Matrix Completion Recommendation Algorithm Based on Attention Mechanism

Lin Wei[1], Zihao Qi[1], Lei Shi[1], Guangtao Zhao[3], Bo Li[4], Yufei Gao[1], Yongcai Tao[2]

[1] School of Cyber Science and Engineering, Zhengzhou University; Songshan lab, Zhengzhou, China
[2] School of Computer and Artificial Intelligence, Zhengzhou University, China
[3] Henan Province Electronic Planning Research Institute Co.LT, Zhengzhou, China
[4] Jinshui Branch of Zhengzhou Public Security Bureau, Zhengzhou, China

Abstract

Matrix completion, as a method of recommendation system, can well represent the user's rating relationship with the item. The traditional matrix completion algorithms rely on a lot of edge information and the model is too complex, and in a real recommendation system, there is usually not enough edge information available, which can lead to a very large recommendation error. Therefore, this paper proposes a matrix completion recommendation model based on attention mechanism without using edge information. The model only uses user-rating matrix, trains the matrix completion model through multi-layer graph convolution neural network, beyond this, uses graph attention mechanism to give different weights to different nodes, which can make full use of the information of import nodes. Experiments on public datasets have shown that our method is superior to the traditional algorithms related to matrix completion.

Keywords

recommendation system, graph neural network, matrix completion, graph convolutional neural network, attention mechanism

1. Introduction

With the rapid development of information technology, especially in the field of e-commerce and social media, the amount of information worldwide is exploding, and while we enjoy the satisfaction of convenient access to information, information overload makes it difficult for users to find content of interest to them in the overwhelming amount of network data. The recommendation system provides a very good solution to information overload[1]. The recommendation system is to combine user information (gender, age, region, etc.), item information (price, place of origin, etc.), and user behavior of past items (whether to buy, whether to collect, whether to click, etc.), using machine learning, deep learning and other technologies to build a user interest model, to provide users with accurate personalized recommendations. It is precisely because the recommendation system improves the efficiency of information distribution and information acquisition that the recommendation system has become the core technology of many Internet manufacturers.

Due to the high industrial value of the recommendation system, the recommendation system has received widespread attention from industry and academia in recent years. From the earliest collaborative filtering recommendations and content-based recommendations to today's deep learning-based recommendation systems, new technologies are constantly improving the performance of the recommendation system. Although the above methods have a certain degree of advancement, there are still certain problems, as follows:

1. Only focus on considering a single modeling user representation or item representation, and do not consider the impact of neighbor information on the user.

ISCIPT2022@7th International Conference on Computer and Information Processing Technology, August 5-7, 2022, Shenyang, China
EMAIL: weilin@zzu.edu.cn (Lin Wei); qizihaoness@163.com (Zihao Qi); shilei@zzu.edu.cn (Lei Shi); guangtaozhao@163.com (Guangtao Zhao); 75174586@qq.com (Bo Li); yfgao@zzu.edu.cn (Yufei Gao); ieyctao@zzu.edu.cn (Yongcai Tao);

2. Using too much edge information. In the absence of edge information in the dataset, it can lead to extremely errors.
3. The important information between entities is not fully mined, resulting in the delivery of important messages being easily lost.

Aiming at the above problems, this paper proposes a matrix completion recommendation model based on attention mechanism, we named it Att-MC. The model does not use edge information, only the user-rating matrix and the user relationship matrix, and the user's predictive rating of an item is composed of the neighbor's rating fusion of the item. When performing feature fusion, an attention mechanism is introduced to achieve adaptive matching of weights to different neighbors, thereby improving the accuracy of the model.

2. Related work

2.1. Traditional recommendation algorithms

Collaborative filtration technology[2] is the earliest recommended algorithm, first proposed in the 1990s. Collaborative filter recommendation technology is simply to use the similar interests and have the preferences of a group of common experience to recommend information that the user is interested in, collaborative filtering algorithm includes two categories: user-based collaborative filtering algorithm and content-based collaborative filtering algorithm. The idea of the user-based collaborative filtering algorithm is to find other users who are similar to the user and recommend items that other users interact with to the user. The idea of the content-based collaborative filtering algorithm is to find other items that are similar to the item that interacts with it and recommend it to that user. From the above, it can be found that the central idea of the collaborative filter recommendation algorithm is to calculate the similarity. However, due to the extreme scarcity of data, user information is very small, the user similarity obtained is likely to be zero, and when calculating the similarity of users or items, the computing resources consumed are also very large.

Feature-based recommendations use more edge information than collaborative filtering, and by training mathematical models to predict how users rating unexacted items, commonly used methods such as Probabilistic Matrix Factorization[3](PMF) and Singular Value Decomposition[4](SVD). The idea of PMF and SVD is to first establish an appropriate mathematical model of the historical interaction data of users and items, and then generate a recommendation list that meets the needs of users through the model, of which the more widely used recommendation is based on matrix decomposition.

