

Integration of Explicit and Implicit User's Feedback with Contextual Information in Hybrid Recommender System

^{*1}Hafiz Ibrahim, ²Abba Almu and ¹Anas Tukur Balarabe

¹Faculty of Computing, Department of Computer Science, Sokoto State University, Sokoto State, Nigeria.

²Department of Computer Science, Usmanu Danfodiyo University, Sokoto State, Nigeria.

*Corresponding authors' email: feezee02@gmail.com

ABSTRACT

Recommender system helps users identify relevant content from the growing volume of information available on digital platforms. However, user preferences are dynamic and may not be adequately represented by a single source of feedback. This study proposes a hybrid movie recommendation system that integrates explicit feedback, implicit behavioural feedback, and temporal contextual information Using a convolutional neural network (CNN). Explicit feedback is represented by user's movie ratings, while implicit feedbacks represented by interaction frequency. Contextual information is derived from users' interaction timestamp to capture changes in preference over time. The movie lens 1m is used as the main source dataset, from which 10,000 rating records were selected for the experiment. The selected records were pre-processed to construct the three information sources and used to train and evaluate the proposed model. Performance was accessed using precision, Recall, F1-score and mean squared error (MSE). The model achieved a precision of 86.25%. Recall of 98.76%, F1-score of 92.08%, and MSE of 0.0845. The results indicate that the Proposed architecture can integrate explicit feedback, implicit feedback and contextual information within a unified recommendation framework. The study demonstrates the feasibility of combining these information sources for personalized movie recommendation. The findings provide a basis for developing recommendation systems capable of delivering more relevant and personalized content in dynamic digital environments.

Received: 21 Sep. 2026

Accepted: 08 Oct. 2026

Published: 12 Oct. 2026

Keywords:

Recommendation, MovieLens, Explicit Feedback, Implicit Feedback, Contextual Information, Personalized Recommendation. Deep learning

INTRODUCTION

The current availability of internet connectivity on a global scale, product manufacturers, vendors, retailers and service providers have the exceptional advantages to advertise to a huge number of customers worldwide. In the same vein consumer's interest have increased enormously. Even though this opens up a vast range of potentialities and wider range of business opportunities, but it also has some significant ramifications. On the one hand, it became very difficult for business owners to boost the effectiveness of their advertising since it was difficult to choose and target just customers whose preferences best matched their products or services given the variety of demands among them. Recommender systems (RSs) attempt to rectify this problem by analysing consumer's preferences and attempting to predict the preference of user for an item, such as product or service. RSs are software tools and techniques that provide suggestions for items that are more likely of interest to a particular user. The suggestions are related to various decision-making processes, such as what items to buy? what music to listen to? or what online news to read? Or which movie to watch? The general term used to describe what the system recommends to users. With the aid of accurate recommendations, users can spend less time looking for items they want and have more enjoyable shopping experience, which enhances sales and profitability for business owners on the internet. More recently, the problem of today's information overload which is one of the current challenges of internet users, has been effectively addressed by personalized recommendation technologies (Yan and Xie., 2020). The mid-1990s witnessed

the emergence of recommender system as independent research area and interest in the field of RS has dramatically increased (Ricci et al., 2015).

Recommender systems are classified into three major categories namely content-based filtering (CBF), Collaborative filtering (CF) and hybrid filtering.

CBF consider the previous items that user liked and recommend similar items to the user. In this type of recommender system user past experienced such as rating history, browsing behaviour and items features are the core attributes.

CF generates recommendation based on the similarity between users. In this category it's considered that users with similar attributes may have the same tastes. CF is well-known and extensively applied method for analysing user interest in the target item based on views, expressed by other similar users. Model based collaborative filtering uses machine learning to discover pattern and make predictions. Hybrid filtering recommender system combines two or more recommendation techniques to improve accuracy and address the limitations of the individual method.

