Movie Recommendation System Based on Word Embedding

Jiale Tian

Huazhong University of Science and Technology, Luoyu Road 1037, Wuhan, China

3295182429@qq.com

Abstract. Amidst the swift progression of Internet and big data technologies, recommendation systems have emerged as crucial conduits linking users to products across various digital platforms. This study delves into the deployment of neural collaborative filtering within the realm of movie recommendation systems, with the objective of constructing a system of high precision. Utilizing the MovieLens dataset, this investigation applies one-hot encoding and embedding techniques within the PyTorch-Lightning framework to effectively model user behaviors and predict cinematic preferences. The neural collaborative filtering (NCF) model leverages deep learning to extract latent features of users and items, exhibiting a notable enhancement in performance compared to traditional collaborative filtering approaches. Empirical results indicate that the NCF model attains an accuracy of 86% on the validation dataset, underscoring its efficacy in extensive recommendation scenarios.

Keywords: Collaborative filtering, Deep learning, Movie recommendation systems, Embedding.

1. Introduction

In the background of the rapid development of the Internet and big data technology, recommendation systems, serving as an important bridge connecting users with products, have been widely applied in various online platforms. From e-commerce to streaming platforms, recommendation systems not only enhance user experience but also greatly boost product sales. Especially in the film industry, the role of film recommendation systems is more prominent. With the continuous increase in global film output and the diversification of viewing needs, it is becoming increasingly difficult for users to find movies that suit their taste among the vast amount of information. Film recommendation systems provide personalized recommendations from a vast movie library to help users find suitable viewing choices within a short time, thus enriching their viewing experience.

However, building an efficient movie recommendation system is not an easy task. As a complex cultural product, movie recommendation involves understanding and analyzing various factors such as user preferences, emotions, and social background. Each user's taste in movies varies greatly, and accurately capturing and predicting these preferences is a major challenge in system design. Additionally, as the scale of the data grows, the movie recommendation system needs to handle massive amounts of user behavior data and movie information, requiring the system to have extremely high computing efficiency and data processing capabilities. At the same time, the diversity and novelty of movie recommendations are also a challenge, as the system not only needs to recommend movies that the user may like, but also needs to avoid over-recommending known or similar content to maintain the user's long-term interest.

Traditional movie recommendation systems primarily rely on collaborative filtering, content-based recommendation, and hybrid recommendation methods. However, these methods have shown some limitations when faced with large-scale data and diverse user needs. For example, collaborative filtering methods rely on a user's historical behavior, which can lead to the "cold start" problem, where the system cannot provide effective recommendations for new users or new movies. Content-based recommendation may not fully uncover a user's latent interests, leading to limited and one-dimensional recommendations. Additionally, traditional methods often struggle to handle complex user preferences and multidimensional data relationships, resulting in insufficient accuracy and diversity in recommendations.

In this context, deep learning-based recommendation systems have emerged, showing significant advantages. Deep learning methods can automatically extract features from massive data and capture deep-level patterns of user preferences, overcoming many limitations of traditional methods. For example, using deep neural networks can better handle non-linear and high-dimensional data relationships, thereby improving the accuracy of recommendations. In addition, deep learning can also be combined with natural language processing, computer vision, etc. to use the text description, posters, and trailers of movies, etc. for recommendation, providing more diverse and personalized recommendation content. These advantages have made deep learning-based recommendation systems a hot research and application topic, providing new ideas for the innovative development of movie recommendation systems.

This study developed a deep movie recommendation system based on word embedding methods, which used one-hot encoding to process user viewing history data and applied deep learning algorithms to model the text data to reveal the user's behavior patterns and provide precise movie recommendations. The goal of this study was to improve the accuracy and diversity of the recommendation system to provide users with more personalized viewing experiences.

2. Previous works

In the history of recommendation systems, collaborative filtering algorithms have long dominated, becoming one of the classic implementation methods. Collaborative filtering algorithms can be broadly divided into two categories: user-based collaborative filtering and item-based collaborative filtering. In addition, the introduction of hybrid recommendation systems and deep learning models has further enriched the implementation methods of recommendation systems.

User-based collaborative filtering predicts a user's preference for a particular item by calculating the similarity between users. The basic assumption of this method is that users with similar interests usually have similar preferences for similar items. By analyzing the historical behavior data of users, the system can recommend other items that other users with similar interests have liked to the current user. The advantage of this method is that it can discover the hidden similarity between users, especially in scenarios where the time sensitivity is strong and the user's personalized needs are not obvious. However, as the number of users increases, the cost of calculating user similarity increases significantly, affecting the system's scalability. In addition, user-based collaborative filtering faces the "cold start" problem, where the system is unable to make accurate recommendations for new users or users with limited historical behavior data.

