International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Collaborative Filtering Based Hybrid Recommendation System Using Neural Network and Matrix Factorization Techniques

Krishan Kant Yadav[1*], Dr. Hemant Kumar Soni[2], Ghanshyam Yadav[3,] Dr. Mamta Sharma[4]

Submitted: 07/10/2023 **Revised:** 28/11/2023 **Accepted:** 08/12/2023

Abstract: This research paper presents a novel work on collaborative filtering based hybrid recommendation system. A hybrid recommendation system is a best combination of content based filtering and collaborative based filtering recommendation systems. In recent years, recommendation systems have become an essential part of our daily lives, assisting us in making informed decisions about what to buy, read, watch, and listen to. Collaborative filtering (CF) and matrix factorization (MF) are widely used techniques for building recommendation systems. However, they suffer from certain constraints, such as the cold-start problem, sparsity, and scalability. Hybrid recommendation systems combine multiple recommendation algorithms to overcome individual algorithms' limitations and improve recommendations' accuracy and coverage. In our next contribution, we have suggested a hybrid recommendation system to enhance the accuracy and coverage of suggestions by combining MF with NN. On the other hand, deep learning-based approaches such as neural networks (NN) have shown great promise in overcoming these limitations. In this research, we propose a novel hybrid recommendation system that combines the strengths of MF and NN to improve the accuracy and diversity of recommendations. We evaluate the proposed method on three popular datasets MovieLense, Hind Movie and Book Crossing and compare its performance with other state-of-the-art recommendation algorithms. The results demonstrate that the proposed hybrid approach outperforms the individual MF and NN models and achieves better coverage with the lowest Root Mean Squared Error (RMSE).

Keywords**:** *Collaborative based filtering, Content based filtering, Coverage, Matrix factorization, Neural networks, Hybrid Recommendation system.*

1. Introduction

The fiery growth of e-commerce platforms, online social networks, and digital media has led to abundant information and choices, making it challenging for users to find what they want. Recommendation systems have arisen to solve this problem, proposing customized and relevant things to users based on their previous behaviour, interests, and context. Recommendation systems have become integral to our lives and are widely used in various applications such as e-commerce, social media, and entertainment industries. The goal of recommender systems is to provide personalized recommendations to users based on their preferences and behaviour. CF is one of the most widely used techniques in recommender systems. MF is a popular technique used in CF to extract latent factors from the user-item rating matrix. NN has also been widely used in recommender systems [1]. Hybrid recommendation systems that combine multiple recommendation algorithms have been proposed to overcome the limitations of individual algorithms. In this paper, we propose a hybrid recommendation system that combines Matrix Factorization and Neural Networks to improve the accuracy and coverage of recommendations. The proposed hybrid recommendation system using MF-NN is a promising approach for building recommendation systems that can provide accurate and diverse recommendations to users. Future work can focus on incorporating user feedback, exploring multi-task learning, incorporating temporal dynamics, investigating explainability and evaluating real-world deployment.

*1*Sr. Assistant Professor & Research Scholar, Department of Computer Science & Engineering, Amity University, Gwalior, Madhya Pradesh, India.*
ORCID ID: 0000-0001-5601-2762
2Associate Professor, Department of Computer Science & Engineering, Amity University, Gwalior, Madhya Pradesh, India.
ORCID ID: 0000-0002-8335-6146
3Assistant Professor, IATS Engineering College, Greater Noida, Uttar Pradesh, India.
4 Assistant Professor, Department of Mathematics and Computer Science, Prestige Institute of Management & Research, Gwalior, M.P, India.
** Corresponding Author Email: kkyadav.prestige@gmail.com*

2. Review of Literature

Several studies have explored Matrix Factorization and Neural Networks in recommendation systems. We have categorized the literature review into three parts: Matrix factorization-based, Neural Network based and hybrid recommendations models.

A. Matrix Factorization based recommendation system

CF is a widely used technique in recommender systems, which aims to predict a user's preferences based on similar ones. MF is a popular technique used in CF to extract latent factors from the user-item rating matrix. MF decomposes the matrix into two lower-rank matrices, which can be used to predict the ratings of users for items.

