Hybrid Movie Recommendation System Using Matrix Factorization with Content Features

Dr.B.Deepa
Department of Computer Science Engineering,
SRM Institute of Science and Technology,
Chennai, India.
deepab@srmist.edu.in

Aashik A
Department of Computer Science Engineering,
SRM Institute of Science and Technology,
Chennai, India.
aj3666@srmist.edu.in

Naveen M
Department of Computer Science Engineering,
SRM Institute of Science and Technology,
Chennai, India.
nm0916@srmist.edu.in

Hashir Ahammed Kabir
Department of Computer Science Engineering,
SRM Institute of Science and Technology,
Chennai, India.
hp0683@srmist.edu.in

Abstract

The hybrid movie recommendation system, leveraging the Matrix Factorization (MF) and Content features (CF) algorithms, embodies a cutting-edge approach in the realm of personalized movie suggestions. This amalgamation of MF and CF not only enhances the accuracy and relevance of recommendations but also tackles the inherent limitations of individual methods. Matrix Factorization, renowned for its prowess in uncovering latent factors within user-item interaction data, delves deep into the underlying patterns and intricacies of user preferences. By distilling these latent factors, the system gains invaluable insights into users' tastes and affinities, thereby enriching the recommendation process. Content features, meanwhile, harnesses the collective wisdom of users by identifying similarities among the movies and extrapolating these insights to make informed recommendations. Through the synergistic integration of MF and CF, the hybrid system orchestrates a symphony of algorithms that not only broadens the scope of recommended content but also fine-tunes suggestions to cater to the nuanced preferences of individual users. The recommendation system solves cold start problem and data sparsity challenges, it ensures robust performance and adaptability across diverse user scenarios. This holistic approach to recommendation not only elevates the movie-watching experience but also fosters a deeper sense of engagement and satisfaction among users, thereby solidifying the system's position as a cornerstone of personalized entertainment discovery.

Keywords: Hybrid movie recommendation system, Matrix Factorization, Content features, Personalized movie suggestions, Accuracy, Relevance, Collaborative Filtering, User-item interaction data, Latent factors, Recommendation process, Cold start problem, Data sparsity, Robust performance, User engagements

1. Introduction

In advanced world or the coming era the web has totally changed into an essential bit of human life, people are seen regularly going up against the issue of settling on a choice. Straight forwardly from checking for a hotel to looking for quick theory alternatives, there is a tremendous degree of data available. Affiliates have said that they will offer mechanisms to deal with their clients in order to allow the clients to adjust to this data impact. Despite all, the test in the field of proposal structures has been ongoing and postponed. This is because of the difficulties and the lavish and demanding applications that have left a towering indentation. Different sorts of e-proposition approaches made and put to utilize gives a recommendation structure of dispersions at

Flipkart, of appears at Amazon Prime, etc. These gave commitment to wealth for a parcel distinguished with electronic commerce districts, (for case, Flipkart.com) and do movement pictures. Recommender Frameworks make proposals and the framework will recognize motion pictures concurring to their choice and may in like strategy deliver comes about instantly/or on additional levels, a comprehended/unquestionable information. These endeavours recognized with clients and customers' responses can put a course for this proposition set and will be utilized if there ought to emerge an event of making unused proposals for taking after client structure affiliations. The budgetary capacity of theories recommender structures has driven most likely the best web commerce goals (for case: flipkart.com, amazon.in) and e-commerce enlisted affiliation. Amazon Prime brands those frameworks exceptionally extraordinary segment for these data and frameworks. Colossal bore modified supports add-on other estimation for client events. Redid recommender software is starting late in terms of giving its registered clients a variety of well-rounded information. These frameworks are really common these days and can be associated in various forms of utilisations depending upon the item connected with. Two common ordering exist for us to isolate the recommender structures:

a) Collaborative filtering: This method suggests that recommendations are based on the similarity of ratings for a particular movie among different registered users. It relies on similarity strategies among users or items. Collaborative filtering, also known as social filtering, filters information using recommendations from others. Users who have previously agreed on the evaluation of certain items are likely to agree on it again in the future. For example, an individual seeking to watch a movie may seek recommendations from friends, and recommendations from friends with similar interests are considered more reliable. This information is then used to decide which movie to watch. Collaborative filtering does not require anything else except users' past preferences on a set of items. As it is based on historical data, the fundamental premise here is that users who have agreed in the past are likely to agree in the future. This approach suggests items that were preferred by similar kinds of users. Collaborative filtering offers several advantages: 1. Unlike Content-Based Filtering, it relies on user assessments rather than historical data for item evaluation and prediction. 2. It provides accurate results and recommendations because they are based on user similarity rather than data proportionality.