Several studies were conducted to improve traditional collaborative filtering based on users' preferences. (Li et al., 2017; Wang et al., 2018; Zhou et al, 2020, Chengyu and liu 2025, Among these studies, the study by (Y Gao, H Zheng, H Cui 2025) integrated CNN model using content based by analysing user preferences, behaviour records and utilizing movie content elements to accurately predicts the possibility that the movie would be accepted. with high consideration of user positive evaluations, to help recommender system to

rank relevant movies to the target users. Despite all the effort, the study by (Y Gao, H Zheng, H Cui 2025), failed to incorporate explicit and implicit users feedback and contextual information that affect system usability and adaptability. This research work proposed incorporating user feedback (explicit & implicit) and contextual information (temporal features) in recommender system in order to recommend movies that align with users preferences to avoid irrelevant recommendations.

The goals of the study is to build a robust recommender system that incorporates users feedback and contextual information for personalized and accurate movie recommendation that align with users preference by developing a CNN model capable of incorporating multiple feedback into a recommendation model using movie Lens dataset and to design a model that incorporate contextual information to improve system usability for accurate recommendations that align with users preferences.

Related Work

The Authors (Gao et al., 2025) proposed a User preference modeling for movie recommendations based on deep learning. The system used Artificial Intelligence (AI), graph-based techniques, and text mining to accurately estimate user preferences. While PageRank ranks the films based on their importance in the individual's history of surfing, Convolutional Neural Network (CNN) predicts the possibility that the movie would be accepted. The experiments employed a dataset of 215 users' browsing activity in 508 movie pages for evaluation. The presented approach achieved significant enhancement in recommendation precision and recall metrics resulting in 7.15% precision expansion and 5.19% recall growth which indicates its potential implementation in personalized movie recommendation systems. However, despite the promising results achieved, the model failed to incorporate user feedback and contextual information which may affect the system usability. Additionally, the dataset used by the existing studies contains restricted user and movie participation which might not reflect actual real-life situations.

The Authors S. Gupta, A. K. Bindal, and D. Prasad proposed a multimodal graph-based recommendation system using a hybrid filtering approach. The system uses various sources of data and advanced graph-based deep learning algorithms to provide more accurate and personalized recommendations to users. The system captures user and item attributes using text, images, videos, and Meta data. The system incorporates these attributes into the graph of user-item interactions using collaborative filtering and content-based filtering. The Graph convolutional networks (GCNs) help the system to identify collaborative filtering attributes. The intrinsic characteristics of items can be better understood and utilized with graph-based content based filtering. The proposed model initially classifies related users and items into groups using unsupervised clustering, then refines its recommendations using a cross-attention approach. Experimental results show that the proposed model provides more accurate and personalized recommendations than existing models. We conduct comprehensive experiments using the publicly accessible datasets such as Movielens 1M, TikTok, MovieLens 10M, and Micro Video 1.7M. Our proposed model demonstrates superior effectiveness compared to the state-of-the-art multimedia recommender systems in various evaluation parameters such as precision, accuracy, recall, Normalized Discounted Cumulative Gain (NDCG), F1-score and RMSE. However, a notable limitation of the study is the increased computational complexity and resource

requirements associated with integrating multimodal data and advanced graph-based methods. This approach faces challenges in scaling to large-scale datasets or real-time applications. Researchers can improve recommendation accuracy in the future by capturing complex user-item interactions using more advanced attention mechanisms and deep learning architectures. Adding more advanced recommendation methods, such as meta-learning and reinforcement learning, to the MGRS-HFA model can enhance its performance and flexibility. Additionally, optimization techniques and distributed computing can significantly improve the system's ability to handle large datasets

Shao and Guo proposed a hybrid convolutional network that automatically generates movie genres in streaming services. It outperformed the advanced models with an impressive F1 score and a 94.4% accuracy rate. This study showcases the effectiveness of hybrid convolutional networks for genre classification. However, the paper provides insufficient details about its hybrid model structure along with its data processing methods.

Zhang et al. proposed the Movie genre Classification method based on Language augmentation and shot sampling (Movie-CLIP) approach, which enables the classification of movie genres. As a result, this model improved the average mean Average Precision by 6–9% on the Movie Net and Condensed Movies datasets and by 1.1% for the scene boundary detection task. This study presents an innovative approach by combining language augmentation and shot sampling, improving performance in genre classification. However, its complexity and reliance on specific datasets might limit its broader application

Feng et al., (2019), proposed a new method to enhance the performance of collaborative filtering based on user-item community detection (UICDR). The method combines co-clustering and CF algorithms to effectively identify user preference using user-item transaction data. Co-clusters are groups of both users and items with similar patterns. Users can have several interests, and items might satisfy several needs. In this method bipartite network is constructed using user-item interaction data, users and items are partitioned into several subgroups. After getting clusters with tightly linked users and items, traditional CF model trained for- each cluster. Using four real world dataset the evaluation results of UICDR showed that it's significantly improved the performance of Top-N recommendation when compare with tradition CF. However, the clusters may be weak if more than one user shared same similarities and preference toward one item.