Mixed recommendation is a way to retain the advantages of different recommendation techniques to avoid their shortcomings, different recommendation algorithms are integrated into the recommendation system, that is, mixed recommendation[5][6], mixed recommendation is mainly divided into 3, pre-fusion, post-fusion, medium fusion.

In recent years, due to the increasing diversity of user attributes and user-item behaviors, traditional recommendation techniques have been unable to meet the diverse needs of users.

2.2. Recommendation system based on graph neural network

In recent years, graph neural networks[7] have become the latest development direction for recommended systems. Recommendation systems based on graph neural networks[8] employ advanced graphical methods to model user preferences and intentions. Unlike other recommendation methods (including content-based filtering and collaborative filtering), graph-based neural network recommendation systems are built on graphs that explicitly or implicitly connect important objects, such as users, projects, and attributes. Graph neural networks are very suitable for use in recommendation systems, because most of the data on the recommendation system is non-Euclidean data, and the objects in the recommendation system include users, projects, attributes, and contexts, which are closely related to each other and influence each other through various relationships[9]. Second, graph neural networks can learn complex graph relationships, and many graph-based learning methods have been developed to learn specific types of relationships modeled by graphs and have proven to be very effective [10].

However, how to effectively disseminate information between users and items is a challenge based on graph neural network recommendation systems. In order to solve this problem, the GC-MC[11] model treats matrix completeness as a link prediction problem on the graph, the autoencoder generates the potential characteristics of the user and the item node by passing messages on the user-item binary diagram, and the GC-MC model combines external information with interactive information, effectively alleviating the performance bottleneck related to the cold start problem. The IGMC[12] model argues that in some extreme cases, there is no edge information for the recommendation system to use, in which case the effect of the model that relies on the edge information will become less pronounced, so the IGMC model proposes a model that does not make the inductive matrix of the edge information complete, and the IGMC model trains the graph neural network based on the pair of 1-hop subgraphs generated from the user-item scoring matrix, and maps these subgraphs to the corresponding ratings. The LightGCN[13] model has demonstrated through various ablation experiments that the two most common designs in GCN, feature transitions and nonlinear activations, contribute little to the performance of co-filtering, and worse, using feature transitions and nonlinear activations increases the difficulty of training and reduces recommended performance. Therefore, the LightGCN model uses only the most important component in the GCN, domain aggregation for collaborative filtering.

However, at present, many recommendation algorithms based on graph neural networks rely on a large amount of edge information, and once the data lacks edge information, it will lead to a sharp decline in recommendation effect. Therefore, this paper proposes a recommendation model that uses only the user-rating matrix, the model does not rely on edge information, and at the same time uses the attention mechanism to give different weights to different nodes, making full use of the information of important nodes.

3. Model introduction and algorithm description

3.1. Formulaic description

Our model aims to predict the user's rating of an item based on the user-rating matrix data and the user's relationship matrix. During the forecasting process, there is no reliance on edge information and long-term preference information of the user.

In this model, we make N_u represent the number of users, N_v represents the number of items, M (shape is $\mathrm{N}_u \times \mathrm{N}_v$) represents the rating matrix, M_{ij} represents the user i's rating of the item j, A (shape is $\mathrm{N}_u \times \mathrm{N}_u$) represents the user's relationship matrix, A_{ij} indicates whether the user i and the user j are connected, if the user i and the user j have both rated an item, it is 1, otherwise 0. In the model proposed in this paper, the rating matrix M and the adjacency matrix A are used to train in the multilayer graph convolutional neural network and the graph attention network, and the reconstructed rating matrix $\breve{\mathrm{M}}$ is obtained, and then the model is optimized by calculating the error between the predicted value and the label value.

3.2. Overarching framework

In Att-MC, we first compose the user-item bipartite graph information into a user-rating matrix, extract the domain information by using GCN, and generate an accurate project implicit vector. The resulting implicit vector of the item is then fed into the attention network, consider the information of important domain nodes to construct a more accurate vector representation and predict the user's rating of the item.

The overall framework of the model is shown in Figure 1, first, we construct the bipartite graph information into a user-rating matrix M, the shape of the user-rating matrix is $\mathrm{N}_u \times \mathrm{N}_v$, where N_u represents the number of users, N_v represents the number of items, according to the user-rating matrix to construct the adjacency matrix A, the construction steps are: If the user i and the user j have both rated the same item, then A_{ij} is 1, otherwise is 0. The user-rating matrix and adjacency matrix are then normalized according to (1). The standardized user-rating matrix and adjacency matrix are used to learn model parameters through two layers of GCN and one layer of GAT. Finally, the resulting implicit vector passes through the ReLU layer to obtain the final reconstruction matrix $\breve{\mathrm{M}}$,

$$\frac{\vec{h}_1}{sum(\vec{h}_1)} \tag{1}$$

where $\vec{h}_1$ represents the feature vector of a single node, $sum(\vec{h}_1)$ represents the sum of all node feature vectors.

