INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN MULTIDISCIPLINARY EDUCATION

ISSN (print): 2833-4515, ISSN (online): 2833-4531

Volume 03 Issue 12 December 2024

DOI: 10.58806/ijirme.2024.v3i12n18, Impact factor- 5.138

Page No. 1972-1980

Evaluating Machine Learning Approaches for Personalized Movie Recommendations: A Comprehensive Analysis

Achuthananda Reddy Polu[1], Bhumeka Narra[2], Dheeraj Varun Kumar Reddy Buddula[3], Hari Hara Sudheer Patchipulusu[4], Navya Vattikonda[5], Anuj Kumar Gupta[6]

[1]Senior SDE, Cloudhub IT Solutions
[2]Sr Software Developer, Statefarm
[3]Software Engineer, Elevance Health Inc
[4]Senior Software Engineer, Walmart
[5]Business Intelligence Engineer, International Medical Group Inc
[6]Oracle ERP Senior Business Analyst ,Genesis Alkali

ABSTRACT: Platforms and movie theatres provide a large range of movies that need to be filtered to each user's tastes. For this objective, recommender systems are a useful tool. This research presents a novel hybrid recommender system for personalized movie suggestions, which integrates content-based methods with collaborative filtering. This study develops a personalized movie recommendation system utilizing the MovieLens 1M dataset, comprising user ratings for a diverse set of movies. The research data undergoes separation into training segments that constitute 80% of the total sample while testing comprises 20% of the data. The evaluation framework incorporates multiple metrics which include F1-Score together with Precision and Root Mean Square Error (RMSE) as well as Mean Absolute Error (MAE) alongside Recall. A Multilayer Perceptron (MLP) model is employed for movie recommendations and compared to a Deep Neural Network (DNN). The outcomes show that a MLP model outperforms the DNN, achieving a lower RMSE of 0.99 and an MAE of 0.80, in contrast to the DNN's RMSE of 1.011 and MAE of 73.5. The training and validation loss trajectories show continuous progress and maintain minimal avoidable patterns. Future work will refine the model accuracy by performing hyperparameter tweaking and developing advanced feature extraction processes together with implementing distinct deep learning models for enhanced recommendation capability and user satisfaction.

KEYWORDS: Personalized Movie Recommendation, Movie Classification, Recommendation Systems, User Preferences, Machine Learning (ML), Movie Lens 1M data.

I. INTRODUCTION

Individualized recommendation functions holistically influence user experiences on multiple digital platforms across the entertainment sector and e-commerce platforms. Modern content discovery systems rely on personalized movie recommendations to become the cornerstone feature that helps users efficiently discover content from massive databases. Customized entertainment experiences emerge from systems that generate content recommendations using viewer historical behaviors along with rating histories and their preferred genres [1]. The rapid expansion of online content demands efficient recommendation systems for users to navigate an endless sea of available content toward personalized choices.

Types of custom recommendation systems employ machine learning (ML) algorithms to explore user feedback patterns alongside content properties within large datasets to develop precise recommendation options[2]. Users may now go through the vast amount of information on the internet and locate just the content that suits them due to movie recommender systems. Through content-based filtering and collaborative filtering alongside hybrid models these systems present personalized recommendations that increase user satisfaction. Users benefit from two types of recommendations based on collaborative filtering which discovers user behavior patterns to suggest content that matches other users' preferences alongside content-based filtering that recommends relevant past-viewed content[3]. Hybrid models utilize various recommendation techniques to produce more efficient results by resolving individual implementation barriers that each technique possesses.

The fundamental need for evaluating machine learning approaches in movie recommendation emerged due to fast technological progress alongside evolving user taste patterns[4]. Researchers investigate how different algorithms process extensive datasets while generating accurate suggestion results. The research examines different machine-learning approaches for movie recommendation systems while assessing their performance benefits and constraints for actual system deployment [5]. This research evaluates multiple machine learning models to help understand how future improvements in recommendation quality and accuracy will impact user experience within an expanding content-based society

A. Motivation and Contributions of the Study

Online movie platforms grew quickly which requires personalized recommendations to achieve better user experiences. The standard recommendation solutions struggle to identify multiple facets of user interests. The research focuses on boosting recommendation accuracy through sophisticated machine learning advances on the MovieLens 1M dataset in order to suggest movies that align with individual preferences. This paper's contributions are as:

- Developed a preprocessing pipeline to handle categorical and temporal data for personalized movie recommendations.
- Applied Min-Max normalization to preserve linear relationships in the dataset and improve model performance.
- Implement and compare the MLP and DNN model for movie recommendation system.
- When assessing the recommendation model, they used a full suite of performance indicators, including MAE, RMSE, Precision, Recall, and F1 Score.