Siakhiyevich and Huang (2019), presented an approach of building user interest and preference profiles for personalized Recommender System. The user interests' profile is obtained from item general description (such as item categories or tags) and user preference profile is obtained from user reviews and describes what exactly user liked about a particular item. The approach comprises three main steps: obtaining items features, building user interests and preference profiles and provides top personal recommendation based on user interests and preference profiles. It's also incorporated Embedding' s from language models (ELMo), to represent item features and user reviews in vectors. ELMo is a deep contextualized word representation that models the complex characteristics of the use of words and their differences in the language context. Cosine similarity measure is employed to compare item features and user reviews and stored the most similar user reviews item features in preference profile. The evaluated

results showed that the approach outperforms the different state-of-the-art algorithms in the accuracy of top personal recommendations. However, it's greatly relied on user reviews which might not be sufficient as some users provide small percentage or ambiguous reviews which may not be enough to predict users' interest.

Zhou et al., (2020) proposed personalized collaborative filtering recommendation algorithm based on user preference and user profile in advanced to traditional collaborative filtering. The method employed user interest model and construct user model which improved the traditional user model. The user interest model uses user preference degree and indicates how much user likes movie of a certain category. The larger the value of user's preference degree the more user likes movie of a certain category. The user profile weight (UPW) is also utilized in order to identify the actual similarity between users. The cosine similarity is used to calculate the similarity distance between two users. The UPW adjust the similarity weight between users in the final calculation of user similarity. The more similar the attributes between users, the higher the final similarity. The results showed that incorporating user profile and user preference gives more accurate recommendation results than traditional recommendation algorithm. However, the system is unable to consider negative feedback in order to optimize users' preference. It also considers two or more users with consistencies in age, job and gender to have the same preference which might not be the case since the users can have difference taste and interest as a result of environment factor and status level.

(Liu, 2025) proposed a deep learning-based Movie Recommendation System designed to efficiently capture user preferences and movie features, thereby providing personalized viewing suggestions. We innovatively applied Neural Collaborative Filtering (NCF) to build a movie recommendation system, integrating embedding techniques with a multi-layer neural network. This approach maps user and movie IDs to a high-dimensional space, effectively capturing complex relationships and latent features. Experimental results show that the system achieved an impressive Hit Ratio of 0.89, highlighting the power of deep learning models in handling large-scale user-movie interaction data and enhancing user experiences and platform engagement. However, despite the accomplishments of our research, several limitations remain that require attention in future work. First, while the Movie Lens dataset is widely recognized, its data structure and user behaviour patterns may not fully reflect real-world scenarios. As a result, although the model performs well on this dataset, further adjustments are necessary to better align the model with practical applications. To more accurately capture real-world user behaviour, second, despite the models strong performance in key metrics like Hit Ratio, biases might still exist due to dataset imbalances or potential model over fitting. Addressing these issues will require increasing dataset diversity, implement regularization techniques, and apply cross-validation methods to mitigate the risk of over fitting. Furthermore, the current model primarily relies on users' historical behaviour data for recommendations, neglecting other influential factors, such as users' social networks or

geographical location. To address this, we propose incorporating external data sources into the system to enhance recommendation quality

MATERIALS AND METHODS

This research follows a step-by-step experimental approach to achieve its aim and objectives, which structured around five (5) different key stages. Each build upon the output of the previous one, ensuring a clear and systematic methodology.