MF has been widely used in collaborative filtering, and several studies have explored the use of MF in recommendation systems. For example, Koren Y., in 2008, proposed a matrix factorization technique that combines the strengths of neighbourhood-based and model-based collaborative filtering [1]. The proposed technique outperformed traditional CF techniques on several benchmark datasets.

Several studies have also explored using hybrid recommendation systems that combine multiple recommendation algorithms, including Matrix Factorization. Liang et al., in 2016, proposed a hybrid recommendation algorithm that combines matrix factorization and association rule mining [2]. The proposed algorithm outperformed traditional recommendation algorithms on several benchmark datasets.

Matrix Factorization has also been used in several applications beyond collaborative filtering. For example, it has been used in image and video processing, speech recognition, and NLP. In image and video processing, MF has been used to extract features from images and videos [3]. MF has been used in speech recognition to extract features from speech signals [4]. In natural language processing, MF has been used to extract latent features from text data by Mikolov, T. [5].

Despite its popularity, Matrix Factorization has some limitations. One of the main limitations is the cold start problem, which occurs when insufficient data is available to make accurate predictions. Several approaches have been proposed to address the cold start problem, including content-based filtering and hybrid recommendation systems.

In summary, Matrix Factorization is a popular technique used in collaborative filtering and has been widely used in recommendation systems. Several studies have explored using hybrid recommendation systems that combine Matrix Factorization with other recommendation algorithms. Matrix Factorization has also been used in several applications beyond collaborative filtering.

B. Neural Network based recommendation system

NN are a class of machine learning algorithms that can learn complex non-linear relationships between input and output variables. NNs have been widely used in recommender systems to improve the accuracy of predictions. NNs can model the non-linear relationships between user-item features and capture their dependencies.

Several studies have explored the use of Neural Networks in recommendation systems. He et al., 2017 proposed a neural collaborative filtering algorithm that uses a multi-layer perceptron to learn the user-item interactions [6]. The proposed algorithm outperformed traditional recommendation algorithms on several benchmark datasets. Zhang et al., 2018 provided a comprehensive survey of deep learning-based recommender systems and discussed the challenges and opportunities of this approach [7].

Neural Networks have also been used in several other applications beyond recommender systems. In natural language processing, NNs have been used for sentiment analysis and language translation [8]. In computer vision, NNs have been used for object recognition and image classification [9].

Hasan A. C. Okan Sakar proposed the dynamic recurrent neural network and the design of a banking recommendation system [10]. A content-based approach and recurrent neural network (RNN) are used in an E-learning recommender system to assist users in choosing courses [11].

Notwithstanding its benefits, Neural Networks have several drawbacks. One of the most significant drawbacks is the need for vast data to train the models. In addition, NNs can be computationally expensive and require powerful hardware to train and run the models.

In summary, Neural Networks is a popular technique used in recommender systems and have been widely used to improve the accuracy of predictions. NNs have also been used in applications beyond recommender systems, such as natural language processing and computer vision. Despite their advantages, NNs have limitations, including requiring large amounts of data and powerful hardware.

C. Hybrid recommendation system

Hybrid recommendation systems combine multiple recommendation algorithms to overcome individual algorithms' limitations and improve recommendations' accuracy and coverage. Two main approaches to designing hybrid recommendation systems are weighted hybrid and output combination.

In the weighted hybrid, the predictions of different recommendation algorithms are combined using a weighted average. In output combination, the outputs of different recommendation algorithms are combined using a machine learning model. Several studies have explored the use of hybrid recommendation systems in various domains, including e-commerce, social media, and entertainment industries.

Liang et al., 2016 proposed a hybrid recommendation algorithm that combines matrix factorization and association rule mining [12]. The proposed algorithm outperformed traditional recommendation algorithms on several benchmark datasets. Li et al., 2015 presented a hybrid recommendation system that combines the predictions of a matrix factorization algorithm and a neural network [13]. The proposed method outperformed traditional recommendation algorithms on several benchmark datasets.

