Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 220 (2023) 456–461

Procedia Computer Science

www.elsevier.com/locate/procedia

The 14th International Conference on Ambient Systems, Networks and Technologies (ANT)
March 15-17, 2023, Leuven, Belgium

A Collaborative Filtering Movies Recommendation System based on Graph Neural Network

Redwane Nesmaoui[a,*], Mouad Louhichi[a], Mohamed Lazaar[a]

[a] *ENSIAS, Mohamed V University in Rabat, Morocco*

Abstract

The implementation of machine learning algorithms in marketing by organisations has been more beneficial in recent years. Overall, it has become a major contributor to a company's success and development in terms of growth and income since it helps to recommend the interesting product/service to the right individuals or groups without requiring them to go through a long complex procedure to receive an interesting item from a list of millions, in the other side Graph Neural Network is used widely in the recent machine learning applications including Recommender Systems. The purpose of this research is the evaluation of a LightGCN Movies Recommendation System, and its efficiency in modelling and building relationship between movies, by providing suggesting new/unknown items to the users that will like them, those recommendations will be based on representing Movies as a node and their ratings as edges of the graph, which will help to build a continuous representation of nodes and edges, this approach required the combination of a classification model to predict the existence of the relationship between movies and their features as genres, release year, etc, this approach will enable us to predict when no neighbourhoods information is known.

Keywords: Artificial intelligence; Machine learning; Graph Neural Network; Recommender Systems; Movie recommendation; Neural Networks

1. Introduction

Recommender Systems (RS) are growing pervasive and crucial to the success of mission-critical systems and applications across many industrial sectors, and they represent the success of the deployment of artificial intelligence systems in social, and commerce environments, as a kind of information filtering designed to offer information items (music, news, movies, music, web pages, books, etc.) that the user may find interesting [13].

Recommender Systems are used on online sales websites since the Internet has become an integral element of our age. They enable e-merchants to automatically promote items that are most likely to appeal to visitors and optimise their revenues. The presented product selection is then customised according to several factors in order to improve

* Corresponding author. Tel.: +49 1521-7266628
E-mail address: redwane.nesmaoui@um5r.ac.ma

10.1016/j.procs.2023.03.058

sales revenue. Amazon is well-known for its sophisticated recommendation engine, but the trend of online shopping is spreading and no longer exclusive to huge companies, it is vital to analyse the intricate interaction between consumers and goods, such as their social network and their competitive relationship [7].

Recent developments in graph neural networks give a solid foundation and potential for addressing the aforementioned problems in recommender systems. Graph neural networks use embedding [2] propagation to iteratively collect neighbourhood embeddings. By stacking the propagation layers, each node may access information about its high-order neighbours, as opposed to merely its first-order neighbours, as is the case with conventional approaches. GNN-based techniques are the current state-of-the-art approaches in recommender systems due to their benefits in handling structural data and exploring structural information.

To effectively implement graph neural networks into recommender systems, it is necessary to overcome many crucial problems. First, the data input of the recommender system should be created as a graph, with nodes representing items and edges denoting relationships. Second, for the given purpose, Components of the graph neural network should be constructed in an adaptable manner, including propagation and aggregation. Existing research has evaluated many methodologies with varying benefits and drawbacks. Third, the optimization of the GNN-based model [18], including the optimization objective, loss function, and data sampling must adhere to the task criteria. [10].

In this paper, we seek to present a thorough and exhaustive evaluation of the research effort, focusing on providing movies recommendation using graph neural networks, which don't need the whole rating matrix as input and may infer ratings for specific user-item combinations. Local graph pattern is the key that liberates us from utilising the content or the whole rating matrix. The rating matrix may be converted into a bipartite graph by adding an edge for each observed rating between the related person and item. Consequently, predicting unlabelled linkages in this bipartite graph is identical to predicting unlabelled ratings. This changes matrix completion into a problem involving link prediction, where graph patterns play a significant role in identifying the presence of links, in general, the goal is reducing the number of elements from millions, or thousands to tens which means the mean challenge will be the efficiency, feature extraction, accuracy, and then defending the relationships of the items.