Based on item collaborative filtering, the similarity between items is calculated by analyzing users' behavior data on items, and items similar to the users' preferred items are recommended accordingly. Compared with user-based collaborative filtering, item-based collaborative filtering's recommendation results are more intuitive and can provide real-time recommendations based on users' current interests. This method performs particularly well in scenarios where users' individual interests are obvious and item updates are less frequent. However, as the number of items increases, the cost of calculating item similarity also increases. Moreover, if the similarity matrix is not updated regularly, new items cannot be effectively recommended, limiting the system's adaptability in dynamic environments.

To compensate for the shortcomings of single recommendation algorithms, hybrid recommendation systems have gradually become a research hotspot. Hybrid recommendation systems combine the advantages of various recommendation algorithms to improve recommendation accuracy and diversity. For example, combining user-based collaborative filtering with item-based collaborative filtering can consider both the similarity between users and the similarity between items, providing more comprehensive recommendations. This method alleviates the "cold start" problem to some extent and performs well in scenarios with diverse needs. However, hybrid recommendation systems are complex, and the training and maintenance costs of the model increase accordingly, especially in application scenarios with large-scale data processing and high real-time requirements, where the system's efficiency and scalability still face challenges.

With the development of deep learning technology, researchers have started to integrate it into recommendation systems in an attempt to overcome the limitations of traditional methods. Deep learning models significantly enhance the performance of recommendation systems through automated feature extraction and recognition of complex patterns. Wide & Deep model is a representative example, which combines the strengths of linear models and deep neural networks [1]. This model can memorize frequent feature combinations while capturing deep-seated user behavior patterns. To some extent, this approach overcomes the limitations of traditional collaborative filtering methods that rely on manual feature design, providing more precise modeling of user preferences. However, the "Wide" part of the Wide & Deep model still relies on manually designed feature engineering, limiting its automation and generalization capabilities. In order to address these limitations, the DeepFM model was proposed, replacing the linear regression model in the "Wide" part with Factorization Machines (FM) [2]. FM can automatically learn second-order feature interactions, reducing reliance on manual feature engineering and improving recommendation accuracy. Nevertheless, DeepFM has limited innovation as its feature interactions are restricted to second-order only and cannot fully capture complex relationships between higher-order features. This limitation somewhat restricts performance improvement for this model. The introduction of Deep Interest Network (DIN) further advances the application of deep learning in recommendation systems [3]. The DIN model introduces activation units to generate personalized embedding vectors based on users' dynamic interests more accurately capturing changes in user interest. This significantly enhances recommendation precision but still falls short in fully utilizing users' historical behavioral temporal characteristics as well as addressing cold start issues. To tackle these challenges, EGES (Enhanced Graph Embedding System) was proposed, which generates graph embeddings by introducing supplementary information and uses KNN methods to generate embeddings for new products effectively solving cold start problems. EGES also incorporates attention mechanisms to handle different attribute importance making it exhibit strong adaptability in practical engineering applications [4].

Despite the fact that deep learning models have shown great potential in recommendation systems, they usually require a large amount of data and computational resources, and also face challenges such as feature engineering and cold start problems. In this context, deep recommendation systems based on word embedding as an innovative solution have gradually attracted the attention of researchers. Word embedding technology maps text data to a low-dimensional vector space, so that semantically similar words are closer in space. This technology not only captures the semantic information of text, but also automatically extracts potential features from movie descriptions, user reviews and other text information through deep learning models. Compared with traditional feature engineering, word embedding can better handle high-dimensional and non-linear data relationships, significantly improving the accuracy and diversity of recommendation systems.

3. Dataset and Preprocessing

In the process of building a recommendation system, the selection and processing of the dataset are crucial steps that directly affect the system's accuracy and effectiveness. Therefore, this paper conducts experiments and analysis using the MovieLens dataset, which is widely used in the field of recommendation system research [5]. The MovieLens dataset was created by the GroupLens project team at the Department of Computer Science and Engineering, University of Minnesota, USA, and is one of the standard datasets for recommendation system research, widely recognized and used.

The MovieLens dataset contains several key features that provide rich data support for building an effective recommendation system. Specifically, the dataset contains several important files, including the genome_scores file, which records the movie ID, tag ID, and relevance, used to quantify the degree of association between a movie and a specific tag; the genome_tags file contains the tag and its corresponding tag ID, which are used to identify specific attributes of a movie and help the system better understand the features of the movie content. In addition, the link file provides the ID correspondence between movie IDs and external databases such as IMDb and TMDb, allowing the MovieLens dataset to be integrated with other movie databases and expanding the data's application scope. The movie file records the basic information of the movie, such as the movie title and release year, which are crucial for building user profiles and recommendation logic; the rating file contains the user ratings for movies, which are the direct expression of user preferences and one of the core data sets for recommendation systems; the tag file stores the tags that users have assigned to movies, which usually reflect the user's subjective evaluation and viewing experience.