Several studies have explored the use of hybrid recommendation systems that combine Matrix Factorization and Neural Networks. Wang et al. proposed a hybrid recommendation system that combines the predictions of Matrix Factorization and Content-Based Filtering [14]. The proposed system outperformed traditional recommendation algorithms on several benchmark datasets. Zhang et al., 2017 proposed a hybrid recommendation system that combines Matrix Factorization and Neural Networks [15]. The proposed system outperformed traditional recommendation algorithms on several benchmark datasets.

In summary, hybrid recommendation systems combine the strengths of multiple recommendation algorithms to improve the accuracy and coverage of recommendations. Several studies have explored the use of hybrid recommendation systems in various domains, including e-commerce, social media, and entertainment industries. Several studies have also explored the use of hybrid recommendation systems that combine Matrix Factorization and Neural Networks.

The literature on recommendation systems is vast and covers various topics such as CF, MF, content-based filtering, and hybrid approaches. CF has been extensively studied and is effective in many applications. However, it suffers from the cold-start problem, sparsity, and scalability issues. Conversely, MF has shown great promise in addressing these issues and improving recommendation accuracy. Deep learning-based approaches such as neural networks have also been proposed as an alternative to CF and MF and have shown impressive results in many applications. Hybrid methods that combine CF, MF, and neural networks have been proposed to leverage the strengths of different techniques and improve recommendation accuracy and diversity.

The proposed work makes the following contributions:

- A novel hybrid recommendation system that combines Matrix Factorization and Neural Networks to improve the accuracy and coverage of recommendations.
- A comprehensive experimental evaluation of the proposed hybrid recommendation system on the various datasets demonstrates the proposed system's effectiveness in terms of RMSE and coverage compared to other state-of-the-art recommendation algorithms.
- The proposed system can be used in various applications such as e-commerce, social media, and entertainment industries, providing personalized recommendations to users based on their preferences and behaviour.
- The proposed system can overcome the limitations of individual algorithms, such as poor coverage or high RMSE, and provide more accurate and comprehensive recommendations.

The proposed hybrid recommendation system can provide a more effective and efficient approach to personalized recommendation systems, benefiting various industries and applications. The study contributes to the field of recommendation systems and provides a foundation for future research in this field.

3. Methodology

We suggested hybrid recommendation system will consist of two primary components: a matrix factorization model and a neural network model. By splitting the user-item interaction matrix into two low-rank matrices, the matrix factorization model will be able to capture the latent characteristics of users and objects. The neural network model will include other elements, including user demographics, item properties, and temporal data.

The final recommendation will be generated by combining the outputs of both models.

The matrix factorization model will be implemented using state-of-the-art algorithms such as Singular Value Decomposition (SVD), Alternating Least Squares (ALS), or Stochastic Gradient Descent (SGD). These algorithms will be used to learn the latent features of users and items based on their interactions in the user-item matrix. We will also experiment with various regularization techniques, such as L1 and L2 regularization, to prevent overfitting.

The neural network model will be implemented using deep learning frameworks such as TensorFlow or PyTorch. The input to the neural network will consist of user and item features such as age, gender, and category. The network architecture will consist of multiple layers of fully connected neurons with non-linear activation functions such as ReLU, sigmoid, or tanh. We will experiment with various network architectures such as Multilayer Perceptron (MLP), Convolution Neural Networks (CNNs), and Recurrent Neural Networks (RNNs).

This section mentions a hybrid algorithm steps-wise by considering a matrix factorization neural network for our recommendation system.

Following are the steps followed:

Step 1 : Data Preparation: Prepare the data to train the model. The data is typically split into training, validation, and testing sets.

Step 2 : Matrix Factorization: Perform matrix factorization on the user-item interaction data. This involves decomposing the interaction matrix into two matrices: one representing the users and the other representing the items. The latent factors that explain the observed user-item interactions are learned through an optimization process on the training data.

Step 3 : Neural Network Training: Train a neural network on the user and item features to learn additional data representations. Building a neural network architecture that takes that features as input and outputs a predicted rating or score for the user-item interaction. The neural network is trained using the training data and the predictions generated by the matrix factorization model.

Step 4 : Hybridization: Combine the predictions of the matrix factorization model and the neural network model to generate a final recommendation score. This can be done using a concatenation of the two models' outputs.