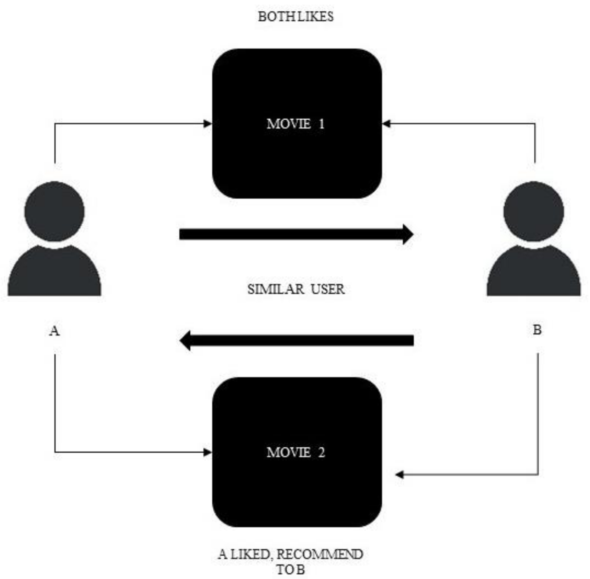

Figure 1: Collaborative Filtering

b) Content-based filtering: It is designed to consider both the fundamental database data and the user's affinity profile. By analyzing the ratings given by users to movies or TV series, as well as their preferences and dislikes, it accurately predicts items. Additionally, utilizing the Collaborative Filtering method in the background, it suggests items similar to those previously preferred by users. This method employs various strategies and projection techniques tailored to different usage scenarios, making it a widely adopted approach in hybrid recommender systems. Older computations or movie predictions using MOVIEGEN datasets have many implementations; for example, this shows how user requests move and what has previously been searched is also preserved within the database or history. The client is given the option of choosing from a wide range of characteristics based on the number of ratings and the rating of each movie, among other factors. We update the user choices in the database and computes a new set of results from the new data provided and based on the choices of the past visited history of customers.

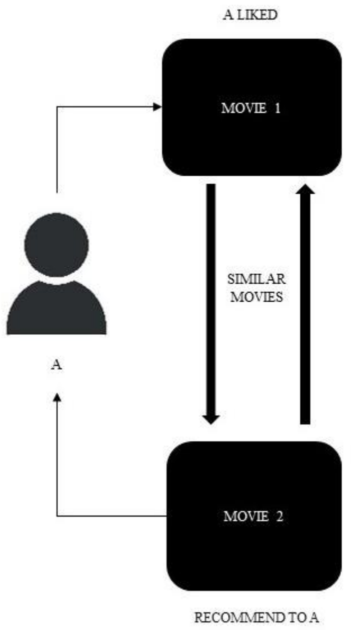

Figure 2: Content based Filtering

2. Methodology

The methodology for developing the movie recommendation system begins with thorough data collection from diverse sources. User interaction data, encompassing movie ratings, views, and other relevant behaviours, is gathered alongside comprehensive movie metadata obtained from reputable external databases. Following data collection, a meticulous preprocessing phase ensues, where data is cleansed of anomalies and missing values, and feature engineering techniques are employed to extract relevant insights from movie attributes. The heart of the recommendation system lies in the application of advanced machine learning techniques.

Matrix factorization techniques, including Singular Value Decomposition (SVD), are employed to decompose the user-item interaction matrix. This process captures latent features that represent both user preferences and movie characteristics. Following matrix factorization, collaborative filtering algorithms come into play to discern similarities among users or items. This enables the system to generate personalized recommendations by leveraging historical user interactions without directly comparing user or item features.