B. Organization of the paper

The structure of the paper is as follows: A survey of the body of research on personalized movie recommendation systems is given in Section II. Section III provides specifics on the technique, which includes data pretreatment, model selection, and assessment measures. The experimental findings are shown in Section IV. The research is finally concluded in Section V, which provides a summary of the results and suggests future paths for boosting model performance and personalized recommendation systems.

II. LITERATURE REVIEW

This section reviews existing movie recommendation models, highlighting methodologies. Table 1 presents a comparative overview of the methodologies, datasets, performances, limitations, and proposed future directions for the referenced studies. Some of reviews are:

In This study, Ahuja Solanki and Nayyar (2019), python was used to build a movie recommender system that made use of the KaggleMovieLens dataset and the K-Means Clustering and KNN algorithms. The study explores machine learning concepts and recommendation systems, employing techniques like Collaborative Filtering and Content-Based Filtering. The model achieved 97% accuracy, demonstrating an effectiveness of ML in personalizing user experiences and optimizing recommendation processes[6].

In this study, Inan, Tekbacak and Ozturk, (2018) goal of this kind of system is to pair the built-in user model with the best possible product. Experimental setup is done using the MovieLens dataset, and methods are compared using MAE. When doing the assessment with 300 training users, the optimal average MAE score is 0.736. Additionally, with a duration of 2.34 seconds, the movie suggestion subtask is the quickest. Experiments demonstrate that adding the collaborative filtering strategy to the content information improves the overall system performance, and the suggested system outperforms the remaining research in the literature [7].

In this study, Banik and Hasan Hafizur Rahman (2018) that analyzing textual reviews manually is complex and time-consuming, necessitating automated systems like Sentiment Analysis (SA) to extract opinions and emotions. Despite extensive implementation in various languages, SA has not been explored for Bangla, which is growing due to Bangladeshi reviewers on the web. Using NB and SVM, this work creates a polarity recognition algorithm for Bangla movie reviews. Results demonstrate that when using stemmed unigram features, SVM achieves a slightly higher accuracy of 0.86 compared to NB [8].

In This study, Wang and Zhang, (2018) Gaussian kernel SVM and logistic regression model were used to analyze movie recommendation data. Comparing in-sample and out-sample errors under varying VC dimensions revealed overfitting reduction. The SVM model achieved 85% accuracy, improving to 93% with optimized VC dimensions, enhancing machine learning's role in predicting customer movie preferences effectively[9].

In this study, Rai and Mewada, (2017), sentiment analysis is conducted on the database of movie reviews found on IMDB. To determine the review's polarity (negative or positive), they look at the sentiment expression. Then, they extract and rank features, and last, they train our multilayer classifier to correctly identify the review. ML is used in this work to categorize movie reviews into positive and negative groups. The method's 99% accuracy is the best of its kind since it makes use of classification methods[10].

TABLE I. SUMMARY OF THE RELATED WORK ON PERSONALIZED MOVIE RECOMMENDATION USING MACHINE LEARNING

References	Methodology	Dataset	Performance	Limitations & Future Work
Ahuja, Solanki, and Nayyar (2019)	K-means clustering, KNN, CollaborativeFiltering, Content-BasedFiltering	MovieLens (Kaggle)	97% accuracy	Focused on accuracy; scalability and data sparsity not addressed.
Inan, Tekbacak, and Ozturk (2018)	Collaborative Filtering with Content-Based Augmentation, Pearson Correlation, Goal Programming Model	MovieLens	Best MAE: 0.736; Recommendation	Limited evaluation scope; reliance on content-based similarity.

			runtime: 2.34 seconds	
Banik and Hasan Hafizur Rahman (2018)	Sentiment Analysis using NB and SVM for polarity detection	Bangla movie reviews	SVM achieved 86% accuracy (stemmed unigram features)	Focused on Bangla language; lacks multilingual or cross-lingual analysis.
Wang and Zhang (2018)	Gaussian Kernel SVM, Logistic Regression, VC Dimension Optimization	Movie recommendation data	SVM improved from 85% to 93% accuracy with optimized VC dimensions	Limited to Gaussian kernel SVM; no exploration of hybrid approaches.
Rai and Mewada (2017)	Sentiment Analysis on IMDB reviews, Feature Extraction, Multilevel Classifier for Polarity Classification	IMDB reviews	99% accuracy for binary sentiment classification	Focused only on binary classification; limited exploration of nuanced sentiment classes.

