussian distributions to generate random values. For the random initialization of features of nodes and edges, set a range to randomly generate corresponding numerical values from the specified distribution.

3.1. Graph Convolution

Among them, $h_i^{(l+1)}$ is the new feature representation of node i in the $l+1$ layer, $h_j^{(l)}$ is the feature representation of node j in the l layer, $N(i)$ is the set of neighboring nodes of node i , c_{ij} is the normalization coefficient of the number of edges between node i and node j , and $W^{(l)}$ is the weight parameter of the l layer, σ is the Activation function. By performing convolution operations between node features and neighboring node features, utilizing the connectivity between nodes for feature updates and propagation

$$h_i^{(l+1)} = \sigma \left(\sum_{j \in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)} + W^{(l)} h_i^{(l)} \right)$$

3.2. Graph Pooling

$$H_g = \max_{i \in V} (U h_i)$$

Among them, H_g is the graph level feature vector, h_i is the feature vector of node i , and U is the channel transformation parameter. By selecting important nodes and edges, or segmenting and sampling the graph, subgraphs can be generated that represent the entire graph structure information. This subgraph can be used as input to the model for feature transfer and aggregation calculations.

3.3. Graph Attention

$$e_{ij} = \text{LeakyReLU}(\alpha \mathbf{W}_1 \mathbf{h}_i \parallel \parallel \mathbf{W}_2 \mathbf{h}_j)$$

$$\alpha_{ij} = \exp(e_{ij}) / (\sum_{k \in N(i)} \exp(e_{ik}))$$

$$h_i' = \sigma(j \in N(i) \sum (\alpha_{ij} W_3(h_j)))$$

Among them, e_{ij} is the edge weight calculated through the attention mechanism, α is the weight vector of the attention mechanism, and \parallel represents the vector concatenation operation, α_{ij} is the attention coefficient between node i and neighboring node j , and h_i' is the updated feature representation of node i .

The specific algorithm for adaptive initialization can be selected or designed based on the needs of the problem and the design of the network. It can be used in conjunction with different GNN models, aggregation methods, and feature update rules to improve the performance and adaptability of the model (Figure 1).

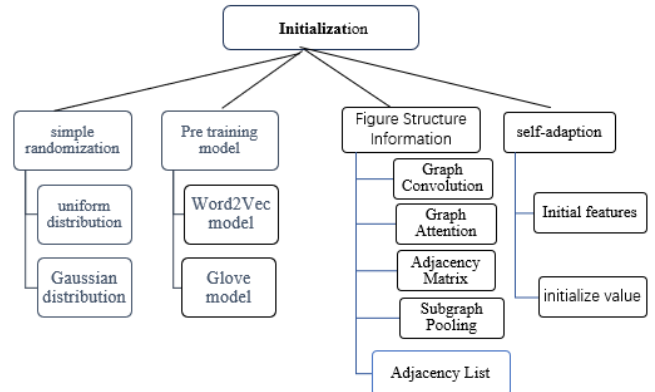


Figure 1. Initialization Model Diagram

3.4. Linear Transformation

For each layer of nodes, linear transformation is a basic computational operation. Its purpose is to multiply the input data by the weight matrix and add a bias vector to obtain an intermediate linear output. Perform nonlinear conversion on linear output and introduce nonlinear factors (Figure 2). Normalize the input data of each layer to a mean of 0 and a variance of 1. By reducing the distribution changes of input data and providing reliable gradient signals, the network is easy to train. The pooling operation reduces the size of feature maps and extracts spatial hierarchical information of features. Reduce the number of parameters and improve the computational efficiency of the network. Randomly discard a portion of the node's output. By randomly deactivating a portion of nodes, the interdependence between each node can be limited, enhancing the model's generalization ability.

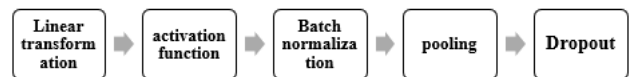


Figure 2. Number Linear Transformation Model.