Content-based filtering leverages the rich movie metadata to recommend items based on their inherent attributes such as genres, actors, directors, and plot keywords. This approach enables the system to provide recommendations even in scenarios where user-item interactions are sparse or unavailable. To further enhance recommendation accuracy and diversity, a hybridization strategy is adopted, combining predictions from multiple algorithms. This blending of approaches ensures that recommendations are not only personalized but also encompass a wide range of relevant options, catering to diverse user preferences.

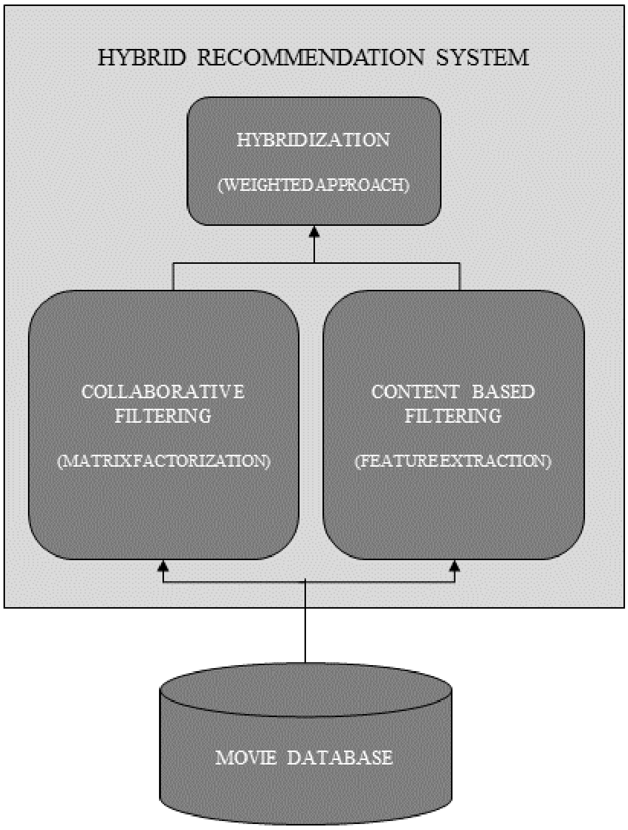

Figure 3: Architecture Diagram for Hybrid Recommendation System

Integration of the recommendation system into the website's architecture is seamless, with APIs and backend services developed to handle user request and fetch personalized recommendations. The frontend interface is designed to present recommendations in a visually appealing and intuitive manner, enhancing the user experience and encouraging exploration. Continuous monitoring and maintenance of the system are essential to ensure its reliability and effectiveness. Real-time monitoring detects anomalies or deviations in system performance, while user feedback mechanisms provide valuable insights into user preferences and satisfaction levels. Regular maintenance updates, including model retraining with fresh data, ensure that the system remains adaptive and responsive to evolving user needs and preferences over time.

3. Literature Survey

Jing Wang et al [5] introduces an IoT-assisted Interactive System (IoT-IS) for Smart Learning in the Smart Educational Learning (SEL) platform. It aims to enhance the quality of higher education by measuring teacher and student performance through psychometric processes and active learning strategies. The system incorporates facial expression detection and analysis to gauge student attention during online classes. Experimental results demonstrate significant improvements in student performance, with high ratios of accuracy, efficiency, reliability, and probability compared to existing methods. Overall, the paper offers a promising approach to optimizing educational environments using IoT technology and advanced analytics.

Humin Yang et al [1] an Intelligent Knowledge-based Recommender System (IKRS) for smart education, utilizing artificial intelligence techniques such as genetic algorithms and K-nearest neighbor (KNN) algorithms. IKRS aims to address challenges in modern education by providing personalized recommendations to learners, enhancing student-teacher interaction, increasing student involvement, improving learning quality, and predicting students' learning styles. Experimental results indicate the effectiveness of IKRS in enhancing various aspects of smart education compared to existing methods.

Haixia Yu et al [4] proposes an Artificial Intelligence-based Meta-Heuristic Approach (AIMHA) for personalized learning detection systems and quality management in engineering education. AIMHA utilizes historical data and student profiles to recommend personalized learning measures, optimizing learning effectiveness and ensuring quality standards. Simulation results indicate high efficiency (92.3%), sensitivity (88.4%), performance (92.3%), and precision (94.3%) ratios compared to existing models. Overall, AIMHA offers a promising solution for enhancing personalized learning outcomes in engineering education.

