Proceedings of CONF-MLA 2025 Symposium: Intelligent Systems and Automation: AI Models, IoT, and Robotic Algorithms
DOI: 10.54254/2755-2721/2025.LD24513

Interaction Behavior Prediction Method for Movie Recommendation Systems Based on Neural Collaborative Filtering

Shilin Wang

Wuhan Institute of Technology, Wuhan, China
2107767932@qq.com

Abstract. As internet technologies develop at a fast pace and information grows in an explosive way, movie recommendation systems are playing a more and more important role in easing the issue of information overload and making user experience better. Conventional collaborative filtering and content-based recommendation approaches run into various challenges when applied in real-life scenarios. These challenges include the cold start issue, the sparsity of related data, and the insufficient ability to build models for nonlinear interaction relationships. In this research, a new method for predicting movie interaction behaviors is put forward, and this method is based on Neural Collaborative Filtering. It makes good use of the advantage that deep learning has in building models for high-order nonlinear interactions. The research adopts the MovieLens dataset, which is open to the public, and builds a binary classification task targeted at the interactions between users and movies. The study learns dense vector representations by means of embedding layers, and it also uses multi-layer perceptrons to build models for the deep interactions of various features. To make the model more robust, the research introduces Dropout and Batch Normalization mechanisms into the model design. Hit Ratio@10 is chosen as the key metric to evaluate the model's performance, and a series of experiments are carried out to test the model. The experimental results show that the model proposed in this study works extremely well in predicting the preferences of users. The hit ratio of the model reaches 0.74, which means it performs much better than all traditional recommendation methods. At the same time, the proposed model also shows a high level of stability in all the experimental tests.

Keywords: Movie Recommendation, Deep Learning, Neural Collaborative Filtering, Embedding

1. Introduction

In the current digital age, internet technologies are developing at a rapid speed and data is increasing in an explosive manner. This trend has brought a huge amount of information from fields like news, music and entertainment into people's everyday lives. Such a wealth of information makes it much easier for people to access and communicate with information, yet it also leads to a severe problem of information overload [1]. When users are faced with too much irrelevant content, they often feel anxious and confused when they filter information and make relevant decisions. This kind of negative feeling has a bad influence on both their mental health and work efficiency. Traditional information

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2025.LD24513

retrieval and filtering technologies can reduce the impact of information overload to a certain degree. But they often cannot satisfy users' personalized needs that are dynamic and uncertain, and they are likely to bring about the Matthew effect. Against this background, recommendation systems have come into being as a smart connection between users and information, and they have quickly become a core technology in the digital economy.

Recommendation systems have an especially important role to play in the film industry. As the number of film and television works keeps going up, audiences often feel confused because they have too many viewing choices to make. Movie recommendation systems make an analysis of users' watching behaviors, their interest preferences and rating data, and then provide them with smart and personalized content recommendations. These systems not only make users' viewing experience better and raise their level of engagement and loyalty to relevant platforms, but also help content producers figure out market demand in an accurate way. This enables producers to carry out targeted marketing activities and boost box office earnings, and in this way, it promotes the healthy development of the film industry [2].

In the past few years, recommendation systems have been widely used in e-commerce and content distribution across the world. For instance, Amazon started to use item-based collaborative filtering as early as 1998. This technology allows the platform to give accurate product recommendations according to users' browsing and purchase history, and it updates recommendation lists in a dynamic way based on users' new behaviors. Domestic platforms including Taobao, Meituan, Douban and Tencent Video also make use of users' rating data, browsing records and geolocation information to build effective and precise recommendation frameworks. These platforms have achieved remarkable success in improving user experience and lifting the overall operating results of the platforms.