To ensure the quality of the data and improve the performance of the recommendation system, we conducted systematic data preprocessing and cleaning. First, considering the need to speed up processing and reduce data volume, we randomly selected 30% of the user data for analysis. Next, we filtered and selected the relevant data for the selected users, retaining only their data to ensure the study's focus. In the further processing stage, we grouped the data by user ID and sorted each user's rating data by timestamp. By assigning the most recent rating record of each user a rank of 1, this sorting method can effectively distinguish the user's current interests from their historical preferences, thereby better capturing the user's current interest dynamics. To train and test the recommendation system, we separated each user's most recent rating record as a test set, while the remaining rating data was used as the training set. This division method not only simulates real-world usage scenarios but also effectively evaluates the system's predictive performance. To ensure that the training data set contains balanced positive and negative samples, we adopted a sample ratio of 1:4 for positive and negative samples, which helps improve the model's generalization ability and avoids the model becoming biased towards a particular type of sample during training. Finally, to simplify the data structure and improve the model's training efficiency, we retained only the userId, movieId, and rating columns, removing other unnecessary fields. This operation not only reduces the data's dimension but also effectively improves the model's training and computing efficiency, providing a more concise data input for the subsequent recommendation algorithm.

4. Model and results

In this study, we adopted the Neural Collaborative Filtering (NCF) model based on the PyTorch-Lightning framework, aiming to predict the matching degree between users and items in a recommendation system and achieve personalized recommendations [6]. The NCF model learns the potential latent features of users and items through their embeddings (embedding) and predicts the degree of preference for a specific item by a user using a multi-layer fully connected network. The implementation process starts from the embedding layer. Specifically, the ID of users and items are mapped to low-dimensional dense vectors, which can effectively capture the key features of users and items and provide a foundation for subsequent feature learning. The importance of this step lies in that it converts the high-dimensional sparse data into low-dimensional dense representations, allowing the model to learn patterns and regularities in the data more effectively. Then, these embedding vectors are concatenated along the feature dimension to form a composite representation vector. This concatenated representation vector contains multiple potential features of users and items, laying a foundation for subsequent feature learning. The concatenated vector is then passed through two fully connected layers in sequence, which are responsible for further combination and transformation of features to learn more complex and deep-level nonlinear feature relationships. After each fully connected layer, the model uses the ReLU activation function ($\mathrm{R(x)} = \max\{0, \mathrm{x}\}$) for processing. The introduction of the ReLU activation function can effectively introduce nonlinear features and enhance the model's ability to capture complex patterns. In the final output layer of the model, we use a single output node and generate the user's preference probability for an item using the Sigmoid activation function $(S(x) = 1/(e^x + 1))$. The Sigmoid function outputs a range between 0 and 1, making the output results easily interpretable as the user's preference probability, which is convenient for subsequent recommendation decisions. Through this design, the NCF model can fully exploit and utilize the embedding information of users and items, combine multi-level nonlinear feature combinations, and significantly improve the accuracy and personalization of recommendations. Especially through the deep feature learning of multi-layer fully connected networks, the model can capture complex relationships that traditional collaborative filtering methods cannot detect, thereby achieving more precise recommendations.

To verify the effectiveness of the model, we conducted experiments using a training set and validation set containing 5985767 data sets. During the training process, the model learned through a large number of iterations, optimizing parameters to improve its ability to capture user behavior patterns. After thorough training, the model achieved an accuracy rate of 86% on the validation set. This result not only showcases the superiority of the NCF model in handling large-scale recommendation tasks, but also indicates its huge potential in practical applications. Through these experiments, we can see that the NCF model based on deep learning can effectively improve the performance of recommendation systems, provide more personalized and accurate recommendation results, and ultimately enhance the overall user experience.

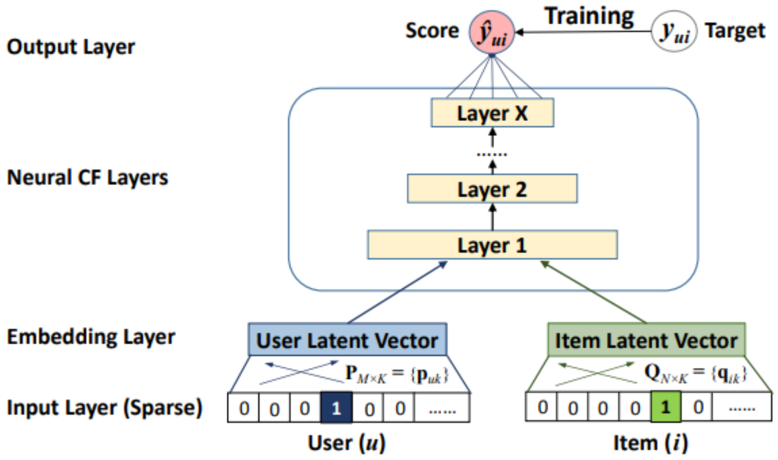

Figure 1. Model architecture.

