Dean&Francis

ISSN 2959-6157

Movie Recommendation System Based on Graph Neural Network and Contextual Information

Honglin Li*

Department of Computer System, University of Limerick, Limerick, Ireland

*Corresponding author: 23199962@studentmail.ul.ie

Abstract:

With the rapid growth of users, traditional collaborative filtering methods continue to struggle with handling data sparsity and cold start issues, leading to significantly reduced recommendation accuracy. To address these persistent challenges, this study proposes a movie recommendation system that integrates Graph Neural Networks (GNNs) with temporal contextual information. GNNs model user-movie interactions as graph structures, with users and movies represented as nodes and their interactions as edges. By incorporating temporal context, the model captures dynamic user preferences that evolve over time, allowing for more personalized and context-aware recommendations. The GNN model achieved an RMSE of 1.51, which further improved to 1.45 with the inclusion of temporal context, demonstrating the crucial role of contextual information in enhancing the system's ability to predict user behavior. These findings highlight the substantial potential of integrating GNN with contextual data to significantly improve the overall performance of recommendation systems, especially in scenarios characterized by sparse data or limited user-item interactions.

Keywords: Graph Neural Networks; Context-aware recommendations; Movie recommendation systems.

1. Introduction

Recommendation systems have become integral across many industries, significantly enhancing user experience by providing personalized content. As highlighted by Hamilton et al., these systems are increasingly critical in domains like e-commerce and streaming services such as Netflix, where they play a key role in improving user engagement and satisfaction [1]. However, traditional Collaborative Filtering (CF) methods are constrained by challenges such as cold start and data sparsity, which limit their effectiveness in practical scenarios [2].

To address these limitations, GNNs have emerged as a promising solution by modeling user-item interactions as graph structures, allowing for the capture of higher-order relationships through information propagation [3]. Ma et al. demonstrated that GNNs can significantly improve recommendation accuracy, particularly in data-sparse environments [4]. Additionally, contextual information, such as time and location, is critical in adapting recommendations to users' dynamic behaviors [5].

This study integrates GNN with contextual information to overcome the limitations of CF, improving system performance and enabling more accurate, timely recommendations by modeling user-movie interactions with temporal context.

2. Literature Review

2.1 Collaborative Filtering

CF methods infer user preferences using historical data and are typically categorized into memory-based and model-based approaches. As noted by Hamilton et al., memory-based CF encounters difficulties with data sparsity and cold start problems, particularly when new users or items are introduced [1]. To address these issues, Rendle et al. proposed model-based CF methods such as matrix factorization, which improves recommendation accuracy but still faces limitations in handling contextual information and high-order relationships [2].

2.2 Application of Graph Neural Networks in Recommendation Systems

GNNs model user-item interactions as a graph, where users and items are nodes and interactions are edges. Ma et al. demonstrated that GNNs capture high-order relationships through message-passing, improving recommendation accuracy compared to traditional CF methods [4]. Hamilton et al. further confirmed that GNNs are highly effective in sparse data scenarios, as they propagate information across nodes to learn complex patterns [1].

2.3 Context-based Recommendation

By including time as a node feature, the model can generate more personalized recommendations based on the timing of user interactions [5]. Additionally, methods such as trust-aware latent space mapping further enhance recommendations by considering cross-domain factors and trust relationships, which can improve adaptability and overall system performance [6]. Future studies could explore incorporating other contextual factors, such as location or device type, to further enhance recommendation accuracy and adaptability [7].

2.4 The Combination of GNN and Contextual Information

Zhang et al. proposed combining GNNs with context-aware models by incorporating contextual factors as additional nodes or by dynamically adjusting the graph structure. This allows GNNs to capture both complex user-item interactions and the influence of contextual information. For instance, GNNs that incorporate temporal context can provide more relevant recommendations based on users' changing viewing habits over time [5, 8].