There are several main types of traditional recommendation methods, such as item-based collaborative filtering [3], user-based collaborative filtering [4], content-based recommendation and hybrid methods [5]. These methods have achieved some positive results in practical applications, but they also have obvious limitations. Item-based collaborative filtering recommends products to users according to the similarity between different items. It has the advantages of strong real-time performance and good interpretability, but it has trouble dealing with the cold start problem and can only provide recommendations with limited diversity. User-based collaborative filtering finds out users who have similar interests and then generates recommendation content for them, which can catch the changes in users' preferences in a dynamic way. However, this method still has to face the problems of data sparsity and cold start. Content-based recommendation methods rely on the attributes of items, such as genre, cast and director, to recommend similar items to users. Though this method works well in solving the cold start problem and is easy to interpret, it often fails to provide diverse recommendations and has difficulty in building models for users' complex interests. Hybrid recommendation systems combine different recommendation strategies to overcome the shortcomings of single methods, which helps improve recommendation accuracy and model robustness. But this kind of system also makes the whole system structure more complex, increases computational costs and reduces the interpretability of recommendation results.

Traditional recommendation methods still have their value in some specific application scenarios. But as users' needs become more complex and diverse, their weaknesses in shallow feature extraction and linear modeling have become more and more obvious. It is necessary to make technological breakthroughs to further improve the quality of recommendations. In recent times, the fast development of deep learning [6] has brought new vitality to recommendation systems through embedding technologies based on deep neural networks. These deep learning models can automatically learn dense low-dimensional representation features for both users and items. They can discover the complex patterns hidden behind users' behaviors and capture the nonlinear and high-order interaction relationships between different factors, which greatly improves the personalization

DOI: 10.54254/2755-2721/2025.LD24513

and accuracy of recommendations. Besides, deep learning technologies support the integration of multimodal data including text, images and sequence data. This makes it possible to build models and make predictions for users' dynamic interests, a capability that traditional recommendation methods are far from having.

Taking the above context into consideration, this paper does in-depth research on a movie recommendation method that is based on deep neural network embeddings. The research aims to improve the overall performance of movie recommendations by using advanced representation learning and interest modeling technologies. This research work offers theoretical support and practical reference for improving user experience and promoting the intelligent development of the film industry.

2. Previous works

In recent years, recommendation system technologies have developed at a fast pace, which has given rise to a variety of mainstream application approaches in the field. Traditional recommendation methods mainly consist of user-based collaborative filtering, item-based collaborative filtering, content-based recommendation methods, and hybrid recommendation systems that blend different strategies together.

User-based collaborative filtering aims to recommend items to a certain target user by first identifying other users who have similar interest preferences with this target user. This method is straightforward in logic and easy to put into practice, but it has high computational complexity when applied in large-scale systems and is also highly sensitive to the cold start problem and data sparsity issues in such systems. Item-based collaborative filtering is designed to recommend items that are similar to those the user has rated before in the system. This method delivers more stable performance and is suitable for recommendation systems with abundant historical user behavior data, yet it still has to face the challenges brought by data sparsity and the modeling of users' dynamic preferences over time. Content-based methods make use of inherent item features, such as genre, cast and director for movies, to model user preferences, and then recommend items with similar feature characteristics to users. This method can work in a really effective way even when the available user data is quite limited, and its logic for giving recommendations is easy to understand and explain clearly. Yet this method has a clear shortcoming: the recommendation results it produces are always in a narrow range, and the content of these results does not have enough different types and diverse elements in it. Hybrid recommendation methods are designed to put several different recommendation strategies together, and typical examples of these strategies include collaborative filtering and content-based recommendation methods. The main purpose of combining these different strategies is to reduce and make up for the various weak points and drawbacks that each single recommendation method has on its own. These hybrid methods can make the accuracy of recommendations much better and also enhance the overall robustness of the whole recommendation system. But it is important to note that such improvements are not obtained for free; they require the system to have a more complex structural design and also need more computational resources to support the whole running process.

In the past few years, recommendation models which are built on the basis of deep learning technology have gained a great deal of attention from the research community and the relevant industry fields that focus on this area. When we compare these deep learning models with the traditional recommendation methods that have been used for a long time, we can find that deep learning models have much stronger abilities in modeling nonlinear relationships. What's more, these models can dig deep into the collected user-item interaction data and find out the complex hidden relationships among various features from the data. By doing this, they can make the overall working effect of recommendation systems get a remarkable improvement in many aspects. Multi-layer perceptrons, which are often called MLPs for short, are constructed by stacking multiple fully

DOI: 10.54254/2755-2721/2025.LD24513

connected neural network layers one by one. Their main function is to model the high-order nonlinear interactions between different features, and this ability enables them to catch and understand the various and different preference trends that different users have in a very effective way. Convolutional Neural Networks, or CNNs, have the special ability to pick out local feature patterns from the data that records users' various behaviors. This unique ability makes CNNs a very useful tool when people need to analyze the small changes and shifts that happen in users' short-term interests and preferences. Recurrent Neural Networks (RNNs) and their improved variant models like LSTM and GRU have their own advantages too: they can recognize and capture the temporal dependencies that exist in the sequential data of users' behaviors. Because of this, they are especially fit and suitable for carrying out the sequential modeling work in the design and running of recommendation systems.