2. Related Works

Movie Recommender Systems have been the subject of several papers, with researchers exploring the topic from a variety of perspectives and using a wide range of feature and classifier algorithm types, including graph convolution networks etc.

LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation has simplified the design of Graph Convolution Network (GCN) [1] to make it usable in recommender systems and called it LightGCN, where this model is based on only including neighbourhood aggregation for collaborative filtering, specially it learns user and item embeddings via linear propagation on the user-item interaction graph, and then uses a weighted sum of the embeddings learned across all layers to produce the final embedding.

In our previous research we used metrics similarity to recommend movies through difference approaches [13], which different than using LightGCN that used the relationship between the nodes (the processing is not based on graph processing which provide an easy way to do node-level, edge-level, and graph-level prediction tasks)

3. Methodology

To come up with a solution to this problem, we will examine the following different kinds of methods to determine their strengths and weaknesses, and then we will report our findings using the 1 million records MovieLens(1m) dataset and the precision-recall metric.

3.1. Graph Methods for Recommendation (LightGCN)

Graph Convolution Network (GCN) have just a while ago seen a surge in popularity across a variety of industries [3], from social networks and knowledge graphs to recommender systems and even the medical sciences. Advances in the field of graph analysis are made possible by the effectiveness of GNN in modelling the relationships between nodes in a graph. The purpose of this essay is to provide a primer on graph neural networks and to present two more

complex algorithms.
In our work we decided to use the LightGCN (Light Graph Convolution) as a graph method, which use the following formula for the propagation rule for item and user embedding between each layer [4][5][6].

- k^{th} layer user embedding

$$e_u^{(k+1)} = \sum_{i \in N_u} \frac{1}{\sqrt{|N_u|}\sqrt{N_i}} e_i^{(k)} \tag{1}$$

- k^{th} layer item embedding

$$e_i^{(k+1)} = \sum_{i \in N_u} \frac{1}{\sqrt{|N_i|}\sqrt{N_u}} e_u^{(k)} \tag{2}$$

Where:

- N_u is the set of items liked by the user u.
- N_i is the set of users liked i.

3.2. Layer Combination and Model Prediction

To train the LightGCN model the only parameter that we can vary is the first (0^{th} element) layer embedding e_u^0 and e_i^0.

After the combination of the embedding obtained at each layer of propagation, the final formulas are the following:

- user embedding

$$e_u = \sum_{k=0} \alpha_k e_u^{(k)} \tag{3}$$

- item embedding

$$e_i = \sum_{k=0} \alpha_k e_i^{(k)} \tag{4}$$

Where:

- α_k, the contribution of the k-th layer embedding to the final embedding is weighted by this hyper-parameter.

4. Architecture

This architecture is based on three main steeps:

1. Message: Each neighbour passes its current embedding to the central node.
2. Aggregation: Each message passed by the neighbours is aggregated to produce a single embedding by taking the sum the average or the max value.
3. Update: The embedding node take the previous layer and combines it with the neighbours aggregation, to get the current embedding of the node.