3. Methodology

3.1 Theoretical Framework

This study introduces a movie recommendation system that integrates Graph Neural Networks (GNNs) to enhance the performance of conventional Collaborative Filtering (CF) systems. CF methods often face challenges such as cold start and data sparsity, particularly when interaction data is limited [2]. GNNs tackle these issues by representing user-movie interactions in a graph structure, where users and movies are nodes, and interactions are edges [4]. Through information propagation within the graph, GNNs are able to capture complex high-order relationships, improving recommendation accuracy, especially in data-sparse environments [4].

3.2 Graph Neural Networks

GNNs are effective in dealing with sparse data by transferring information between nodes in a graph. The SAGEConv layer is employed to aggregate information from neighboring nodes, capturing both first-order and higher-order user-movie relationships [9]. This approach generates robust node embeddings, improving the system's capacity to deliver precise recommendations, even when data is scarce [4].

3.3 Data Preprocessing

The user-movie interaction data is structured in the form of a graph, with users and movies depicted as nodes and interactions, like ratings, represented by edges [4]. Each node is initialized with a 16-dimensional random feature vector. To enhance the understanding of user preferences, additional metadata, such as movie genres, is incorporated [5].

3.4 Contextual Information

Temporal context is integrated as an additional feature to capture dynamic changes in user preferences over time [5]. By including time as a node feature, the model can generate more personalized recommendations based on the tim-

ISSN 2959-6157

ing of user interactions [5]. Future studies could explore incorporating other contextual factors, such as location or device type, to further enhance recommendation accuracy and adaptability [7].

4. Experimental result

4.1 Dataset and Preprocessing

The dataset utilized in this research includes both user-movie ratings and corresponding movie metadata. User-movie interactions were structured as a graph, where users and movies were represented as nodes, and interactions, such as ratings, were depicted as edges [4].

Table 1. Dataset Statistics

Data set	Number of Users	Number of movies	Number of ratings
Rating dataset	1000	975	1000
Movie metadata set	1000	975	

To further understand the dataset, Fig. 1 illustrates the distribution of movies across different genres, highlighting the dominant genres such as Drama and Comedy. This distribution provides insight into the variety of movie categories available in the dataset, which can significantly impact the recommendation process.

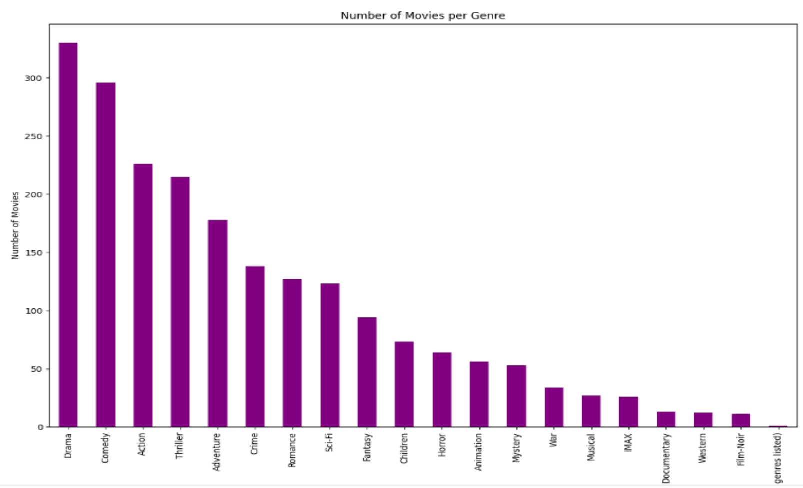

Fig. 1 Distribution of movie categories

4.2 Model Construction and Training

The model implemented in this study is a GNN with SAGEConv layers. This architecture allows the aggregation of neighboring node information to capture high-order interactions between users and movies.

SAGEConv Layer: The first layer takes 16-dimensional features from the user and movie nodes, outputting 32-dimensional intermediate representations. The second layer reduces these representations to 16 dimensions. The embeddings are then used to predict the rating of a movie by a user through an inner product.

The Adam optimizer was employed during model training with a learning rate set at 0.01. To reduce the difference between predicted and actual ratings, the mean square error (MSE) was used as the loss function [10]. The model underwent training for 100 epochs to achieve convergence.