Among all the recommendation methods that take deep learning as their core foundation, Neural Collaborative Filtering, known as NCF in short, has brought a new change to the field. It uses neural network structures to take the place of the traditional matrix factorization approach that was widely used before, and in the specific task of predicting the interactions between users and items, it has shown extremely strong and outstanding working performance. NCF adds nonlinear transformation layers into its own model structure, and the main aim of adding these layers is to model the complicated relational connections between users and various items more accurately. This design of the model breaks through the original limitations that are caused by the linear inner-product calculation in traditional recommendation methods, and at the same time, it can catch and collect much richer and more detailed interaction signals between users and items from the relevant data. Besides, NCF supports the integration of various additional feature data, such as user attribute information and contextual behavior data, making the model highly scalable in practical application. Based on various common evaluation metrics in the field, NCF often outperforms traditional methods in terms of recommendation accuracy and recall rate, and it is especially suitable for the fine-grained prediction of user behavior in specific movie recommendation scenarios. However, this model has relatively high training overhead in the model training stage and its interpretability is quite limited, and these existing problems are all aspects that warrant further in-depth investigation and research.

3. Dataset and preprocessing

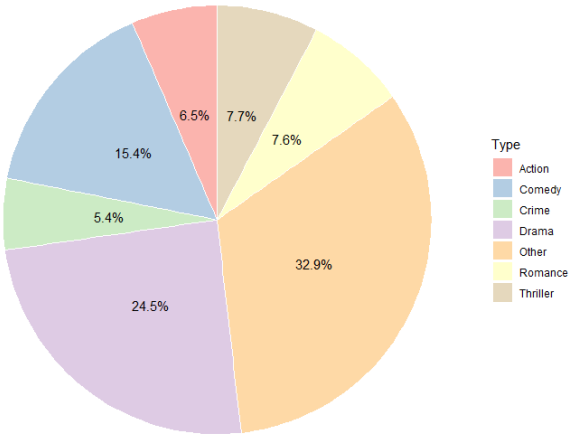


Figure 1. Movie genre distribution

This study makes use of the MovieLens dataset, an open-source dataset that has gained wide recognition in relevant research fields [7]. This dataset is officially released by the GroupLens research group, and it holds a large amount of information about the ratings given by users as well as detailed metadata related to various movies. It is a commonly adopted dataset in the research area of

DOI: 10.54254/2755-2721/2025.LD24513

recommendation systems. Specifically, the dataset covers 27,278 individual movies, and these movies are categorized into 20 different genres as shown in Figure 1. In addition, 138,493 users have participated in rating, and the total number of rating records in the dataset reaches 20,000,263, which is presented in Figure 2.

In the preprocessing stage of this research, the original and unprocessed rating data was first loaded into the dedicated research system we used. After that, we standardized the timestamp fields in the raw rating data and converted all of them into a unified standard datetime format. This specific data processing step is set up for a clear purpose: it makes the processed data fully suitable for the time-series ordering analysis and the user behavior modeling work that need to be done in the follow-up part of the research. Since the original dataset is of a very large scale, we adopted a random sampling method to make the whole research process more efficient and optimized. In this sampling, we randomly selected 30 percent of all the users in the original data, and we kept all the rating records that are linked to these randomly sampled users completely intact without any deletion or modification. After we finished this sampling operation, we finally formed the research subset that was used in the later study, and this subset contains 41,547 users as well as 6,027,314 corresponding rating records related to these users. This kind of data processing method has two important effects: on one hand, it makes sure that the behavioral diversity of the original dataset is well maintained in the selected research subset, and on the other hand, it can effectively control the amount of resource consumption in the subsequent research and computation process of the whole study.