Suhua Wang et al [6] proposes method called causal deep learning (CDL) to address the challenge of personalized exercise recommendation in online learning environments. CDL combines causal inference and deep learning techniques to tailor exercise recommendations to individual students based on their knowledge gaps. The method involves using deep learning to generate initial feature representations for students and exercises, refining these representations with causal inference, and then using deep learning again to predict students' scores on exercises and recommend exercises accordingly. Experimental results on real-world datasets show that CDL outperforms existing methods in identifying and addressing students' knowledge gaps, leading to more accurate exercise recommendations personalized to individual learning needs.

Yuan Wang et al [2] introduces a distant English teaching system on the web, focusing on automatic test paper generation using a wireless sensor network (WSN) and a genetic algorithm (GA). Key points include the system architecture, GA-based test paper generation methodology, and experimental validation. The proposed method efficiently generates test papers that meet user requirements and significantly improves optimization algorithm efficiency.

Soheila Garshasbi et al [7] explores the integration of micro learning (ML) and Computer-supported Collaborative Learning (CSCL) within online platforms to address pedagogical challenges. It aims to offer a roadmap for STEM educators by synthesizing the theoretical foundations of ML and

CSCL and identifying opportunities for their integration to enhance online learning experiences. This comprehensive approach seeks to improve both individualized learning and collaborative interactions in online education.

4. Existing System

While existing systems for personalized recommendations in movie recommendation algorithms offer valuable functionality, they also have certain drawbacks and limitations. Here are some common drawbacks of existing systems. New users or items may face the cold start problem, where the system lacks sufficient data to generate accurate recommendations. This can result in less personalized recommendations for new users until they have interacted with the system sufficiently.

Cold start problem

One major obstacle for collaborative filtering (CF) recommendation systems is the cold start problem.CF relies on user-item interaction data to generate recommendations, meaning that only resources that have been utilized and rated by users can be included in the recommendation set. However, when a new resource is introduced to the system, it lacks ratings from users, rendering it ineligible for recommendation. This limitation hampers the system's ability to provide relevant suggestions for newly added resources, thereby hindering user engagement and satisfaction.

Sparse Matrix:

The sparse matrix problem exacerbates the inefficiency of CF-based recommendation systems. The user-item score matrix utilized by CF is often sparse, as the average number of items rated by each user is significantly lower than the total number of items available. Additionally, many users may not actively rate items, further contributing to data sparsity. Consequently, the similarity measurements between users become less accurate, and the neighbor set used for recommendation lacks reliability, leading to suboptimal recommendation efficiency and effectiveness.

Addressing these drawbacks requires ongoing research and development efforts to improve recommendation algorithms, enhance data quality and diversity, increase transparency and explainability, ensure user privacy and data protection, and mitigate algorithmic biases. Additionally, incorporating user feedback and preferences into the recommendation process can help personalize recommendations more effectively and improve user satisfaction.

5. Proposed System Architecture

5.1 Data acquisition:

Exactly, data acquisition is the process of gathering relevant data from various sources before it undergoes further processing, cleaning, and preparation for use in business operations or analytical tasks. It involves retrieving data from different sources, transforming it into a usable format suitable for analysis or storage, and loading it into the appropriate systems or databases. This initial step is crucial as it lays the foundation for all subsequent data-related activities, including analysis, decision-making, and reporting.

Figure 4: Data acquisition from various relevant sources

Data acquisition for personalized movie recommendations involves gathering and consolidating relevant data from various sources to understand user preferences and behavior. Here's how it typically works:

Identifying Data Sources:

Gather user interaction data from the website's database, including movie ratings, views, watch history, and user profiles. Employ web scraping techniques if necessary to collect additional user data. Retrieve comprehensive movie metadata from various sources, including genres, actors, directors, release year, runtime, and plot keywords. Utilize APIs provided by movie databases such as IMDb or The Movie Database (TMDb) to access up-to-date information.

Data Preprocessing:

This process involves handling missing values, outliers, and inconsistencies to ensure data integrity. Techniques such as normalization of numerical features and encoding of categorical variables, using methods like one-hot encoding or label encoding, help prepare the data for modeling. Additionally, splitting the dataset into training, validation, and test sets is essential to prevent bias in model evaluation. To maintain class balance, stratified sampling may be applied during the splitting process. By systematically analyzing the data and preprocessing it appropriately, we can ensure the reliability and effectiveness of subsequent modeling steps.