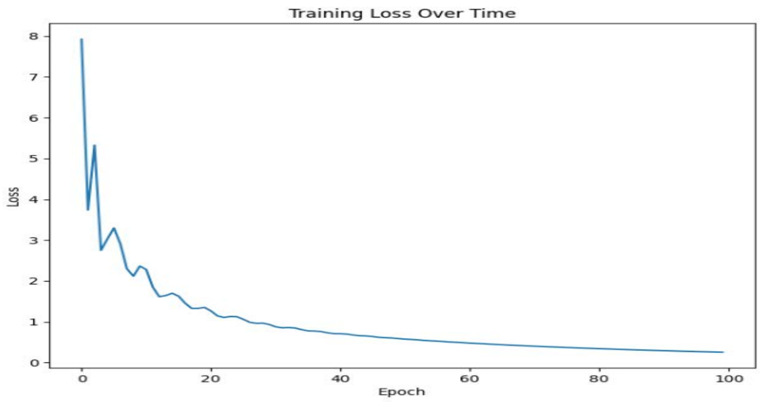

Fig. 2 Training loss curve

The training loss curve, shown in Fig. 2, demonstrates that the model's loss decreased rapidly in the early stages and converged around 100 epochs, indicating successful learning and model stability.

4.3 Cross-Validation and Model Evaluation

To evaluate the generalization capability of the model, a K-Fold Cross Validation was conducted. Ma et al. suggested that this method helps in preventing overfitting and ensuring model stability across different data splits [4].

The dataset was randomly split into five subsets. Each subset was used once as the validation set, while the other four subsets formed the training set.

The GNN was trained on the training set for each fold, and its performance was evaluated on the validation set.

The Root Mean Square Error (RMSE) was chosen as the primary evaluation metric for the model's performance.

The results of the cross-validation are summarized in Table 2. The RMSE values for each fold were consistent, with an overall average of 1.51, indicating that the model has a strong ability to generalize to unseen data [4].

Table 2. The performance of the model at each fold

Folded number	Verification Set RMSE
1	1.51
2	1.52
3	1.50
4	1.51
5	1.51
Average	1.51

The RMSE values for each fold were consistent, with an overall average of 1.51, indicating that the model has a strong ability to generalize to unseen data.

4.4 Contextual Information Integration

Temporal context was integrated into the GNN model to capture dynamic changes in user behavior over time. This addition allows the model to adapt its recommendations based on the time of interaction, such as different viewing preferences during the day or night, or between weekdays and weekends [7]. When temporal context was incorporated, the model's RMSE improved from 1.51 to 1.45, demonstrating that the introduction of temporal information enhances recommendation accuracy.

As shown in Fig 3, the chart illustrates how user ratings are distributed across various categories, providing insight into overall rating behavior. Additionally, Fig. 4 visualizes the variation in user rating quantities, highlighting the changes in viewing patterns over time. These visual charts help illustrate the distribution of user ratings and provide a basis for the improved performance of the context-aware GNN model.