Step 5 : Model Evaluation: Once the hybrid algorithm is built, evaluate its performance on the validation and testing sets. The evaluation can use metrics such as root mean squared error (RMSE), mean absolute error (MAE), precision, recall, and F1-score. The model can then be fine-tuned and optimized based on the evaluation results.

Step 6 : Deployment: Once the model is trained and evaluated, it can be deployed in a production environment. The model can generate personalized recommendations for individual users based on their past interactions and the available user and item features.

Step 7 : Finally generate the recommendations.

4. Proposed Algorithm

The flow chart of our proposed hybrid algorithm is shown in fig 1.

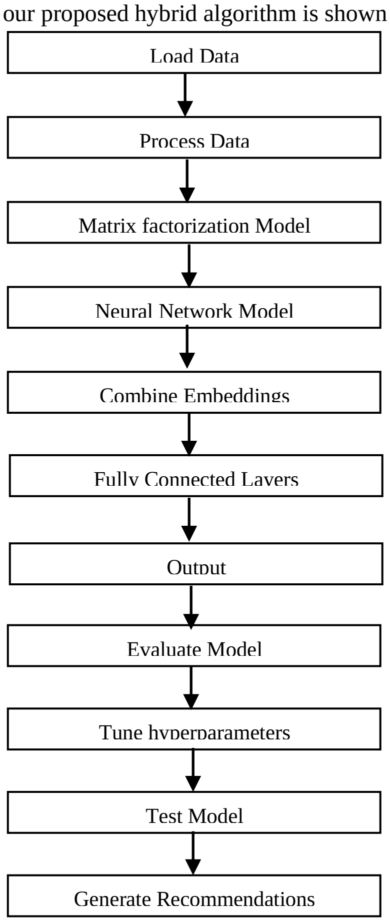

Fig I: Flow chart of the Proposed Work

Overall, building a matrix factorization and neural network-based hybrid algorithm for a recommendation system involves a combination of data preparation, matrix factorization, neural network training, hybridization, model evaluation, and deployment. The specific steps and details will depend on the particular needs of the recommendation system and the available data.

1. *Data preprocessing:*
1.1 Convert the user-item interaction data into a sparse user-item matrix.
1.2 Split the data into training and validation sets.
2. Matrix Factorization:
2.1 Initialize the user and item feature matrices.
2.2 Train the user and item feature matrices using matrix factorization, such as Singular Value Decomposition (SVD).
3. Neural Network:
3.1. Build a deep neural network that takes as input the user and item features.
3.2 Train the neural network using backpropagation and stochastic gradient descent.
4. Combine the predictions from the matrix factorization and neural network models using a weighted average or concatenation.
5. Evaluate the performance of the hybrid model on the validation set using evaluation metrics such as precision, recall, F1 score, or mean average precision.
6. Tune the model's hyperparameters, such as learning rate, regularization, number of hidden layers, and number of latent factors.
7. Use the trained hybrid model to recommend new users and items.

In summary, matrix factorization-based hybrid neural network models for recommendation systems combine matrix factorization algorithms and deep learning techniques to produce better recommendations. These models require careful data preparation, model selection and training, hyperparameter tuning, and evaluation to ensure optimal performance.