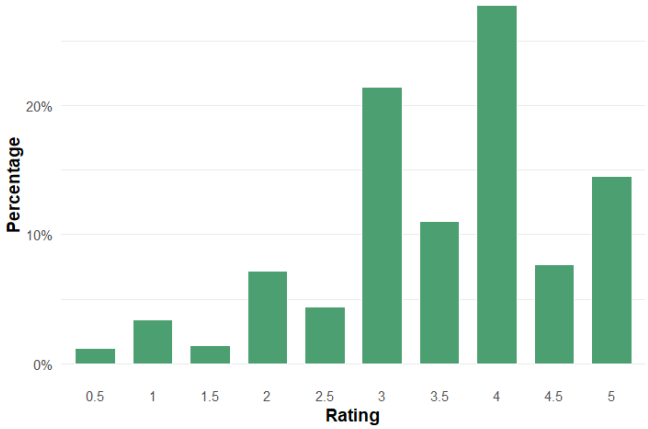


Figure 2. Rating distribution

To meet the actual need of predicting users' future behaviors in real life, this research adopted a train-test split strategy based on time sequence. First, we sorted all the rating records of each single user according to the time when they were created. We set the latest rating record of each user as the test sample, and all the rating records made before this latest one were assigned to the training dataset. Different from the random split methods used in some studies, this time-sequenced strategy can reflect the real patterns of users' behaviors much more accurately. It also gets rid of the possible risks of data leakage and ensures that the whole process of model evaluation is carried out in a strict and rigorous way. In this research, we only kept the core data features such as user ID, movie ID and the specific rating scores given by users. All the other redundant data fields were removed, and this step was taken to reduce the dimensionality of the whole dataset and cut down the interference caused by useless data noise. Considering that most recommendation systems used in real practice cannot get clear and explicit rating data from users, and they only rely on the observable interaction behaviors of users, we converted the original specific rating values into binary implicit feedback labels. A label of 1 was used to show that a user had had an interaction with a certain movie, while a label of 0 meant that there was no such interaction between the user and the movie. This conversion of data labels not

DOI: 10.54254/2755-2721/2025.LD24513

only makes the work of model construction much simpler, but also is in line with users' actual online behaviors like clicking a movie link or playing a movie video.

To improve the model's ability to tell the positive samples and negative samples apart clearly, we introduced negative sampling technology in the experiment. For each valid positive pair that stands for the interaction between a user and a movie, we randomly selected four movies that the user had never rated as negative samples, and these negative samples were used to form non-interaction pairs with the corresponding user. This negative sampling method not only expanded the overall scale of the training dataset, but also eased the serious problem of data sparsity in the original data. All the samples generated through this process were structured with three key elements: user ID, movie ID and the corresponding interaction label. This data structure was specially designed to fit the training requirements of supervised learning models. In the end, we organized the whole complete dataset into a batch-compatible structure that is fully adapted to common deep learning frameworks. This optimized data structure realized efficient data loading in the model training stage and made the iterative optimization of the model run smoothly. In this way, the model could reach its optimal performance level while the consumption of computational resources was kept to the minimum.

4. Model and results

Table 1. Model

	Layer Name	Type	#Params
0	user_embedding	Embedding	1 M
1	item_embedding	Embedding	1 M
2	fc1	Linear	2 K
3	fc2	Linear	8 K
4	fc3	Linear	2 K
5	fc4	Linear	528
6	output	Linear	17
7	dropout	Dropout	0
8	batch_norm1	BatchNorm1d	256
9	batch_norm2	BatchNorm1d	128
10	batch_norm3	BatchNorm1d	64

This study implemented an NCF-based movie recommendation model via the PyTorch Lightning framework to boost training efficiency and modularity. Its architecture (Table 1) consists of user and movie embedding layers, multiple fully connected layers, Dropout and Batch Normalization, with Sigmoid activation for binary classification of user-movie interactions. Given user and movie IDs, the model extracts and concatenates their dense embeddings, then feeds them into four fully connected layers equipped with ReLU activation, batch normalization and dropout to strengthen nonlinearity, stability and generalization. The final output of the Sigmoid function narrows down all prediction results to the interval of [0,1], and this value stands for the probability of interaction between a user and an item. The model was trained by using the Binary Cross Entropy Loss as the loss function, and the Adam optimizer was applied to optimize the training process with the learning rate set at 0.001. During the training, the batch size was fixed at 512, and the whole training process was set to run for a maximum of 5 training epochs. To reduce the operational overhead of the training system, the logging and checkpointing functions were turned off, and a single 2080Ti GPU was used to speed up the entire model training process significantly.