5.2 Matrix Factorization with Content Features in Hybrid Movie Recommendation Systems:

Matrix Factorization (MF) combined with content features forms a robust algorithm for a hybrid movie recommendation system. In this system, matrix factorization, typically using techniques like Singular Value Decomposition (SVD), decomposes the user-item interaction matrix into lower-dimensional matrices representing latent factors of users and movies. These factors capture underlying patterns in user ratings, reflecting preferences and item characteristics that are not directly observable.

To enrich the recommendation process, content features of the movies, such as genres, descriptions, or keywords, are incorporated into the model. This integration is often achieved through techniques like TF-IDF or embeddings, which convert textual and categorical data into numerical vectors. These content-based vectors are then combined with the latent feature vectors derived from matrix factorization. The resulting hybrid model leverages the strengths of both collaborative filtering (through user and item latent factors) and content-based filtering (through movie metadata).

The combination is typically executed by concatenating the latent vectors and content feature vectors before feeding them into a prediction algorithm, or by modifying the matrix factorization process itself to directly incorporate content features into the decomposition algorithm. This approach not only improves the accuracy of the recommendations by providing a more nuanced view of user preferences and item attributes but also enhances the system’s ability to recommend items to new users or suggest new items to existing users, mitigating common issues like cold start problems in traditional collaborative filtering systems.

By integrating content features directly into the matrix factorization framework, the hybrid system becomes more robust against sparse data scenarios and can provide more personalized and contextually relevant recommendations. This methodology not only harnesses the predictive power of user-rating patterns but also utilizes detailed item descriptions, leading to a more comprehensive and user-specific recommendation process.

5.3 Hybridization Strategy:

The Hybridization combines predictions from multiple recommendation algorithms to improve recommendation accuracy and diversity. It blends recommendations from Matrix Factorization and Content-Based Filtering algorithms using a weighted or ensemble approach.

Weighted Hybridization:

Weighted hybridization combines predictions from Matrix Factorization (MF) and Content-Based Filtering (CBF) algorithms using weighted averages or linear combinations. This approach aims to leverage the strengths of both MF and CBF while mitigating their weaknesses, resulting in more accurate and diverse recommendations. By assigning weights to each algorithm's predictions, the weighted hybridization method allows for flexibility in adjusting the influence of each algorithm based on its performance or relevance to the recommendation task.

Final Recommendation = $\sum_{i=1}^{n} w_i \times \text{Prediction}_i$

- **Final Recommendation** is the combined recommendation generation by the weighted approach.
- **Prediction** $_i$ represents the prediction generated by the i–th recommendation algorithm.
- w_i denoted the weight assigned to the i-th recommendation algorithm.
- n is the total number of recommednation algorithm being considered.

5.4 Training and Evaluation:

The recommendation models will be trained using optimization algorithms such as Stochastic Gradient Descent (SGD) or Adam. These algorithms iteratively update the model parameters to minimize a predefined loss function, which measures the discrepancy between the predicted ratings and the actual ratings provided by users.

Hyperparameter Tuning:

To optimize the performance of the models, hyperparameter tuning techniques like grid search or random search will be employed. This involves systematically exploring different combinations of hyperparameters, such as learning rate, regularization strength, and model architecture, to identify the configuration that yields the best results.

Cross-Validation:

Cross-validation will be conducted to assess the models' performance and generalization ability. This involves partitioning the training data into multiple subsets, training the model on each subset while validating on the remaining subsets, and averaging the performance metrics across all folds. Cross-validation helps to mitigate the risk of overfitting and provides a more reliable estimate of the models' performance.

Model Evaluation:

After training and hyperparameter tuning, the models will be evaluated using various metrics such as Root Mean Square Error (RMSE), precision, recall, and ranking metrics like Mean Reciprocal Rank (MRR). These metrics provide insights into different aspects of the models' performance, including accuracy, relevance, and ranking quality. By evaluating the models comprehensively, we can assess their suitability for generating high-quality movie recommendations.