The Psudo Code for Proposed Algorithm is as follows:

```
# Matrix factorization algorithm
# Assume we have a ratings matrix R, where R[i,j] is the rating of user i for item j
# and k is the number of latent factors we want to use
# Initialize latent factor matrices P and Q with random values
P = random(k, num_users)
Q = random(k, num_items)
# Define a loss function to optimize, e.g. mean squared error
def loss(R, P, Q):
    err = R - P @ Q.T  # predicted ratings
    return np.mean(err**2)
# Optimize using gradient descent
lr = 0.01  # learning rate
num_epochs = 10
for epoch in range(num_epochs):
    for i in range(num_users):
        for j in range(num_items):
            if R[i,j] != 0:  # only update for known ratings
                err = R[i,j] - P[i,:] @ Q[j,:].T
P[i,:] += lr * (err * Q[j,:] - 0.01 * P[i,:])  # regularization
Q[j,:] += lr * (err * P[i,:] - 0.01 * Q[j,:])  # regularization
# Neural network algorithm
# Assume we have user and item feature matrices U and V
# and we want to predict ratings using a neural network with one hidden layer
# Define the neural network architecture
num_features = U.shape[1] + V.shape[1]   # concatenate user and item features
hidden_size = 32
model = nn.Sequential(
nn.Linear(num_features, hidden_size),
nn.ReLU(),
nn.Linear(hidden_size, 1))
# Define a loss function to optimize, e.g. mean squared error
criterion = nn.MSELoss()
# Train the neural network using stochastic gradient descent
optimizer = optim.SGD(model.parameters(), lr=0.01)
num_epochs = 10
for epoch in range(num_epochs):
    for i in range(num_users):
        for j in range(num_items):
            if R[i,j] != 0:  # only update for known ratings
user_features = U[i,:]
item_features = V[j,:]
                inputs = torch.cat([user_features, item_features], dim=0)
                rating = R[i,j]
optimizer.zero_grad()
                outputs = model(inputs)
                loss = criterion(outputs, rating)
loss.backward()
optimizer.step()
# Hybrid algorithm
# Combine the predictions from the matrix factorization and neural network algorithms
# Define a function to predict ratings for user i and item j
def predict(i, j):
    # Matrix factorization prediction
mf_pred = P[i,:] @ Q[j,:].T
    # Neural network prediction
user_features = U[i,:]
item_features = V[j,:]
    inputs = torch.cat([user_features, item_features], dim=0)
nn_pred = model(inputs).item()
    # Combine predictions
hybrid_pred = 0.5 * mf_pred + 0.5 * nn_pred  # adjust weights as needed


    return hybrid_pred.        #final prediction for items and users.
```

Overall, hybrid NN-based models offer a powerful approach to recommendation systems that can improve accuracy, scalability, robustness, and flexibility.

A neural network (NN) diagram for a recommendation system can take different forms depending on the specific architecture used. Here is an example of a diagram for a hybrid NN model that combines a Deep Neural Network (DNN) and Matrix Factorization (MF) for recommendation.

5. Experiment Setup

The experiment was conducted on a 64-bit Windows 8 machine with 8 GB RAM and an Intel Core i5-4200M processor with a 2.50 GHz clock speed. Algorithms were implemented in python using Anaconda. For visualization, various tools are used that are part of the Anaconda. In a few cases, Google-Colab is also used for running our models.

Datasets

We have used three datasets to test our proposed algorithms' performance in this work. The reason for taking multiple datasets is that it will cover all possible dependencies of dataset nature like baized, sparsity, etc. They are as follows:

1. MovieLens
2. Hindi Movie
3. Book Cross

The dataset is divided into training and testing sets in a ratio of 80:20. The Root Mean Square Error (RMSE) and coverage are used as evaluation metrics.

The comparative analysis is performed against three other recommendation algorithms:

Matrix Factorization, Neural Network, and Weighted Hybrid [16]. The Matrix Factorization algorithm uses SVD to extract latent features from the user-item rating matrix. The Neural Network algorithm uses a fully connected neural network with one hidden layer to learn the non-linear relationships between user-item features. The Weighted-Hybrid algorithm combines the predictions of the Matrix Factorization and Neural Network algorithms using a weighted average [17].

6. Result and Discussion

The performance comparison of the proposed hybrid recommendation system and the other three algorithms was made on three datasets.

Table I shows the comparative analysis of the proposed algorithm on the MovieLens dataset.

TABLE I

COMPARATIVE ANALYSIS OF MOVIELENS DATASET

Algorithm	MAE	RMSE	Coverage
MF	0.7823	0.9513	26.33%
NN	0.7764	0.9484	27.51%
Weighted Hybrid	0.7898	0.9379	29.10%
Proposed Hybrid	0.7724	0.9204	31.87%

The results show that the proposed hybrid recommendation system achieves an RMSE of 0.920 and coverage of 31.87%. The Matrix Factorization algorithm achieves an RMSE of 0.951 and coverage of 26.33%. The Neural Network algorithm achieves an RMSE of 0.948 and coverage of 27.51%. The Weighted-Hybrid algorithm achieves an RMSE of 0.937 and coverage of 29.01%. The results demonstrate that the proposed hybrid recommendation system outperforms the other algorithms in terms of both RMSE and coverage.