System Architecture

The proposed hybrid CNN-based recommendation architecture was developed to incorporate explicit, implicit feedback and contextual information for personalized movie recommendation. The architecture begins with user- movie interaction data obtained from the dataset. The input features are organized according to the type of information they provide about user preference. Explicit feedback is represented primarily by user's movie ratings, while implicit behavioural information is represented through interaction count recency. Contextual information is obtained from the timestamp associated with each interaction and transformed into a recency representation and normalized recency score. Prior to the model training the categorical user and movie identifiers are encoded and transformed into a learnable embedding representation. The rating values are normalized from their original 1-5 scale to the 0-1 range using min-max scaling. This normalization improves numerical consistency during neural network training while retaining the relative ordering of the user's preferences. Interaction count is used to represent the level of user engagement whereas regency provides information about the temporal freshness of the user's interaction. The resulting feature representations are subsequently combined to form a unified representation of the user-movie interaction. The fusion constitutes the hybrid aspect of the proposed model because information from different sources is made available to the deep learning component rather than relying solely on explicit ratings. The combined representation is then supplied to the CNN-based feature learning module. Within the CNN module, one dimensional convolutional layer is used to extract meaningful representation. Maxpooling layers reduce the dimensionality of the resulting feature maps while retaining important learned features. Multiple convolution and pooling operations enable the network to progressively learn higher level relationships among the user, movie, behavioural and temporal features. The resulting feature maps are subsequently flattened into a one-dimensional feature vector. The flattened representations passed through fully connected layers consisting of dense 256. These layers learn nonlinear relationships among the extracted features and progressively transform them into a representation suitable for prediction. Dropout is incorporated to reduce over fitting and improve the capability of the model. Finally, a single output neuron produces the predicted normalized rating. Since the target rating is normalized to the 0-1 range, the output represents the estimated preference score on the same normalized scale and can be transformed back to the original rating scale 1-5 when required as presented in figure 1.

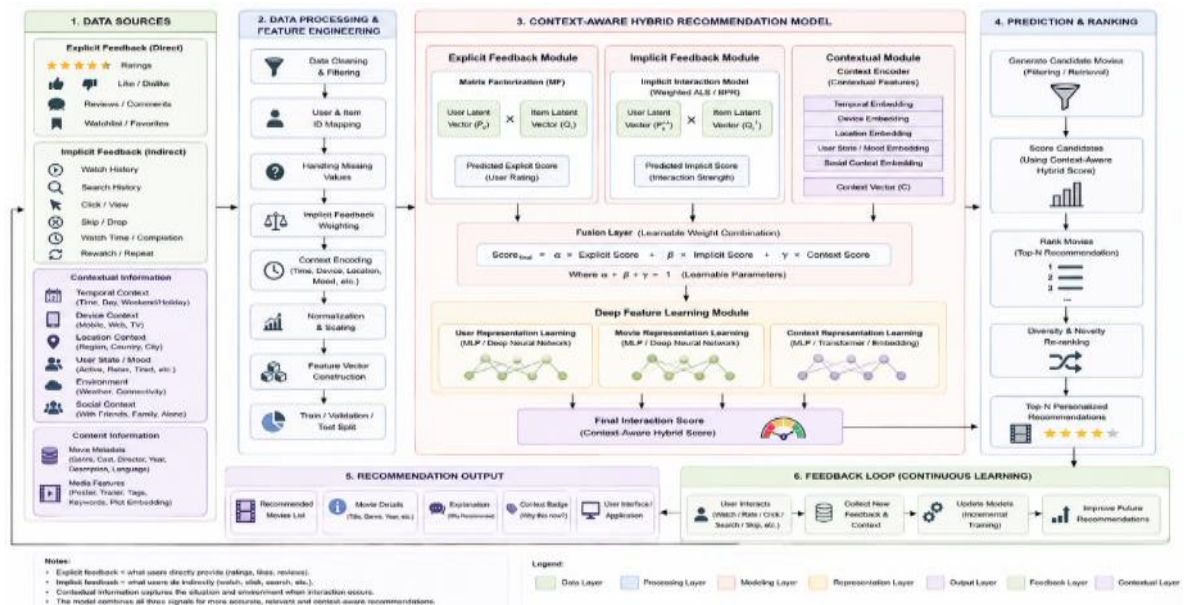


Figure 1: Model System Architecture

Incorporation of Explicit and Implicit Feedback with Contextual Information

Explicit feedback represents the user's expressed preference for a movie and is obtained from the movie-rating records. Users express their preferences by assigning ratings on a scale of 1 to 5. These ratings provide direct information about the user's evaluation of a particular movie and therefore serve as the ground preference information for training the proposed hybrid CNN-based recommendation model. Explicit feedback directly indicates the user's stated level of preference for a movie. Prior to model training, the original rating values are normalized using Min-Max scalar to transform the ratings into a consistent numerical range between 0 and 1. This normalization reduces differences in numerical scale and allows the rating feature to be handled consistently with other features that may be incorporated into the recommendation model. The min-max is shown by the following equation 3.1:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Where,