III. METHODOLOGY

The personalized movie recommendation system in this study utilizes the MovieLens 1M dataset, consisting of ratings provided by users for a wide range of movies. The methodology starts with preprocessing the data by merging individual user ratings and applying label encoding to convert categorical features, such as user and movie IDs, into numerical format. The ratings are then organized into sequences based on their timestamp, ensuring that the user's interactions are presented chronologically for training the model. Min-max normalization is applied to rescale the data to a range between 0 and 1, ensuring the preservation of linear relationships among variables. After preprocessing, the data is split into training (80%) and testing (20%) sets. MAE, RMSE, Precision, Recall, and F1 Score are among the metrics used to evaluate the system's performance. These measures are used to determine how well the model can propose movies based on user preferences. The overall steps are shown in Figure 1.

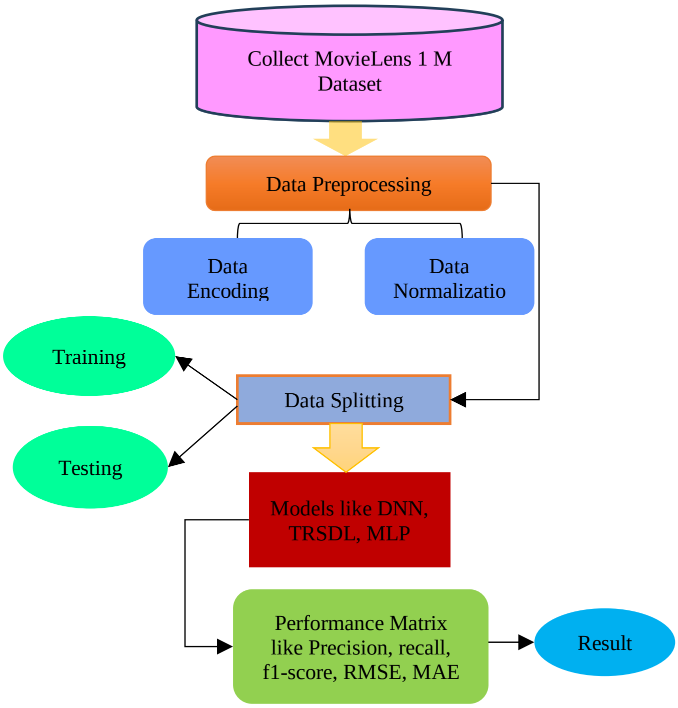

Fig. 1. Flowchart for Movie Recommendation System

This section provides a concise explanation of the procedures involved in creating a data flow diagram:

A. Data Collection

This study makes use of the MovieLens 1M dataset, which was gathered by the University of Minnesota's Group lens team. There are 6040 users' ratings of 3,883 films in the Movie Lens 1M dataset, which totals 1,000,209. The data includes user attributes, movie attributes, and movie ratings. User ID, movie ID, rating, rating time, and other related data are included in the rating information. The released movies based on year are shown in Figure 2.

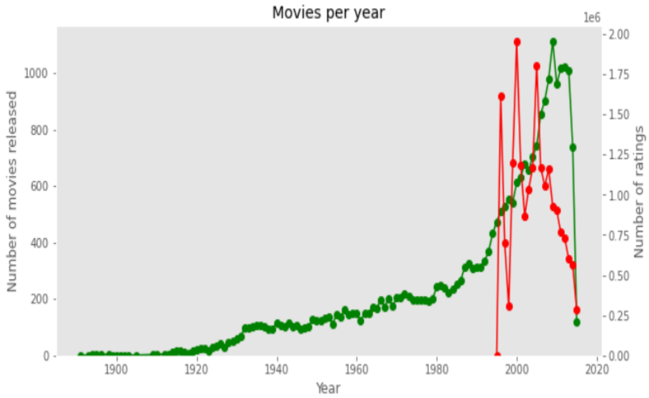

Fig. 2. FigLine Graph for the Number of Movies released

Figure 2 illustrates the historical trend of movie releases per year from 1900 to 2020. The green line (left y-axis) shows a gradual rise in movie production from 1900 to 1980, followed by a sharp increase peaking at around 1000 releases per year in the early 2000s. A decline was observed after 2010, with a steep drop around 2020, likely due to the COVID-19 pandemic. This trajectory highlights the growth of the film industry, its digital era expansion, and recent challenges.