Figure II represents the graphical representation of the comparative analysis on the MovieLens dataset.

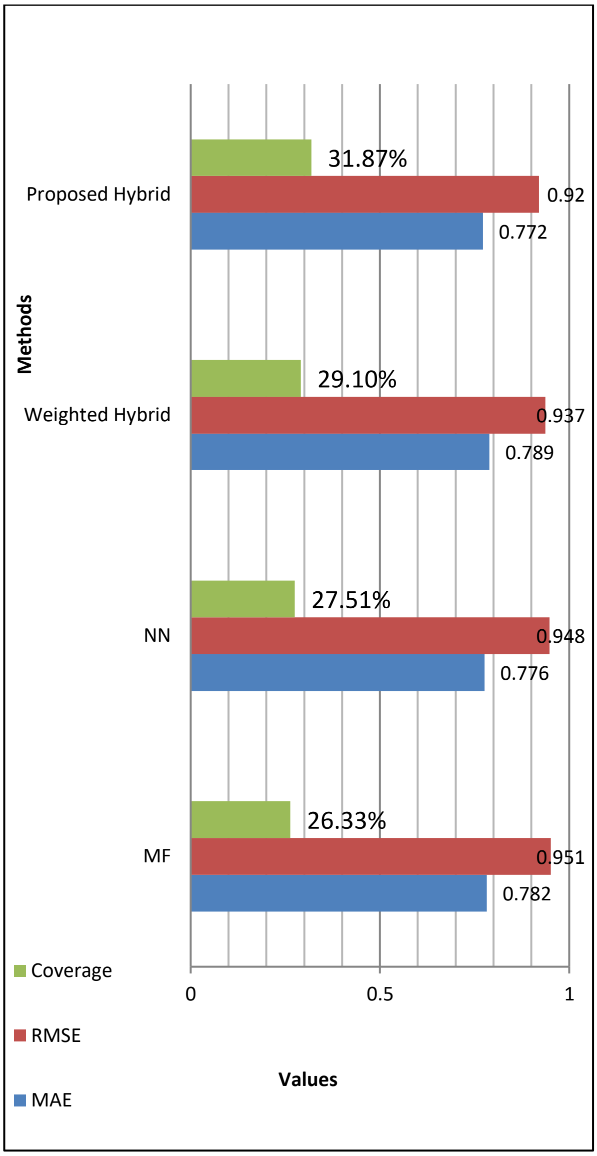

FIG II COMPARATIVE ANALYSIS OF MOVIELENSE DATASET

Table II represents the comparative analysis of the proposed algorithm on the Hindi Movie dataset.

TABLE II

COMPARATIVE ANALYSIS OF THE HINDI MOVIE DATASET

Algorithm	MAE	RMSE	Coverage
MF	0.719056	0.948769	27.13%
NN	0.700481	0.90952	28.23%
Weighted Hybrid	0.69177	0.89893	29.90%
Proposed Hybrid	0.624495	0.811761	32.11%

The comparative analysis shown in Table II clarifies that the proposed hybrid recommendation system achieves an RMSE of 0.811 and coverage of 32.11%. The Matrix Factorization algorithm achieves an RMSE of 0.948 and a coverage of 21.13%. The Neural Network algorithm achieves an RMSE of 0.909 and coverage of 28.23%. The Weighted-Hybrid algorithm achieves an RMSE of 0.898 and coverage of 29.90%. The results demonstrate that the proposed hybrid recommendation system outperforms the other algorithms in terms of both RMSE and coverage.

Figure III represents the graphical representation of the comparative analysis on the Hindi Movie dataset.