4. Proposed Solutions and Solutions

4.1. Proposed Problem Solving

When using GNN to solve learning resource recommendation systems based on user habits, the following issues related to GNN are involved:

Model complexity and training efficiency: Complex GNN models may lead to slow training and require significant computational resources.

Data sparsity: Users tend to have very sparse data, which may affect the performance and recommendation accuracy of GNN models.

Hierarchical diversity: User habits data may have multiple

hierarchical structures, and how to flexibly model this structure is a challenge.

Information dissemination and long-distance dependence: Information dissemination in GNN is usually local, and information transmission between distant nodes may not be effective enough.

Context and temporal modeling: Learning resource recommendation needs to consider context and temporal relationships, and how to introduce them into the GNN model is an important issue.

Model interpretability: GNN models are usually black box models, making it difficult to explain their internal information transmission and decision-making processes.

4.2. Solution

To address the above issues, we have the following solution process to incorporate the user's node z_i and its weight w_i into the model. Firstly, we need to convert the nodes and edges in the graph into representations that can be processed by the computer. Nodes will be represented as vectors or matrices, and edges will be represented in the form of Adjacency matrix or adjacency list. The characteristics of each node and the characteristics of its neighbor nodes are aggregated, and the connection relationship and weight between nodes are determined according to the Adjacency matrix or adjacency list to calculate the aggregated neighbor feature representation. Perform a linear transformation on the aggregated neighbor features, multiplying them by the weight matrix and adding a bias vector. The Activation function is used for nonlinear conversion of the features after linear transformation. Use the nonlinear transformed features as new features for node transfer in subsequent layers. Repeat the feature transfer process until the set number of convolutional layers is reached. Based on the features transmitted through a series of graph convolutional layers, corresponding prediction tasks are performed in the output layer and the final results are obtained.

Through the above process, GNN can transfer and aggregate the input graph data through a series of graph convolutional layer features, and ultimately obtain the predicted results of the graph or node. This forward propagation process enables GNN to fully utilize the topology information between nodes and perform features between layers when processing graph structure data

Transfer and aggregate to extract meaningful representations and make predictions (Figure 3).

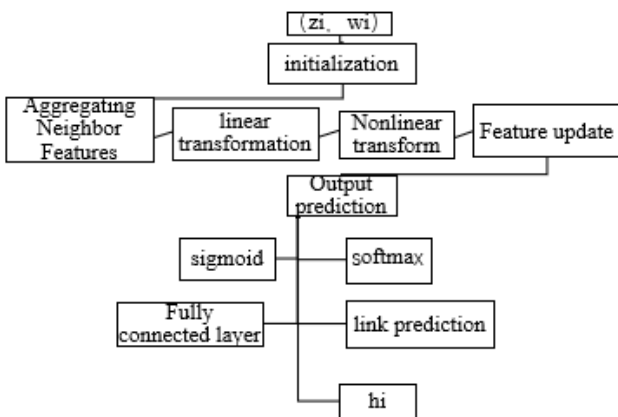


Figure 3. GNN flowchart.

5. Experiments

5.1. Datasets

Build a learning resource recommendation system based on user habits, including modules such as user information collection, data analysis, recommendation algorithms, and user feedback (Figure 4).

The results shown in the following figure show that as users use the recommendation system, the collection time of the dataset increases, and the browsing time of users shows a significant increase. One possible reason is that users' interest in learning resources has increased, and recommendation algorithms have more accurate analysis and judgment of users after collecting more data, thereby stimulating their use.