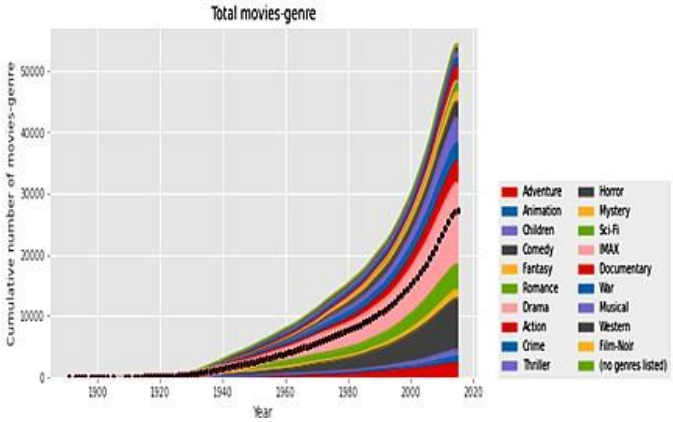

Fig. 3. Cumulative Number of Movies-generation

A cumulative number of films made in various genres from the early 1900s until 2020 is depicted in Figure 3, which offers insights into how the popularity of each genre has changed over time. As the film industry grew, the figure shows a steady rise in film production across all genres, with a notable acceleration around 1980.

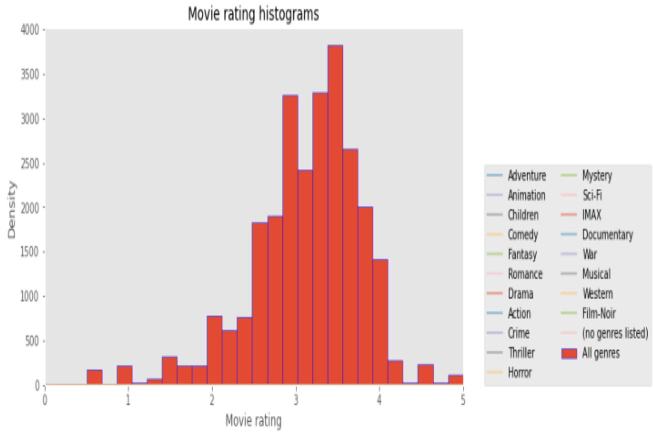

Fig. 4. Histogram of Movie rating

The Figure 4 histogram illustrates the distribution of movie ratings across various genres, offering insights into audience preferences and rating trends. The y-axis displays the density of ratings, while the x-axis provides movie ratings from 1 to 5. The red bars, representing all genres combined, exhibit a normal distribution pattern, peaking around a rating of 3 to 4. This indicates that most movies receive moderate to high ratings, with fewer films rated at the extremes (1 or 5). The histogram provides a comprehensive view of rating patterns, highlighting the central tendency and variability across different genres and audience evaluations.

B. Data Preprocessing

The preparation procedures that are necessary for our implementation. They leverage user evaluations from the Movie Lens dataset to build a recommendation system in our work. First, combine each user's rating data into the format needed for our transformer model's input needs. Afterward, they build a dictionary of movie and user IDs and arrange user interactions in chronological order. The training model is fed a subsequence of all user interactions that have been sorted by interaction timestamp.

C. Data encoding

Label encoding techniques are used to transform the numerical data kinds from the categorical data types. By giving each category a distinct number label, label encoding transforms categorical data into numerical data. For the purpose of converting categorical data to numerical form, the sklearn package includes a library named LabelEncoder.

D. Data Normalization

Using the Min-Max normalization approach, the data is made to normal. The data is rescaled from zero to one using the Min-Max normalization approach, which effectively applies a linear transformation to the collected data in this study. The linear connections among the variables in the data set may be maintained with the aid of this approach. The Min-Max normalization approach is mathematically described in Eq. (1) [11].