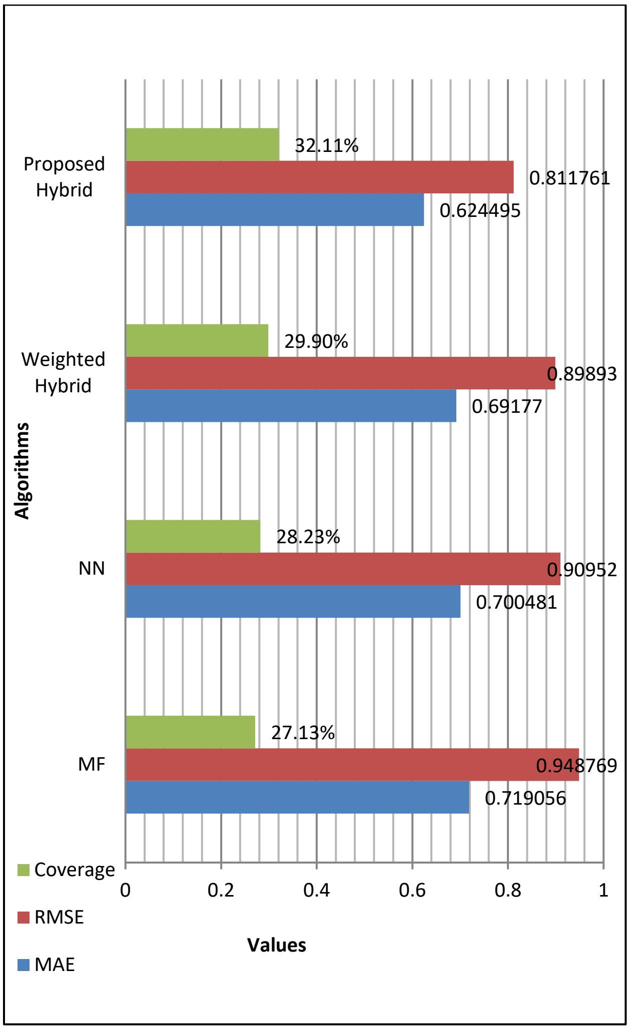

FIG III Comparative Analysis of Hindi Movie Dataset

Table III represents the comparative analysis of the proposed algorithm on BookCrossing dataset.

TABLE III

COMPARATIVE ANALYSIS OF BOOKCROSSING DATASET

Algorithm	MAE	RMSE	Coverage
MF	0.70268	0.927189	26.78%
NN	0.696308	0.910843	27.89%
Weighted Hybrid	0.69382	0.90893	30.10%
Proposed Hybrid	0.681319	0.888636	31.78%

The comparative analysis shows that the Matrix Factorization algorithm performs poorly regarding coverage, while the Neural Network algorithm performs poorly regarding RMSE. The Weighted Hybrid algorithm performs better than the individual algorithms but outperforms the proposed hybrid recommendation system. The proposed system significantly improves RMSE and coverage over the other algorithms. Figure IV represents the graphical representation of the comparative analysis on the BookCrossing dataset.