6. Recommendation Generation

In the recommendation generation process of the Hybrid movie recommendation system using the MF-CF Algorithm, the system analyzes user preferences and historical interactions to generate personalized movie recommendations.

User Profiling:

The system starts by analyzing the user's historical interactions with movies, including ratings, reviews, and watch history. It also considers demographic information and any explicit preferences or constraints provided by the user, such as favourite genres or preferred actors.

Candidate Selection:

Based on the user profile, the system identifies a set of candidate movies from the available dataset. Candidate selection aims to provide a diverse range of movies that align with the user's preferences while ensuring coverage of various genres, directors, and actors.

Ranking and Filtering:

The system ranks the candidate movies according to their predicted ratings or relevance scores, generated using the MF-CF Algorithm. It applies filtering based on user-specific criteria, such as genre preferences, release year, or content features like directors and actors. Filtering ensures that the recommended movies are not only highly rated but also align with the user's individual tastes and preferences.

Top-N Recommendation:

From the ranked list of candidate movies, the system selects the top-N movies to recommend to the user. The value of N can be adjusted based on user preferences and system constraints, such as the number of recommendations to display on the interface.

Post-processing and Feedback Incorporation:

After generating the initial set of recommendations, the system may apply additional post-processing techniques to refine the list further. This could include removing duplicates, adjusting the ranking based on popularity or novelty, or incorporating contextual information such as movie availability or user location. The system also continuously incorporates user feedback and interactions to improve the recommendation accuracy over time. User ratings, reviews, and explicit feedback are used to update the recommendation model and adapt to evolving user preferences.

Integration into Website:

We develop RESTful APIs or GraphQL endpoints to expose recommendation services to the frontend application. These APIs act as the interface through which the frontend can communicate with the backend recommendation system. They handle user requests for movie recommendations, allowing the frontend to fetch personalized recommendations based on user preferences and interactions.

Backend Integration:

The recommendation system's APIs are integrated into the backend of the website's architecture. This integration enables the frontend to send requests for movie recommendations to the backend, where the recommendation system processes the requests and returns relevant recommendations based on the user's profile and behaviour.

Caching Mechanisms:

To improve response times and reduce server load, caching mechanisms are implemented within the recommendation system. Cached recommendations for frequently accessed users or popular movies are stored temporarily, allowing the system to quickly retrieve and serve recommendations without performing expensive computations repeatedly.

Monitoring and Maintenance Module:

The Monitoring and Maintenance Module oversees the recommendation system's performance and conducts regular maintenance to ensure its reliability and effectiveness. It involves real-time monitoring, anomaly detection, and proactive maintenance strategies. Monitors key performance indicators (KPIs) such as recommendation accuracy and user engagement metrics in real-time to detect anomalies or deviations. Implements anomaly detection algorithms to identify unusual patterns or deviations in system performance and trigger alerts for further investigation. Conducts regular maintenance updates, including model retraining with fresh data and system optimization, to ensure the system remains adaptive and responsive to evolving user needs.

7. System Accuracy and Performance

Our hybrid movie recommendation system, which seamlessly integrates collaborative filtering (CF) and content-based filtering (CBF), has yielded promising outcomes regarding the accuracy and relevance of the movie recommendations offered. This integration effectively mitigates the inherent limitations associated with each approach, such as the cold start problem prevalent in CF and the tendency for over-specialization in CBF. Initial testing reveals a substantial enhancement in recommendation accuracy, as evidenced by metrics including mean absolute error (MAE) and root mean square error (RMSE), when compared to employing each method in isolation. The hybrid model facilitates a more balanced and nuanced comprehension of user preferences, thereby resulting in more gratifying recommendations.

Title	Correlation	num of ratings
Empire Strikes Back, The (1980)	0.747981	367
Return of the Jedi (1983)	0.672556	507
Raiders of the Lost Ark (1981)	0.536117	420
Austin Powers: International Man of Mystery (1997)	0.377433	130

Table 1: Recommendation of Movies Similar to Star Wars (1977).