5. Conclusion

In this study, we used the Neural Collaborative Filtering (NCF) model to analyze and learn the MovieLens dataset using one-hot encoding and embedding methods, ultimately building a movie recommendation system with a high accuracy rate. The research results show that the NCF method has significant advantages in mining the hidden features of users and items. This further verifies the potential of deep learning and neural networks in improving collaborative filtering performance. The work in this study provides strong support for the research and application of recommendation systems, and showcases the broad application prospects of deep learning technology in recommendation systems.

Despite the satisfactory results achieved in accurate movie recommendation by this study, there are still some worthwhile areas for further exploration and improvement. Firstly, the data set used in this study is relatively small, and only a limited number of tags were selected. While this has reduced the computing cost to some extent, it has also limited the model's generalization ability and accuracy. A larger and more comprehensive data set may further enhance the model's performance and generate more precise recommendation results. Secondly, although the NCF model can effectively mine users' latent features and provide recommendations for new users, it still shows insufficient information mining when dealing with the text information before one-hot encoding. Failing to fully utilize these text data may result in certain potential user interests being missed, thus affecting the comprehensiveness and accuracy of the recommendation results. Furthermore, the simplicity of the model, while helpful for understanding and application, also limits its performance in handling complex data relationships.

The design of collaborative filtering-based recommendation systems involves various theories, methods, and technologies. While this study validated the effectiveness of the NCF method, it also exposed issues that need to be addressed in practical applications. In future research and development, several directions deserve further exploration. Firstly, in addition to the embedding method used in this study, more advanced algorithms such as Word2Vec or GloVe can be explored. These natural language processing methods can better capture the semantic information in text, thereby improving the depth and breadth of information mining and further enhancing the performance of the recommendation system [7] . Secondly, although the data set used in this study is of a suitable size, there is still room for further expansion. Training the model with a larger data set can provide a more comprehensive reflection of user behavior patterns and enhance the model's generalization ability and recommendation accuracy. The MovieLens website provides a larger data set, which can be included in future experiments to further verify and enhance the performance of the model. Thirdly, although the current model can provide a certain degree of personalized recommendations, there is still room for improvement in deeply customizing the user experience. Future research can explore more complex user modeling methods, such as combining user social network information and real-time behavior data, to further enhance the personalization and accuracy of recommendations. Furthermore, from the perspective of deep learning, there is still vast room for development in recommendation systems. Deep learning can not only help mine complex user-item relationships, but also enable the fusion of multi-modal data (such as text, images, and videos) to provide more diverse and multi-faceted recommendations. With the continuous advancement of technology, deep learning is expected to bring more innovations in personalized recommendation, real-time recommendation, and context-aware recommendation.

In summary, this study demonstrates the application value of the NCF model in movie recommendation systems, and also reveals the limitations and improvement space of current methods. Future work will focus on optimizing the model algorithm, expanding the scale of the data set, and exploring more diverse personalized recommendation methods to drive further development of recommendation system technology.

References

[1] H.-T. Cheng et al., “Wide & Deep Learning for Recommender Systems,” in Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, in DLRS 2016. New York, NY, USA: Association for Computing Machinery, Sep. 2016, pp. 7–10. doi: 10.1145/2988450.2988454.

[2] H. Guo, R. Tang, Y. Ye, Z. Li, X. He, and Z. Dong, “DeepFM: An End-to-End Wide & Deep Learning Framework for CTR Prediction,” arXiv.org. Accessed: Sep. 08, 2024. [Online]. Available: https://arxiv.org/abs/1804.04950v2

[3] S.-S. Wang, Y.-Y. Pan, and X. Yang, “Research of Recommendation System Based on Deep Interest Network,” J. Phys.: Conf. Ser., vol. 1732, no. 1, p. 012015, Jan. 2021, doi: 10.1088/1742-6596/1732/1/012015.

[4] Z. Shokrzadeh, M.-R. Feizi-Derakhshi, M.-A. Balafar, and J. Bagherzadeh Mohasefi, “Knowledge graph-based recommendation system enhanced by neural collaborative filtering and knowledge graph embedding,” Ain Shams Engineering Journal, vol. 15, no. 1, p. 102263, Jan. 2024, doi: 10.1016/j.asej.2023.102263.

[5] 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.

[6] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, “Neural Collaborative Filtering,” in Proceedings of the 26th International Conference on World Wide Web, in WWW ’17. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, Apr. 2017, pp. 173–182. doi: 10.1145/3038912.3052569.

[7] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), A. Moschitti, B. Pang, and W. Daelemans, Eds., Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162.