(X) Represents the original feature value

(X_{min}) represents the Minimum value of the feature

(X_{max}) represents maximum feature value

(X) Normalized feature value

The incorporation of explicit feedback in the proposed hybrid CNN-based recommendation model is achieved through a supervised regression framework in which the original 1–5 ratings are therefore transformed into values ranging from 0 to 1. This normalization places the target values on a consistent numerical scale and is appropriate for the regression-based learning objective adopted by the model. A rating of 1 is transformed to 0, a rating of 3 is transformed to 0.5, and a rating of 5 is transformed to 1. The normalized explicit rating is not directly supplied to the CNN as an input feature. Instead, it is assigned as the target variable (y) that the model is trained to predict. The CNN does not use the rating as an input feature; rather, it learns to predict the normalized rating from the encoded user and movie representations together with the derived implicit-feedback score. The predicted preference score is evaluated against the actual normalized rating using MSE loss, allowing the model to adjust its learned representations and parameters during

training. This approach enables the proposed model to exploit the complementary characteristics of explicit and implicit feedback, with explicit ratings providing direct preference information and implicit behavioural signals providing additional evidence about the user's interaction patterns. The final predicted normalized score can subsequently be interpreted as the model's estimated level of user preference for a movie and can be used to support movie recommendation and ranking. Therefore, the explicit feedback is demonstrated in the final model output through the prediction of the user's rating, while its contribution to model learning is demonstrated through the comparison of the predicted rating with the normalized ground-truth rating using the MSE loss function. The CNN-based model learns the relationship among these inputs and generates the estimated user rating for the corresponding movie.

The Implicit feedback is derived from users' interaction behaviour patterns with movies and provides information about user engagement that may not be explicitly expressed through ratings. In the proposed hybrid CNN-based recommendation model, two behavioural characteristics are considered to represent implicit preference: user interaction frequency and interaction recency. The combination of these characteristics enables the model to capture both the extent of a user's engagement with movies and the temporal relevance of the user's interactions. The first component interaction frequency, is represented by the user interaction count. This measure is obtained by user identifier and counting the number of movie records associated with each user. A higher interaction count indicates a greater level of user engagement with the movie collection. Since the resulting counts may have different numerical magnitudes, they are normalized to the interval 0 and 1 using Min-Max scaling. The second component of the implicit-feedback representation is interaction recency, which captures the temporal relevance of users' movie interactions. Recency is derived from the timestamp associated with each user movie interaction. The implementation first identifies the most recent timestamp in the dataset and uses it as the reference point. The temporal distance of each interaction from this reference time is then calculated. The system calculates the number of movies associated with each user as an interaction-count measure and

normalizes this value. In addition, recency is derived from the timestamp of each interaction, where more recent interactions are assigned greater relevance through a recency score. These two behavioural components are combined with equal weights to produce an implicit score. In particular, the frequency of user interactions is used to represent the strength of implicit preference. The recency of each interaction is considered to account for changes in user preferences. This function assigns higher values to recent interactions, which approach 1, while interactions that occurred further in the past receive progressively lower values, approaching 0. Consequently, recent user activities have a greater influence on the representation of user preferences than older activities. The frequency and recency components are subsequently combined using a weighted average, with equal importance assigned to both factors. The interaction and the recency features are blended by taking a balance weighted average (0.5 recency score + 0.5 frequency score). The frequency score is obtained by aggregating the total number of movies with which each user has interacted. The CNN therefore learns latent representations from the user and movie embedding, while the implicit score provides additional information concerning the user's behavioural engagement. The interaction count is min-max then the recency score is calculated through the following equation 3.2

$$Recency = T_{max} - T_i; R_i = \frac{1}{1+Recency} \quad (2)$$

Where,

T_{max} is the max is the maximum timestamp in the dataset

T_i is the timestamp associated with the interaction.