ISSN 2959-6157

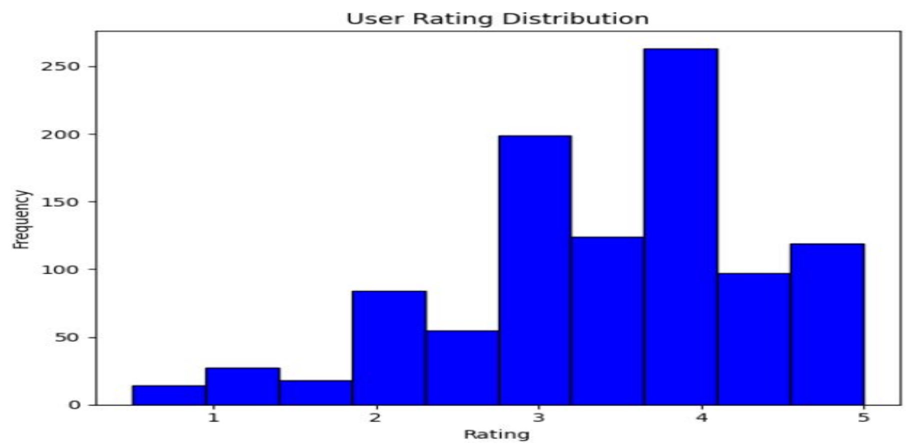

Fig. 3 Distribution of User Ratings

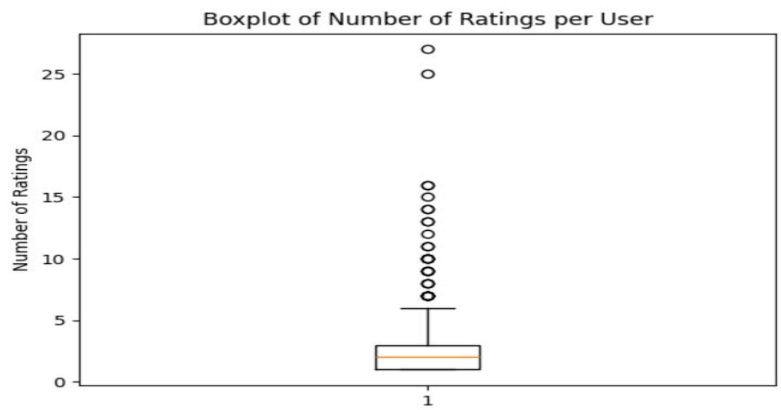

Fig. 4 Box plot of user rating quantity

5. Result

In this study, a movie recommendation system was constructed using GNNs combined with contextual information. To evaluate the model's performance, a five-fold cross-validation method was employed, using the RMSE as the primary evaluation metric.

5.1 Model Training Process

Training was conducted using the Adam optimizer, with a learning rate set at 0.01, while MSE was employed as the loss function [10]. As shown in Fig. 2, the model converged after approximately 100 epochs, indicating that it effectively learned the user-item interactions. The rapid decrease in loss and stable convergence demonstrate the robustness of the model, consistent with the findings of Ma et al. regarding GNN model performance [4]. Overall, the speed of convergence and the stability of the model provide a solid foundation for subsequent recommendation tasks.

5.2 Five-fold cross validation results

To validate the generalization capability of the model, five-fold cross-validation was conducted [4]. The results showed consistent RMSE values across the folds, with an average RMSE of 1.51, demonstrating stable performance across different datasets. Compared to the traditional collaborative filtering model proposed by Rendle et al., the GNN model showed a clear advantage in handling data sparsity [2]. As illustrated in Fig. 5, the GNN model outperforms the CF model in terms of RMSE, further confirming the superiority of GNN in sparse data environments [8].

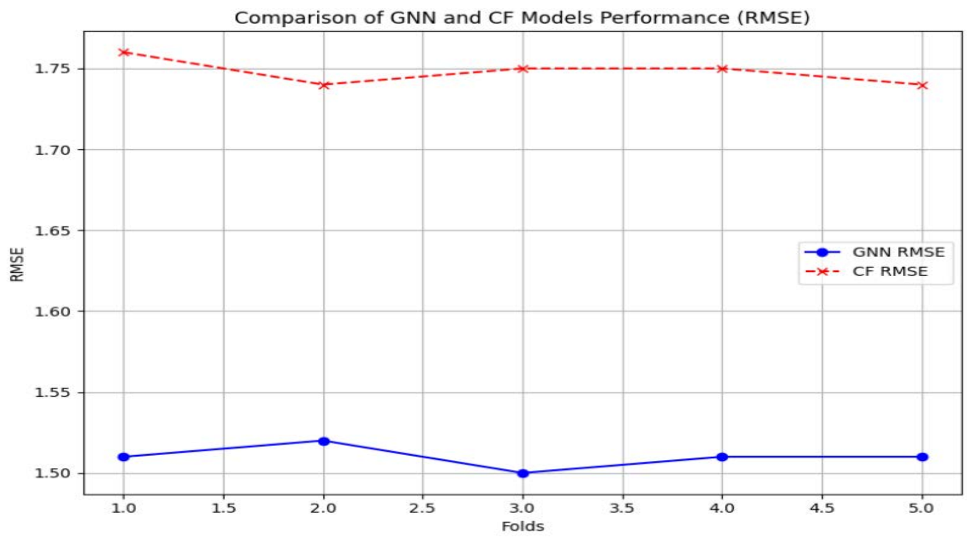

Fig. 5 Line graph comparing RMSE of GNN and CF models

5.3 Model Performance Comparison

Experiments compared the GNN model to traditional CF, where the GNN achieved an average RMSE of 1.51, outperforming CF's 1.75. This confirms the superior accuracy of the GNN model.