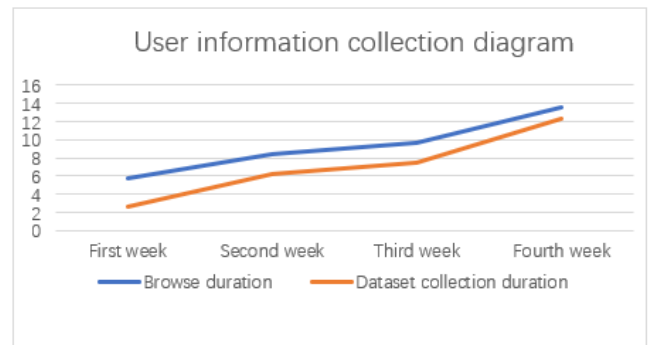


Figure 4. User Information Collection Diagram

5.2. Testing

The hidden layer dimension defines the dimension of the representation of each node in the GNN model. As shown in the above figure, a smaller hidden layer dimension may lead to information loss, resulting in underfitting of the model. A larger hidden layer dimension may lead to too many parameters, increasing computational and storage costs, and also being prone to overfitting. We used a lot of data to determine which dimension to choose (Figure 5)

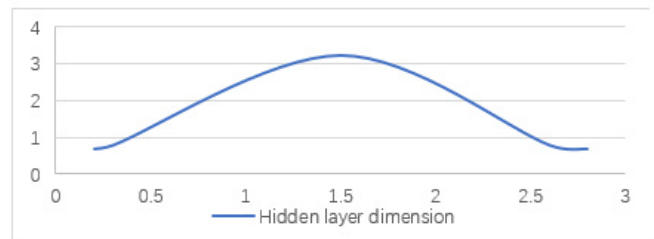


Figure 5. Hidden layer dimension diagram

5.3. Experimental Results

5.3.1. Test Procedure

Build a learning resource recommendation system based on user habits, including user information collection, and the data graph is the data representation of GNN compared with other models. From it, we can see that GNN can generally achieve better performance than traditional machine learning algorithms based on Feature engineering when processing graph

structure data. GNN can automatically learn the representation of nodes and edges from data, and capture the relationships between nodes through information transmission and aggregation.

5.3.2. Result Verification

Experimental verification: Through experiments, after

continuous iteration and training, the recommendation algorithm is optimized to improve recommendation accuracy and user satisfaction (Table 1).

Table 1. Comparison between GNN and other models

data set	Model accuracy	Local Receptive	Process Graph Data
Traditional machine learning algorithms	0.76823	0.5624	1.5419
CNN	0.75695	0.5495	1.5942
RNN	0.75261	0.5689	1.5913
Attention Mechanism	0.81656	0.5974	1.4159
GNN	0.82459	0.6912	1.6962
total	3.91894	2.9694	7.8395

6. Conclusion

Traditional recommendation systems mostly use Convolutional neural network (CNN) to achieve simple image data collection for recommendation, but we are a GNN based recommendation system based on user habits. Our software can handle more complex relationships and images using GNN, and GNN can carry out end-to-end learning without manually designing features, which makes us more competitive in the market;

Traditional recommendation systems only recommend a class of resources that users often browse. The most obvious example is the TikTok recommendation system. The videos

recommended by it are easy to solidify into a class and difficult to broaden, which will only trap users in a single interest circle. Our recommendation system will predict the learning resources that users are interested in and make recommendations. If users browse, our algorithm will identify, then add such resources to the user interest library and recommend this type of resource. And our system will intelligently identify the resources that users are browsing and correct them, achieving the effect of pushing positive energy resources to users.

References

- [1] LEHMAN A, MILLER S J. A theoretical conversation about responses to information overload [J]. *Information*, 2020, 11 (8): 379-389.
- [2] ZHOU J, CUI G, ZHANG Z, et al. Graph neural networks a review of methods and applications [J]. *arXiv: 1812. 08434*, 2018.
- [3] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C] // *Advances in neural information processing systems*. Long Beach: Curran Associates, Inc, 2017: 6000-6010.
- [4] CHENG Z Y, DING Y, HE X N, et al. A³NCF: an adaptive aspect attention model for rating prediction [C] // *Proceedings of the 27th international joint conference on artificial intelligence*. Louisiana: AAAI, 2018: 3748-3754.
- [5] PEI W J, YANG J, SUN Z, et al. Interacting attention-gated recurrent networks for recommendation [C] // *Proceedings of the 2017 ACM on conference on information and knowledge management*. Singapore: ACM, 2017: 1459-14.