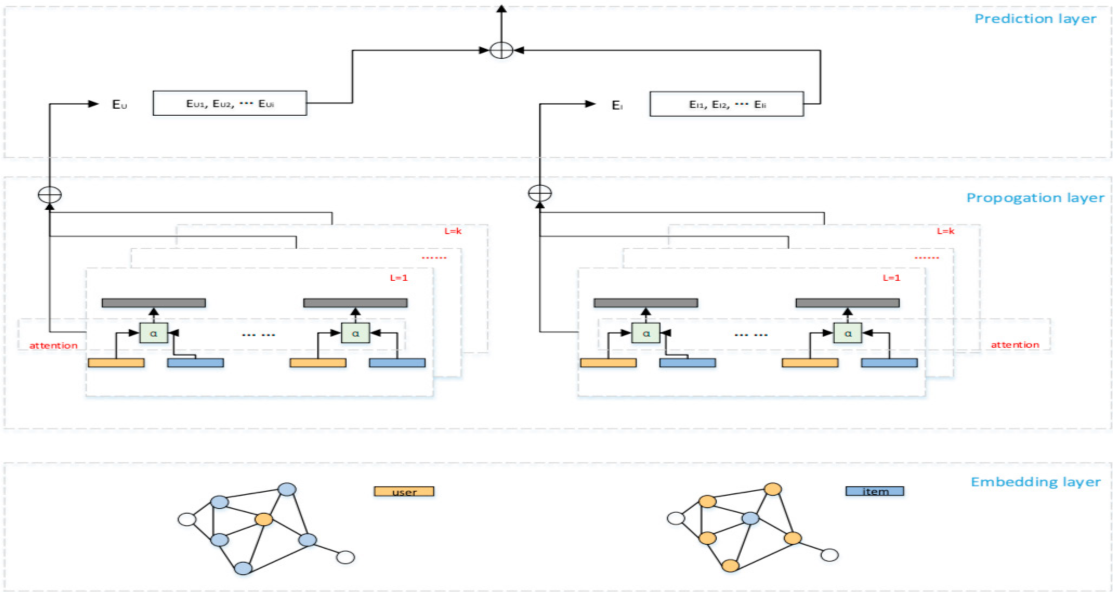


Fig. 1: Overview of LightGCN model architecture.

5. Experiments and Performance Evaluation

5.1. Dataset

To appraise the efficiency of our methods, we will operate with the MovieLens (1M) data set which consists of over a million movie ratings provided by 6000 users on 4000 movies.

The MovieLens dataset is represented as a 2 Dimensions NumPy array with the following format: $A_{i,j}$ (A_i act in place of the movies, and the values are centered on zero by subtracting the mean from the respective elements).

5.2. Adam Optimiser

We present Adam, a low-memory, first-order gradient-based approach to efficient stochastic optimization. Adam is an abbreviation for "adaptive moment estimation," which is how the approach gets its name. The method estimates first and second moments of gradients to calculate individual adaptive learning rates for distinct parameters [15].

5.3. Bayesian Personalized Ranking (BPR)

This method is allowing to rank the items for a chosen user, and recommend the top-K most likely items the he will like. This model's concept revolves on selecting positive item that the user engaged with as well as negative ones, and finally run a pairwise comparisons [9], and the loss of this method is giving as the following formula:

$$L_{BPR} = -\sum_{u=1}^{M} \sum_{i \in N_u} \sum_{j \notin N_u} ln \quad \sigma\left(\hat{y}_{ui} - \hat{y}_{uj}\right) + \lambda \|E^{(0)}\|^{2} \tag{5}$$

5.4. Collaborative Filtering

Collaborative Filtering (CF) is mainly recommending items that the user may like, based on similarity with other users that have the same taste, for example a user has a music playlist and we want to suggest other tracks to his playlist, so this recommendation will based on other users that have nearly similar taste (liking and listening to similar music that this user is listening to). [16].

5.5. Performance Metric

The evaluation of our experimental requires the usage of Recall and Precision.