R_i is the elapsed time of the interactions

Contextual Information Incorporation

Contextual information was incorporated into the recommendation model by considering the recency of users' interactions. The system uses the timestamps associated with each user interaction to determine how recently the interaction occurred. For each interaction, the time difference between the most recent interaction timestamp and the timestamp of the specific interaction is calculated. This time difference is then transformed into a bounded recency score, where more recent interactions receive higher scores and older interactions receive progressively lower scores. The resulting recency score is subsequently combined with the implicit interaction score to produce an updated user-item preference score. In this way, the model gives greater importance to recent user behaviour while still retaining information from historical interactions. The incorporation of contextual recency therefore enables the system to capture changes in user interests and improve the relevance of recommendations over time.

Technologies Used in Conducting the Experiment

Tools and technologies have been employed in conducting the experiment and evaluating the performance of this model. For the hardware, a laptop with Intel Core i5, 6GB RAM, 256GB HDD storage, 32bit Windows 10 OS, and Python 3.12 as programming language was utilized. Jupyter Notebook used as a programming environment. Libraries such as NumPy, pandas, matplotlib, scikit-learn, scikit-image, TensorFlow also used. The details of the experimental parameters are presented in table 1.

Table 1: Experimental Parameters Setup

Parameter	Experimental setup
System specification	Corei5 with 6GB RAM, Hard disk HDD 256, windows 10
Sample size	10,000 records
Number of runs	Multiple
Epochs	50
Batch size	32
Activation Function	Loss
Optimizer	Adam
Loss function	Binary cross-entropy
Hyperparameter tuning method	Random
Training test split	80%, 10%, 10%
Libraries	NumPy, pandas, scikit learn, Matplotlib and seaborn
Embedding size	64
Execution time	200secs
Evaluation metric	Precision, Recall, F1-score, and MSE

Dataset

The primary Dataset used in this research is Movielens1M, publicly released by group lens research at the University of Minnesota. The Dataset was downloaded from [http://grouplens.org/dataset/Movie lens -1m](http://grouplens.org/dataset/Movie%20lens%201m). It consists of three tab-separated files: ratings.dat (user Id, Movie Id, Rating, Timestamp). Users.dat (User ID, Gender, Age occupation) Movies.dat (Movie ID, Title, Genres). In total the dataset contains 1,000,209 ratings from 6,040 users across 3,952 movies out of this a subset of 10,000 sample records (10,000 rows and 4 columns) was extracted for the development and evaluation of the proposed personalized movie recommendation model. The raw data were collected and processed to include information related to users, movies, explicit feedback, implicit feedback, and contextual information. The processed data were cleaned, transformed, and converted into comma-separated values (csv) format to

make them suitable for machine learning and recommendation model development. The processed csv dataset was subsequently imported into Anaconda Jupyter Notebook, which served as the programming environment for data pre-processing, exploratory data analysis, feature engineering, model training, and evaluation. The final dataset consists of multiple records, with each row representing an individual user movie interaction, together with its corresponding features and target label. The dataset incorporates explicit feedback, such as user ratings, likes, dislikes, reviews, and favourites, which directly indicate a user's opinion or preference for a movie. It also incorporates implicit feedback, including viewing search history, watch duration, in addition, contextual information is included to capture the circumstances under which a user interacts with a movie. Such contextual information may include time of interaction. These contextual features help the model

understand that a user's preference may vary depending on the situation in which the movie is accessed. The combination of explicit feedback, implicit feedback, movie characteristics, user information, and contextual information provides a more comprehensive representation of user preferences. This enables the proposed recommendation model to learn not only what movies users have previously liked or watched, but also how and under what circumstances they interact with movies. Consequently, the dataset provides an appropriate foundation for developing a personalized recommendation model capable of generating more relevant and context-aware movie recommendations. The dataset structured appropriately to suit the model. Furthermore, the processed data randomly divided into training and testing sets using train/test split to ensure unbiased model evaluation. Model training and validation: Employs supervised learning with an 80:10:10 train-test-validation split. Evaluation metrics: Four metrics were used precision, recall, F1-score, and MSE to provide a comprehensive assessment of the models ability.