We define the loss function as the error between the predicted value and the label value, and at the end of the model, we use the Adam algorithm to train the proposed model.

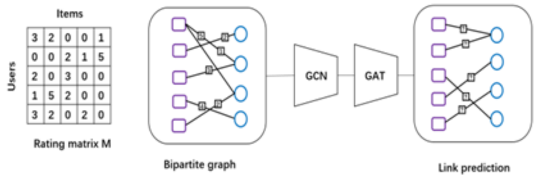

Figure 1: Workflow of the proposed method

3.3. GCN layer

GCN (Graph convolutional network) is a neural network architecture that manipulates graph data, and it is very powerful to generate useful feature representations of nodes in graph networks. However, using too many GCNs can cause the model to be overly smooth, therefore this model uses two layers of GCN. The definition of the first layer of GCN is shown in (2), due to the need to reconstruct the new user-rating matrix, therefore, the first layer of GCN does not consider the influence of the node itself on itself,

$$\mathrm{H}^{l+1} = \delta(\mathrm{A}\mathrm{H}^{l}\mathrm{W}^{l}) \tag{2}$$

where W^{l} represents the parameter matrix of the lth layer, H^{l} is the embedding matrix of graph nodes in the lth layer of convolution, $\delta(\cdot)$ is a nonlinear activation function.

The definition of the second layer GCN is shown in (3), since the first layer has eliminated the information of its own nodes, the second layer can adopt its own degree matrix to solve the self-passing problem,

$$\mathrm{H}^{l+1} = \delta(\widetilde{\mathrm{D}}^{-\frac{1}{2}}\widetilde{\mathrm{A}}\widetilde{\mathrm{D}}^{-\frac{1}{2}}\mathrm{H}^{l}\mathrm{W}^{l}) \tag{3}$$

where D is the embedding dimension, $\widetilde{\mathrm{A}}$ is the adjacency matrix of the graph with self-loop,$\widetilde{\mathrm{D}}^{ii} = \sum_j \widetilde{\mathrm{A}}_{ij}$.

3.4. GAT layer

GAT (Graph attention network) aggregates neighbor nodes through self-attention mechanism, and realizes adaptive matching of weights for different neighbors, thereby improving the accuracy of the model.

The formula for calculating the attention coefficient is as follows:

$$\begin{gathered}
\mathrm{h}_i^{l+1} = \delta\left(\sum_{j\epsilon N_i} a_{ij}\mathrm{W}^l\mathrm{h}_j^l\right) \\
a_{ij} = \frac{\exp\left(\mathrm{L}eaky\,\mathrm{ReLU}\left(\mathrm{a}^T\left[\mathrm{W}^l\mathrm{h}_i^l||\mathrm{W}^l\mathrm{h}_j^l\right]\right)\right)}{\sum_{k\epsilon N_i}\exp\left(\mathrm{LeakyReLU}\left(\mathrm{a}^T\left[\mathrm{W}^l\mathrm{h}_i^l||\mathrm{W}^l h_k^l\right]\right)\right)}
\end{gathered} \tag{4}$$

where a_{ij} is the propagation weight from node j to node i and N_i is the neighborhood set of node i, including i itself. As shown in (4), he attention mechanism is implemented via a fully-connected layer parameterized by a learnable vector a**,** followed by the $softmax$ function.

4. Experimental results and analysis

This section mainly introduces the datasets and evaluation indicators used by the model. The method of this paper is then comprehensively compared with other methods. Finally, a detailed analysis of the model is given.

4.1. Dataset description

In order to verify the accuracy of this model, the ML-100K and ML-1M in the public MovieLens dataset are selected for implementation, and the MovieLens dataset contains multiple user rating data on multiple movies, as well as movie metadata information and user attribute information, and Table 1 gives the basic information of the dataset.

Table 1
Basic information of recommendation system datasets

Dataset	Users	Items	Ratings	Density
ML-100K	943	1682	100000	0.0630
ML-1M	6040	3952	1000209	0.0419

4.2. Evaluation indicators

In order to evaluate the recommended performance of this model, the mean absolute error (MAE) and root mean square error (RMSE) are used as the evaluation indicators of the algorithm.