$$Recall = \frac{T_P}{T_P + F_P} \tag{6}$$

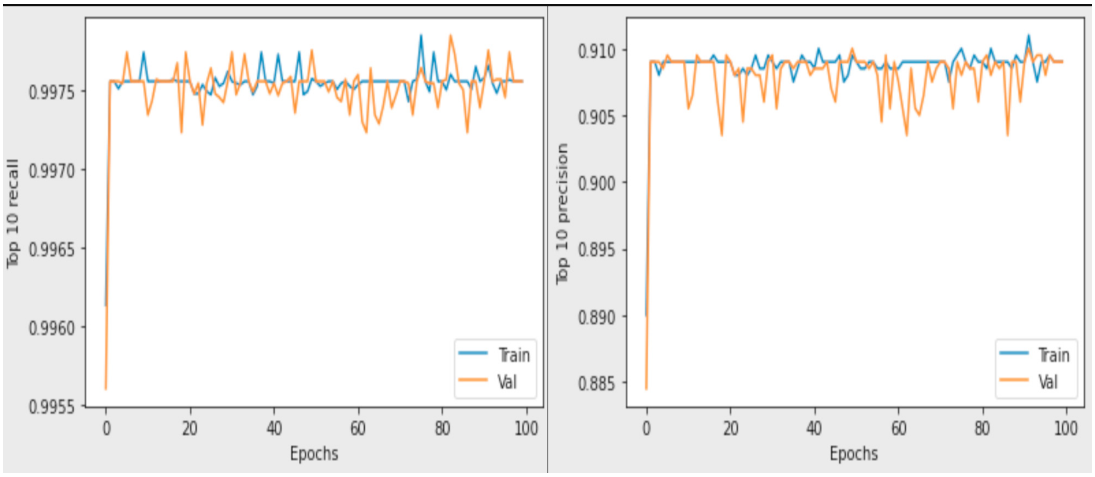
$$Precision = \frac{T_P}{T_P + F_N} \tag{7}$$

Where:

- T_P = True Positive
- F_P = False Positive
- F_N = False Negative

6. Results Analysis

The results of the application of our model are shown in the following figures, and those figures show that for top-10 our precision = 0.9101 and recall= 0.9925. The results show that by increasing of the epochs does not show any degradation of performance, in other way round it shows an increase of performance and getting improved results in terms of precision, and recall, which explains the importance of having more items to check the similarity of users behaviour to provide better recommendations.



(a) The recall of our module in different epochs (b) The Precision of our module in different epochs

7. Conclusion

Recommender systems are a power way to provide the right services/products to the rights target user, even if the user is not showing a clear request to those services/products.

In a general those systems are helping to improve the user's experience with the provided services/products, by analysing his historical and try to generate recommendation that the user could like and interact with it in a positive way, which make the digital marketing mechanisms more efficient, because it makes specifying the right user for the marketing campaign more dynamic and it helps the companies and also the client to find what they are looking for.

But all that can not be achieved with the usage of deep learning, which has many benefits, specially in the aspects

of making the recognition and the classification performing much better, because the impact of deep learning for example is better than that of the classic methods, as well as that deep learning is more relevant, and can provide good approaches to solve many issues as recommendations that machine learning approaches can not, specially because of that recommender systems have emerged as an essential technology for many online platforms attempting to forecast whether or not a user would engage with an item.
Collaborative Filtering (CF) models, for example have significant results because they are based on learning the users behaviour based on the item that he liked before, but LightGCN method showed enhanced results which based on the embedding representation of users and the items they like, because the graph connections are based on aggregating feature information from neighbours, and extract possible items that the user may like, but in case of dividing users into multiple subgraphs can affect the efficiency of the recommendations because of the loss of information between the subgraphs (Groups of users), so based on that we suggested our approach which based on passing messages between the graphs, like that we can link users and the items which are part of other subgraphs acquire the final node embedding by incorporating numerous embedding propagation layers that encode higher-order connectivity links.

References

[1] He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M., 2020. LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20. Association for Computing Machinery, New York, NY, USA, pp. 639–648. https://doi.org/10.1145/3397271.3401063

[2] Wang, S., Hu, L., Wang, Y., He, X., Sheng, Q.Z., Orgun, M.A., Cao, L., Ricci, F., Yu, P.S., 2021. Graph Learning based Recommender Systems: A Review. https://doi.org/10.48550/arXiv.2105.06339

[3] Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P., 2019. A Comprehensive Survey on Graph Neural Networks.