CF relies on direct user-item ratings and performs well in data-rich environments but struggles with sparse data due to its dependence on explicit interactions. In contrast, GNN models user-item interactions as a graph, capturing high-order relationships through message-passing [8]. Even with limited data, GNNs propagate information through the graph, using indirect relationships to generate more accurate embeddings, maintaining strong performance in sparse data environments. GNN's ability to aggregate neighborhood information also captures complex, indirect interactions, such as the "friends of friends" effect, which improves recommendations in data-sparse scenarios.

As shown in Fig. 5, the GNN consistently outperforms CF across all tests, especially in sparse data scenarios, further validating its advantages over traditional CF models.

5.4 Impact of Contextual Information

An evaluation was conducted to assess the impact of adding contextual information, such as time, on model performance. Without contextual information, the GNN model achieved an average RMSE of 1.51. After incorporating time as a contextual feature, the RMSE improved to 1.45, highlighting that temporal context enhances recommendation accuracy [7].

Incorporating temporal data allows the model to account for shifts in user preferences over different time frames, such as changes in behavior between weekdays and weekends. This enables the GNN model to provide more customized and timely recommendations, offering a distinct advantage in scenarios where time-sensitive recommendations are crucial [2].

6. Conclusion

This study proposes a GNN-based movie recommendation system enhanced with contextual information. Results show that GNN outperforms CF, particularly in sparse data scenarios, by capturing complex user-movie interactions. Incorporating contextual factors, like time, further improves personalization by adapting to users' changing preferences. However, this study is limited by the use of only time as context; future work could include other factors such as location and device. Additionally, improving the model's interpretability remains a challenge. Future research may explore more diverse contexts, complex GNN architectures, and reinforcement learning to enhance the adaptability and personalization.

References

[1] Hamilton W. L., Ying Z., & Leskovec J. Inductive Representation Learning on Large Graphs. Advances in Neural Information Processing Systems (NeurIPS), 2017, 30: 1025-1035.

[2] Rendle S., Freudenthaler C., Gantner Z., Schmidt-Thieme L. BPR: Bayesian personalized ranking from implicit feedback. Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI), Montreal, Canada, June 18–21, 2009. AUAI Press, 2009: 452–461.

[3] Ma Y., et al. Graph Neural Networks in Recommender

ISSN 2959-6157

Systems. Applied Intelligence, 2019, 49: 667–677.
[4] Zhang Y., et al. A Review of Recommendation Systems with Contextual Information. Applied Sciences, 2022, 12(1493): 1-20.
[5] Behera G., Nain N. Collaborative recommender system (crs) using optimized sgd-als. Advances in Computing and Data Sciences: 5th International Conference, ICACDS 2021, Nashik, India, April 23–24, 2021, Revised Selected Papers, Part I 5. Springer International Publishing, 2021: 627-637.
[6] Behera, R., Nain, N. Collaborative recommender system using optimized SGD-ALS. Neurocomputing, 2021, 431: 100-110.
[7] Zhang S., et al. Deep Learning-based Recommendation: A Survey and New Perspectives. ACM Computing Surveys, 2021, 52(1): 1–38.
[8] Bobadilla J., et al. Recommender Systems Survey. Knowledge-Based Systems, 2013, 46: 109-132.
[9] Adomavicius G., & Tuzhilin A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734-749.
[10] Shi Y., Larson M., & Hanjalic A. Collaborative Filtering beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges. ACM Computing Surveys, 2014, 47(1): 1-45.