Evaluation Metrics

Precision is an essential evaluation metric used to measure the effectiveness of a model in accurately identifying true positive. It defined as the ratio of correctly predicted positive instances to the total number of instances predicted as positive by the model. A high precision score indicates a low rate of false positives, meaning that samples rarely misclassified as, thereby enhancing the reliability of the model.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (3)$$

Recall, also referred to as sensitivity or the true positive rate, measures the ability of a model to correctly identify all actual instances within the dataset. It defined as the ratio of correctly predicted cases to the total number of actual. A high recall score indicates that the model effectively detects most instances, minimizing the occurrence of false negatives.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (4)$$

The F1-score is a composite evaluation metric that combines precision and recall to provide a balanced measure of a model's performance. It is defined as the harmonic mean of precision and recall, thereby considering both false-positive and false-negative predictions. The F1-score is particularly

useful when there is a need to achieve a balance between correctly identifying positive instances and minimizing missed detections.

$$F1 \text{ Score} = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (5)$$

RESULTS AND DISCUSSION

This section discusses the result of the entire experiment. The study utilizes a CNN model based on movie recommendation task. The model trained to incorporate explicit, implicit and contextual information among the samples enabling it to effectively recommend movies that align with users' preference.

Result

As shown in Figure 2, the recommender system demonstrated progressive improvement in its performance during training. The continuous improvement in performance across the training epochs suggests that the model effectively learned the underlining patterns and relationships within the dataset. The evaluation results shows that the recommender system achieved a precision of 86.25%, indicating that a high proportion of the recommendations generated by the system were relevant to the users. The recall of 98.76% demonstrates the systems' strong ability to identify actual relevant items and provide recommendations that align with users' preferences. This is particularly important in recommender system because identifying relevant items contributes to improved users' satisfaction and recommendation quality. The model achieved an F1-score of 92.08%, indicating a good balance between precision and recall. In addition, the MSE of 0.0845 indicates a relatively low prediction error suggesting that the systems predicted values were close to actual values in reasoning. Overall, the recommender system demonstrated strong performance capability across the reported evaluation metrics particularly in terms of recall, with 98.76% of relevant movies correctly identified. The F1-score of 92.08% confirms that the system provides a satisfactory balance between precision and recall. Therefore, based on the reported evaluation metrics, the recommender system shows promising performance to provide accurate and relevant recommendations to the users.

Table 2: Performance Metrics Results

Performance Metrics	Result (%)
Precision	0.8625
Recall	0.9876
F1 score	0.9208
MSE	0.0845

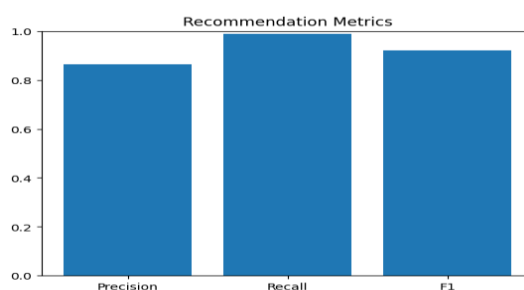


Figure 2: Recommendation Metrics

Comparison between Benchmark Study and Proposed Study

The benchmark system, based on the work of yang Gao, Hong Zhang & Haonan Cui (2025) evaluated recommendation performance using Precision and Recall, reporting

improvements of 0.8108 and 0.6904 respectively. These metrics assess the ability of the recommender system to identify and rank relevant movies. In contrast, the proposed system was evaluated using precision, recall, F1 score and

MSE as presented in table 2. The proposed system achieved a precision of (0.8625) a recall of (0.9876) an F1 score of (0.9208) an MSE of (0.0845). These metrics assess the ability of the recommender system to identify and rank relevant movies as well as measure the accuracy of predicted user ratings. The low error value obtained by the proposed model indicate high prediction accuracy. Furthermore, the proposed system incorporates explicit feedback, implicit feedback, and contextual information, together with a substantially larger training dataset (approximately 10,000 records from the MovieLens dataset). This proves that the proposed system has

outperform the existing system in terms of accuracy and recommendation. These enhancements produce more personalized and relevant movie recommendations based on user preference than systems relying on browsing behaviour. The proposed system utilized a CNN hybrid (explicit & implicit) and contextual information model, which addressed the shortcomings of the benchmark approach. Through the incorporation of user feedback and contextual information the hybrid CNN model improved accuracy prediction while enabling faster relevant movie recommendation as presented in table 3.