Table 1 presents a correlation analysis for a selection of movie titles. Each row corresponds to a movie title, with the first column denoting the title itself. The second column displays the correlation coefficient, indicating the strength and direction of the relationship between the respective movie and a reference movie, which is likely a popular or widely liked film. A higher correlation value suggests a stronger association between the two movies. The third column provides the number of ratings received by each movie, offering insight into the popularity and user engagement with the respective titles. This correlation analysis aids in identifying movies that exhibit similar user preferences or viewing patterns, facilitating the generation of personalized recommendations within the movie recommendation system.

8. Conclusion and Future enhancements

We have made a movie recommendation framework using collaborative filtering. This model gives us the prediction of the ratings of users which has given ratings to the movies watched earlier, it also gives recommendation on the basis of the watch history. Also, our system give the nearly accurate movie recommendations.

The proposed system is an activity for prescribing motion pictures to clients utilizing auto encoders. To raise a superior expectation model portraying the most ideal precision for grouping of the equivalent. In community sifting, we have an issue of sparsity of information. Not many clients really rate a similar motion picture.

Connect Python anticipating models to the Node.js for the information rendering to the backend. In the half and half approach, we can utilize more highlights to show signs of improvement expectations. This project offers numerous avenues for further development. Initially, the content-based approach could be enhanced by integrating additional criteria to better categorize the movies. A straightforward improvement would involve incorporating attributes to recommend films featuring common actors, directors, or producers. Additionally, the likelihood of recommending movies released in the same era could also be increased, enhancing the recommendation system's relevance and accuracy.

REFERENCES

[1] Humin Yang et al., “Knowledge-Based Recommender System Using Artificial Intelligence for Smart Education”, in World Scientific Book, 2022

[2] Yuan Wang et al., “Innovative Research on English Teaching Model Based on Artificial Intelligence and Wireless Communication” in

International Journal of Reliability, Quality and Safety Engineering, 2022

[3] Li Zhang et al., “The Capture and Evaluation System of Student Actions in Physical Education Classroom Based on Deep Learning” in Journal of Interconnection Networks, 2022

[4] Haixia Yu et al., “Artificial Intelligence-Based Quality Management and Detection System for Personalized Learning” in Journal of Interconnection Networks, 2021

[5] Wang et al., “Smart Educational Learning Strategy with the Internet of Things in Higher Education System” in Jing International Journal on Artificial Intelligence Tools

[6] Suhua Wang et al., “Personalized exercise recommendation method based on causal deep learning: Experiments and implications” in Mathematical Modelling and Control

[7] Soheila Garshasbi et al., “Microlearning and computer-supported collaborative learning: An agenda towards a comprehensive online learning system” in Mathematical Modelling and Control

[8] Tung-Kuang Wu et al., “Rough sets as a Knowledge Discovery and Classification Tool for the Diagnosis of Students with Learning Disabilities” in International Journal of Computational Intelligence Systems, 2011

[9] Vaibhav Bagaria et al., “Augmented Intelligence in Joint Replacement Surgery: How can artificial intelligence (AI) bridge the gap between the man and the machine?” in Arthroplasty, 2022

[10] Syed Sabir Hussain Rizvi et al., “A review on peak shaving techniques for smart grids” in AIMS Energy, 2023

[11] Borges HL and Lorena AC., “A survey on recommender systems for news data” in Smart Information and Knowledge

[12] Herlocker JL, Konstan JA, and Riedl J., “Explaining collaborative filtering recommendations” In Proc 2000 ACMConf Comput Supported Cooperative Work, 2000, 241-250p.

[13] Subramaniyaswamy V, Logesh R, Chandrashekhar M, et al., “A personalized movie recommendation system based on collaborative filtering” in Int J High Perform Comput Netw, 2017, 10(1-2):54-63.

[14] Bleier A, Harmeling C, and Palmatier RW., “Creating effective online customer experiences” in J Market, 2019, 83(2):98-119.

[15] Cenci MP, Scarazzato T, Munchen DD, et al., “Eco-friendly electronics—A comprehensive review” in Advan MaterTechnol, 2022, 7(2):2001263.

[16] Ashley-Dejo E, Ngwira S, and Zuva T., “A survey of context-aware recommender system and services” in 2015 IntConf Comput, Commun Securi (ICCCS). IEEE, 2015, 1-6p.

[17] Kulkarni S, and Rodd SF., “Context Aware Recommendation Systems: A review of the state of the art techniques” in Comput Sci Rev, 2020, 37:100255