[4] Covington, P., Adams, J., Sargin, E., 2016. Deep Neural Networks for YouTube Recommendations, in: Proceedings of the 10th ACM Conference on Recommender Systems, RecSys '16. Association for Computing Machinery, New York, NY, USA, pp. 191–198. https://doi.org/10.1145/2959100.2959190

[5] Rendle, S., Gantner, Z., Freudenthaler, C., Schmidt-Thieme, L., 2011. Fast context-aware recommendations with factorization machines, in: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information - SIGIR '11. Presented at the the 34th international ACM SIGIR conference, ACM Press, Beijing, China, p. 635. https://doi.org/10.1145/2009916.2010002

[6] Ying, R., He, R., Chen, K., Eksombatchai, P., Hamilton, W.L., Leskovec, J., 2018. Graph Convolutional Neural Networks for Web-Scale Recommender Systems, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining. pp. 974–983. https://doi.org/10.1145/3219819.3219890

[7] Ahmed, F., Cui, Y., Fu, Y., Chen, W., 2021. A Graph Neural Network Approach for Product Relationship Prediction. https://doi.org/10.48550/arXiv.2105.05881

[8] Cheng, H.-T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., Anderson, G., Corrado, G., Chai, W., Ispir, M., Anil, R., Haque, Z., Hong, L., Jain, V., Liu, X., Shah, H., 2016. Wide Deep Learning for Recommender Systems. https://doi.org/10.48550/arXiv.1606.07792

[9] Zhang, Q., Ren, F., 2021. Potential Preference Based Bayesian Personalized Ranking for One-Class Collaborative Filtering.

[10] Gao, C., Zheng, Y., Li, N., Li, Yinfeng, Qin, Y., Piao, J., Quan, Y., Chang, J., Jin, D., He, X., Li, Yong, 2021. Graph Neural Networks for Recommender Systems: Challenges, Methods, and Directions.

[11] Koren, Y., Bell, R., Volinsky, C., 2009. Matrix Factorization Techniques for Recommender Systems. Computer 42, 30–37. https://doi.org/10.1109/MC.2009.263

[12] MovieLens 1M Dataset [WWW Document], 2015. . GroupLens. URL https://grouplens.org/datasets/movielens/1m/ (accessed 10.28.22).

[13] Nesmaoui, R., Louhichi, M., Lazaar, M., 2022. A Hybrid Machine Learning Method for Movies Recommendation, in: Lazaar, M., Duvallet, C., Touhafi, A., Al Achhab, M. (Eds.), Proceedings of the 5th International Conference on Big Data and Internet of Things, Lecture Notes in Networks and Systems. Springer International Publishing, Cham, pp. 517–528. https://doi.org/10.1007/978-3-031-07969-6_39

[14] Rendle, S., 2010. Factorization Machines, in: 2010 IEEE International Conference on Data Mining. Presented at the 2010 IEEE International Conference on Data Mining, pp. 995–1000. https://doi.org/10.1109/ICDM.2010.127

[15] Ogundokun, R., Maskeliunas, R., Misra, S., Damaševičius, R., 2022. Improved CNN Based on Batch Normalization and Adam Optimizer. pp. 593–604. https://doi.org/10.1007/978-3-031-10548-7_43

[16] Chen, L., Wu, L., Hong, R., Zhang, K., Wang, M., 2020. Revisiting Graph based Collaborative Filtering: A Linear Residual Graph Convolutional Network Approach. https://doi.org/10.48550/arXiv.2001.10167

[17] Afoudi, Y., Lazaar, M., Al Achhab, M., 2021. Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network. Simulation Modelling Practice and Theory 113, 102375. https://doi.org/10.1016/j.simpat.2021.102375

[18] Wang, X., He, X., Wang, M., Feng, F., Chua, T.-S., 2019. Neural Graph Collaborative Filtering, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'19. Association for Computing Machinery, New York, NY, USA, pp. 165–174. https://doi.org/10.1145/3331184.3331267

[19] Zhang, M., Chen, Y., 2020. Inductive Matrix Completion Based on Graph Neural Networks. https://doi.org/10.48550/arXiv.1904.12058

[20] Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M., 2021. Graph Neural Networks: A Review of Methods and Applications.