The average absolute error MAE represents the average of the absolute error between the predicted value and the true value, and the smaller the MAE value, the higher the recommended accuracy, as defined as follows:

$$M_{MAE} = \frac{\sum_{\mathrm{i,j}} |\hat{\mathrm{r}}_{\mathrm{ij}} - \mathrm{r}_{\mathrm{ij}}|}{\mathrm{n}} \tag{5}$$

where $\hat{\mathrm{r}}_{ij}$ represents the predicted rating, r_{ij} represents the true rating, n is the sum of rating.

RMSE represents the square root of the difference between the predicted value and the true value and the sum of the squares of the ratio n to the number of predictions. RMSE reflects the degree of dispersion of the sample, and the smaller the RMSE, the higher the recommended accuracy. The definitions are as follows:

$$R_{RMSE} = \sqrt{\frac{\sum_{i,j} (\hat{r}_{ij} - r_{ij})^2}{n}} \tag{6}$$

4.3. Analysis of results

To verify the effectiveness of the Att-MC, we compare it with several existing representative methods for matrix completion recommendation problems. They are matrix decomposition (MF) [14], Probability Matrix Decomposition (PMF) [15], Bias-Singular Value Decomposition with bias (Bias-SVD) [14], and singular value decomposition incorporating implicit feedback information (SVD++) [16]. Table 2 shows a comparison of experimental results. As can be seen from Table 2, the two evaluation errors of the proposed algorithm are better than other comparison algorithms, this is because our method can extract the social information in the user-rating graph very well, and at the same time, the use of the graph attention network can increase the weight of important neighbor node information, so that an accurate user-rating matrix can be generated.

Table 2
Error value comparison of different recommendation algorithms

Evaluation	MF	PMF	Bias-SVD	SVD++	Att-MC
RMSE	0.9662	0.9654	0.9446	0.9436	0.8925
MAE	0.7618	0.7609	0.7488	0.7493	0.7056

5. Conclusion

In this paper, a matrix completion recommendation algorithm based on attention mechanism is proposed, which does not require complex edge information and long-term user dependence on items to achieve better recommendation performance. However, at present, this model is only suitable for static graph networks, once the nodes are added to the graph, the entire network needs to be recalculated. Therefore, the next task is to carry out research on the recommendation algorithm of dynamic network incremental computation on the basis of this model, so that the model can adapt to dynamic graphs.

6. Acknowledgements

This work was supported in part by the National Key Technologies R&D Program (2020YFB1712401, 2018YFB1701400), the Key Project of Public Benefit in Henan Province of China (201300210500), the Key Research Projects of Universities in Henan Province of China (7A520015, 21B520018), the Fundamental Science Projects of Railway Police College (2020TJJBKY002), and the Key Scientific and Technological Research Projects in Henan Province of China (192102310216).

7. References

[1] Elciyar, K. . Technostress: Information Overload and Coping Strategies. 2021.
[2] Glauber, R. , and A. Loula . "Collaborative Filtering vs. Content-Based Filtering: differences and similarities." (2019).
[3] Salakhutdinov, R. , and A. Mnihdepartment . "DRAFT. A revised version will appear in NIPS 2007. Probabilistic Matrix Factorization." (2008).
[4] Kalman, Dan . "A Singularly Valuable Decomposition: The SVD of a Matrix." The College Mathematics Journal 27.1(1996):2-23.
[5] LMD Campos, et al. "Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks." International Journal of Approximate Reasoning 51.7(2010):785-799.
[6] Pazzani, M. J. . "A Framework for Collaborative, Content-Based and Demographic Filtering." Artificial Intelligence Review 13.5-6(1999):393-408.
[7] Gao, C. , et al. "Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions." (2021).
[8] Wang, S. , et al. "Graph Learning based Recommender Systems: A Review." (2021).
[9] Hu, Liang, et al. "Deep modeling of group preferences for group-based recommendation." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 28. No. 1. 2014.
[10] Wu, Zonghan, et al. "A comprehensive survey on graph neural networks." IEEE transactions on neural networks and learning systems 32.1 (2020): 4-24.
[11] Berg, Rianne van den, Thomas N. Kipf, and Max Welling. "Graph convolutional matrix completion." arXiv preprint arXiv:1706.02263 (2017).
[12] Zhang, Muhan, and Yixin Chen. "Inductive matrix completion based on graph neural networks." arXiv preprint arXiv:1904.12058 (2019).
[13] He, Xiangnan, et al. "Lightgcn: Simplifying and powering graph convolution network for recommendation." Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020.
[14] Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." Computer 42.8 (2009): 30-37.
[15] Zhang, Zhijun, and Hong Liu. "Application and research of improved probability matrix factorization techniques in collaborative filtering." International Journal of Control & Automation 7.8 (2014): 79-92.
[16] Jia, Yancheng, et al. "Users' brands preference based on SVD++ in recommender systems." 2014 ieee workshop on advanced research and technology in industry applications (wartia). IEEE, 2014.