Table 3: Performance Metrics Comparison

Metric	Yang Gao, Hong Zhang & Haonan Cui (2025)	Proposed System
PRECISION	0.8108	0.8625
RECALL	0.6904	0.9876
F1 SCORE	Not reported/available	0.9208
MSE	Not reported/available	0.0849

CONCLUSION

Through appropriate model configuration and systematic training, the proposed recommender system achieved promising movie recommendation performance on the dataset used in the study. The results indicate that the model achieved a precision of 86.25%, recall of 98.76%, F1-score of 92.08%, and MSE of 0.0845, demonstrating a strong ability for recommendations while maintaining a relatively low prediction error. The high recall is particularly important because it shows that the model successfully identified the majority of actual movies liked by the users. The findings also suggest that a relatively simple architecture can achieve effective recommendation performance without necessarily relying on highly complex model structures. This indicates that increasing architectural complexity does not always result in proportionally better performance, particularly when computational resources and dataset size are limited. Therefore, achieving an appropriate balance between recommendation performance, computational requirements, and model complexity is important when selecting an architecture for recommender system. However, despite achieving this promising result the proposed recommender system may still be affected by data sparsity, cold start problem, and limited generalizability. Future studies should utilize larger and more diverse datasets containing, directors, actors, and other metadata to enhance the model's learning capability. Overall, the results demonstrate that proper model design, systematic training, and comprehensive evaluation are essential for developing a reliable and efficient recommender system model.

REFERENCES

- Gao, Y., Zheng, H., & Cui, H. (2025). User preference modeling for movie recommendations based on deep learning. *Scientific Reports*, 15(1), 1–16. <https://doi.org/10.1038/s41598-025-00030-5>
- Liu, C. (2025). *ScienceDirect. Procedia Computer Science*, 266, 32–40. <https://doi.org/10.1016/j.procs.2025.08.005>
- Z. Zhang, Y. Gu, B. A. Plummer, X. Miao, J. Liu, and H. Wang, Movie genre classification by language augmentation

and shot sampling, in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2024, pp.

Z. Cai, H. Ding, J. Wu, Y. Xi, X. Wu, and X. Cui, Multi-label movie genre classification based on multimodal fusion, *Multimedia Tools Appl.*, vol. 83, no. 12, pp. 3682336840, Jul. 2023.

Ricci, F., Rokach, L., Shapira, B., Kantor, P.B., 2015. *Recommender Systems Handbook*. Springer-New York Heidelberg Dordrecht London. 978-1-4899- 7636-9. <https://doi.org/10.1007/978-1-4899-7637-6> and <https://doi.org/978-1-72818337-4>

Stakhiyevich, P., & Huang, Z (2019). Building user profiles based on user interest and preference for recommender systems.

Zhou, L., Xiong, C., Deng, N., Shen, L. (2020). Personalized Recommendation Algorithm Based on User Preference and User Profile. In: Barolli, L., Amato,

Y. Shao and N. Guo, recognizing online video genres using ensemble deep convolutional Learning for digital media service management, *J. Cloud Computer.*, vol. 13, no. 1, pp.117, May 2024.

Feng, L., Zhao, Q., & Zhou, C (2019). Improving Performance of Top-N Recommendation with co-clustering method. *Expert Systems with applications*, *Science Direct*. 143(2020) 113078.

C. Channarong, C. Paosirikul, S. Maneeroj, and A. Takasu, "HybridBERT4Rec A Hybrid (Content-Based Filtering and Collaborative Filtering) Recommender System Based on BERT," *IEEE Access*, vol. 10, pp. 56193–56206, 2022, <https://doi.org/10.1109/ACCESS.2022.3177610>

Y. Yan and H. Xie, Collaborative Filtering Recommendation Algorithm Based on User Preferences, *J. Phys. Conf. Ser.*, vol. 1549, no. 3, 2020, <https://doi.org/10.1088/1742-6596/1549/3/032147>