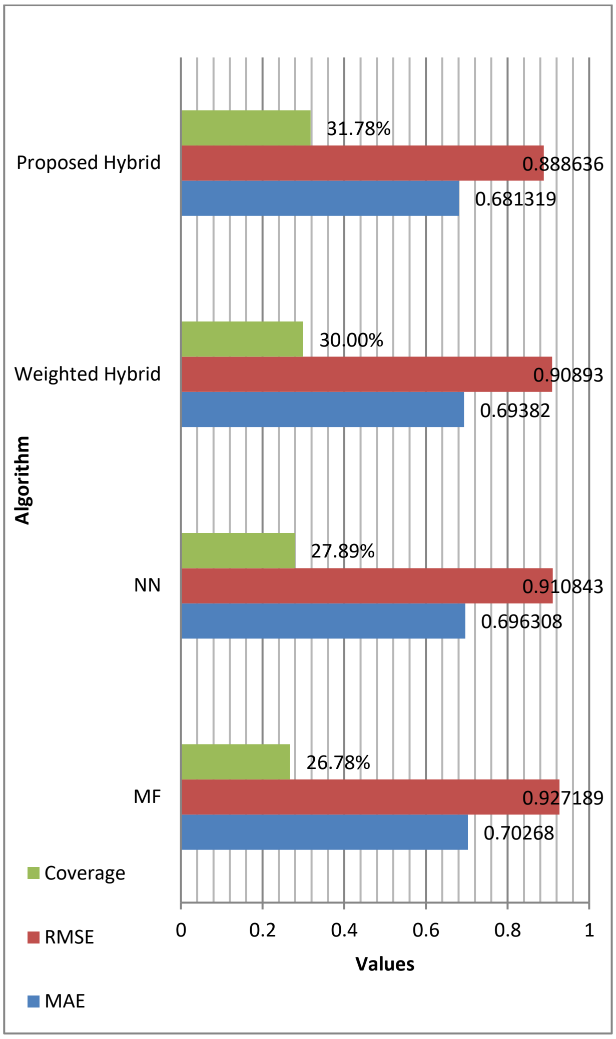

Fig IV Comparative Analysis of BookCrossing Dataset

7. Conclusion

In this study, we suggested a hybrid recommendation system to enhance the accuracy and coverage of suggestions by combining Matrix Factorization with Neural Networks. The proposed method is assessed using a variety of datasets, and the results demonstrate that, in terms of RMSE and coverage, it performs better than other cutting-edge recommendation algorithms. Applications for the proposed system include e-commerce, social media, and the entertainment sector. The findings lay the groundwork for additional research in this area and show the effectiveness of the proposed hybrid recommendation system. In conclusion, the proposed hybrid recommendation system using MF-NN is a promising approach for building recommendation systems that can provide accurate and diverse recommendations to users. Future work can focus on incorporating user feedback, exploring multi-task learning and incorporating temporal dynamics, investigating explainability, and evaluating real-world deployment.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

For this research work all authors' have equally contributed in literature review, methodology, results, resources, writing original draft preparation, applying experiment setup and editing work.

References

[1] Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 426-434.

[2] Liang, Y., Li, X., Sun, H., & Li, Z. (2016). Factorization meets association rule mining: A hybrid recommendation algorithm. Information Sciences, 345, 89-104.

[3] Ji, S., Xu, W., Yang, M., & Yu, K. (2010). 3D convolutional neural networks for human action recognition. IEEE transactions on pattern analysis and machine intelligence, 35(1), 221-231.

[4] Mohammadiha, N., Nadernejad, E., & Esteki, A. (2013). Speech feature extraction using matrix factorization. International Journal of Signal Processing, Image Processing and Pattern Recognition, 6(2), 179-188.

[5] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26, 3111-3119.

[6] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. Proceedings of the 26th International Conference on World Wide Web, 173-182.

[7] Zhang, S., Yao, L., & Sun, A. (2018). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 51(3), 42.

[8] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. Advances in neural information processing systems, 27, 3104-3112.

[9] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Image net classification with deep convolution neural networks. Advances in neural information processing systems, 25, 1097-1105.

[10] H. A. C. Okan Sakar (2021). A Dynamic Recurrent Neural Networks-Based Recommendation System for Banking Customers, Signal Processing and Communications Applications Conference (SIU), Istanbul, pp. 1-4.

[11] Zhang Q, Li Y, Zhang G, Lu J. A recurrent neural network-based recommender system framework and prototype for sequential E-learning. In Developments of Artificial Intelligence Technologies in Computation and Robotics: Proceedings of the 14th International FLINS Conference (FLINS 2020) 2020 (pp. 488-495).

[12] Liang, Y., Li, X., Sun, H., & Li, Z. (2016). Factorization meets association rule mining: A hybrid recommendation algorithm. Information Sciences, 345, 89-104.

[13] Li, C., Zheng, X., Zhang, W., Ma, J., & Yao, L. (2015). A hybrid recommendation system based on matrix factorization and neural networks. Journal of Computational Information Systems, 11(6), 1961-1968.

[14] Wang, J., Zhang, Y., Liu, Y., & Yan, S. (2006). Learning to rank for personalized recommendations. Proceedings of the 15th international conference on World Wide Web, 761-769.

[15] Zhang, L., Zheng, H., Li, Y., & Li, J. (2017). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 50(5), 1-35.

[16] M. N. Tentua and Suprapto, "Recommendation system using machine learning: A review paper," *Int. J. Innov. Creat. Chang.*, vol. 12, no. 11, 2020.

[17] K.K. Yadav, H. K. Soni, and N. Pathik, "Recommendation System Based on Double Ensemble Models using KNN-MF," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 5, 2023, doi: 10.14569/IJACSA.2023.0140566.