For the evaluation of the model's performance, the Hit Ratio@10 metric, which is widely used in relevant research, was chosen as the evaluation standard. This metric is used to check whether the target item is listed among the top-10 recommended items given by the model. For each interaction sample in the test dataset, 99 negative items that the user had never interacted with were randomly

DOI: 10.54254/2755-2721/2025.LD24513

selected. These 99 negative items and the single target item were combined to form a candidate set with 100 items in total. Then the model made predictions for this candidate set, and the top-10 items with the highest prediction scores were selected for evaluation. A valid hit was recorded if the target item was found in these top-10 recommended items. After completing the evaluation for all test samples, the overall hit ratio of the model was calculated. This calculation method provides a practical and effective way to evaluate how well the model performs in real-world recommendation scenarios. The NCF model achieved a Hit Ratio@10 score of 0.74 on the MovieLens dataset, which means the model has an average success rate of 74% in recommending the target movies to users. This result fully proves that the NCF model has high prediction accuracy and strong robustness in practical application. When compared with traditional recommendation methods such as similarity-based collaborative filtering and matrix factorization, the neural network structure of the NCF model can model the nonlinear interaction relationships between users and items in a much more efficient way. It also eases the common problems of data sparsity and cold start that trouble many recommendation systems. Besides, the NCF model kept a stable performance level when applied to different groups of users: active users who had plenty of interaction data with the system got higher hit ratios because their feature embeddings contained more useful information, while inactive users who had few interaction records still received reliable recommendation results from the model. This verifies that the NCF model has excellent generalization ability and a good adaptability to solve the cold start problem in recommendation systems.

5. Discussion and conclusion

This research built a movie recommendation system based on deep learning with the use of Neural Collaborative Filtering, and it did tests on this system with the MovieLens dataset to check its effectiveness. The results from the experiments proved that the system worked really well in the Top-N recommendation tasks, and it reached a Hit Ratio@10 of 0.74, which clearly shows that it can predict what users are interested in with great accuracy. The model used embeddings, MLP architectures, Dropout and Batch Normalization in its design, and these elements made it have much better generalization ability and robustness in practical use. The results of this study make it clear that deep learning can play a key role in describing the complex nonlinear interactions in recommendation systems, and it also offers a practical method to make movie recommendations that are personalized and precise for different users.

But even with these good results, the study still has some obvious limitations. First of all, the research only made use of the MovieLens dataset in all the tests. Although this dataset can stand for common user behavior to some extent, its data is organized in a clear and regular way, and the user behavior it records is relatively focused, so it can not fully show all the different situations that happen in real life. Second, the model still needs to rely on the data of users' historical interactions to make recommendations, and it is not able to solve the cold start problem completely in the actual application process. Besides, the deep learning model is like a "black box" when it runs, it has no clear interpretability, and it may also have a bias toward users who are more active on the platform or the movie items that are more popular among the public.

For future research, there are several good research directions to explore, and these directions can help make the model's expressiveness, generalization ability, diversity and personalization better in a further way. Recommendation systems are not staying the same, they are developing from the early collaborative filtering and shallow models to more intelligent deep learning architectures step by step. New and developing AI technologies bring new chances for the improvement of recommendation systems, including large language models (LLMs) [8], graph neural networks (GNNs) [9], multimodal fusion and transfer learning [10]. LLMs are trained with autoregressive or autoencoding methods, and they have achieved very good results in understanding and generating text content. If

DOI: 10.54254/2755-2721/2025.LD24513

we combine LLMs with recommendation systems, the systems can understand unstructured data such as users' movie reviews and movie descriptions much better. What's more, LLMs can also let recommendation systems realize conversational recommendation through natural language communication with users, which can make the systems more interpretable and let users be more engaged in the recommendation process. GNNs are good at dealing with graph-structured data, and they can catch the complex connections between different elements like users, movie items, movie genres, actors and directors. By collecting and sorting information from different types of heterogeneous graphs, GNNs can provide embeddings that are rich in content and have a clear structure. Multimodal techniques put different kinds of data together, such as text, video, images and audio, and integrate them into a single unified learning framework. For example, if we combine visual data and textual data in the model, it can better catch the styles and themes of different movies, and this makes the recommendations match with users' aesthetic preferences more closely. Last but not least, transfer learning makes it possible to move knowledge from one domain or task to another, and this feature is especially useful in situations like new-user recommendation, new-item recommendation or cross-platform recommendation. Using pretrained models or doing fine-tuning on existing models can cut down the need for a large amount of data and shorten the time needed for model development. If we combine transfer learning with domain adaptation, self-supervised learning or meta-learning, it can also make the effect of cross-domain recommendation much better.

To put it in short, this study designs an interaction prediction model for movie recommendation that is based on NCF, and it uses a public dataset to test and prove that this model is effective in practice. The experimental results show that deep learning can greatly improve the ability to model users' different preferences in recommendation systems. The research work in the future will focus on exploring more abundant features and more advanced model architectures, and these efforts will help improve the accuracy and generalization ability of movie recommendations, so that the system can be better applied in real-world personalized recommendation scenarios.

References

[1] X. Zhang, Ding ,Xiaoyan, and L. and Ma, "The influences of information overload and social overload on intention to switch in social media," Behaviour & Information Technology, vol. 41, no. 2, pp. 228–241, Jan. 2022, doi: 10.1080/0144929X.2020.1800820.

[2] H. Yuan and A. A. Hernandez, "User Cold Start Problem in Recommendation Systems: A Systematic Review," IEEE Access, vol. 11, pp. 136958–136977, 2023, doi: 10.1109/ACCESS.2023.3338705.

[3] M. Gao, Z. Wu, and F. Jiang, "Userrank for item-based collaborative filtering recommendation," Information Processing Letters, vol. 111, no. 9, pp. 440–446, Apr. 2011, doi: 10.1016/j.ipl.2011.02.003.

[4] L. H. Son, "HU-FCF: A hybrid user-based fuzzy collaborative filtering method in Recommender Systems," Expert Systems with Applications, vol. 41, no. 15, pp. 6861–6870, Nov. 2014, doi: 10.1016/j.eswa.2014.05.001.

[5] Z. Huang, C. Yu, J. Ni, H. Liu, C. Zeng, and Y. Tang, "An Efficient Hybrid Recommendation Model With Deep Neural Networks," IEEE Access, vol. 7, pp. 137900–137912, 2019, doi: 10.1109/ACCESS.2019.2929789.

[6] H. Zhou, F. Xiong, and H. Chen, "A Comprehensive Survey of Recommender Systems Based on Deep Learning," Applied Sciences, vol. 13, no. 20, Art. no. 20, Jan. 2023, doi: 10.3390/app132011378.

[7] F. M. Harper and J. A. Konstan, "The MovieLens Datasets: History and Context," ACM Trans. Interact. Intell. Syst., vol. 5, no. 4, p. 19:1-19:19, Dec. 2015, doi: 10.1145/2827872.

[8] ChangYupeng et al., "A Survey on Evaluation of Large Language Models," ACM Transactions on Intelligent Systems and Technology, Mar. 2024, doi: 10.1145/3641289.

[9] G. Corso, H. Stark, S. Jegelka, T. Jaakkola, and R. Barzilay, "Graph neural networks," Nat Rev Methods Primers, vol. 4, no. 1, pp. 1–13, Mar. 2024, doi: 10.1038/s43586-024-00294-7.

[10] M. Gholizade, H. Soltanizadeh, M. Rahmanimanesh, and S. S. Sana, "A review of recent advances and strategies in transfer learning," Int J Syst Assur Eng Manag, vol. 16, no. 3, pp. 1123–1162, Mar. 2025, doi: 10.1007/s13198-024-02684-2.