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Where,

cap X sub n o r m denotes normalized_data, cap X sub m i. n represents minimum rescaling_value, and cap X sub m a. x inindicatesx escaling_value.

E. Data Splitting

The dataset was split into a training dataset and a testing dataset once the data had been preprocessed. Eighty percent of the Movie lens dataset was in the training dataset, while the other twenty percent was testing data.

F. Classification Machine Learning Models

As a fundamental architecture of neural networks Multilayer Perceptron's (MLPs) find extensive applications in collaborative filtering systems [12]. The complete coupling between neurons in MLPs enables the network to develop complex non-linear data representations. In this context, the interaction between users and items is modeled by concatenating their embedding vectors. CapP_u and Q_i, as x_{ui} = $[p_u, Q_i]$. This concatenated vector is passed through several hidden layers with non-linear activation functions, typically $ReLU$, as follows (2):

$$h_1 \; ReLU(W_1\, x_{ui} + b_1)$$

$$h_2 = \text{ReLU}\ (W_2\, h1 + b_2)$$

$$h_L = ReLU(W_L h_L - 1 + b_L)$$

Finally, the output layer applies a linear transformation followed by a sigmoid function to predict the rating: $r\hat{}_{ui} = \sigma(W_o\, h_L + b_o)$. MLPs generate more precise product recommendations because they understand complex behavioral patterns and product relationships in nonlinear ways [13].

G. Performance Matrix

An analysis of the performance categorization indicators was done in this study to gauge the model efficiency and effectiveness, as well as establish their risk threshold. The following performance indicators are formulated below:

- **Mean absolute error (MAE):** The MAE provides an error assessment by studying absolute value differences between observed and predicted outcomes [14]. It is shown in Equation 5:

$$MAE = \frac{1}{|\hat{R}|} \sum \widehat{r_{u,i}} \in \hat{R} \left| r_{u,i} - \widehat{r_{u,i}} \right|$$

where $\hat{R}$ is the set of predictions

- **Root Mean Square Error (RMSE): In order to compute negative numbers,** RMSE performs a square root operation on squared errors instead of calculating absolute value. RMSE additionally severely penalizes more substantial mistakes [15].RMSE is calculated as follows in Equation 6:

that $$RMSE = \sqrt{\frac{1}{|\hat{R}|} \sum \widehat{r_{u,i}} \in \hat{R}\left(r_{u,i} - \widehat{r_{u,i}}\right)^2}$$

Precision: The statistical measurement of precision represents the frequency at which correct categorical assignments are made within all prediction evaluations. The equation for it is provided by (7):

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$

Where,

True Positive (TP): TP occurs when both the actual and projected values are positive.

False Positive (FP): A negative actual value with a positive anticipated value is an example of FP.

Recall: The rate at which samples are properly categorized for a certain class type, given all instances of that class type, is known as recall. The formula (8) is used to compute it.

$$\text{Recall} = \frac{TP}{TP+FN}$$

where,

False Negative (FN): A situation known as FN occurs when the expected value is negative while the actual value is positive.

F1 score: The F1 score, which is a harmonic mean of the recall and precision scores, is sometimes called the F measure. There is no such thing as a modest precision or recall value that won't get the F measure closer to it. The following equation (9) describes the F1 score:

$$F1 = \frac{2*(\text{precision}*\text{recall})}{\text{precision}+\text{recall}}$$

Are following evaluation parameters evaluate the model efficiency and determine the best model accuracy for movie recommendation system.

IV. Result Analysis And Discussion

A desktop computer running "MicrosoftWindows 10" with 16 GB of RAM, an Intel Core i7 CPU running at 2.2 GHz, and other specifications are used to conduct the following experiment. The experimental results for Personalized Movie Recommendations in a Machine Learning context are presented in this section. This experiment uses the Movie Lens 1M dataset for train the models. This study applies the MLP model for a movie recommendation system and compares it with DNN[16], as shown in Table III. The following Table II provides the performance of the proposed model performance on the same dataset.

TABLE II. Outcome of MLP model on the Movie lance 1M dataset For Movie Recommendation System

Performance Matrix	Multilayer perceptron (MLP)
RMSE	0.9927
MAE	0.8007
Precision	0.5838
Recall	0.4723
F1-Score	0.5222

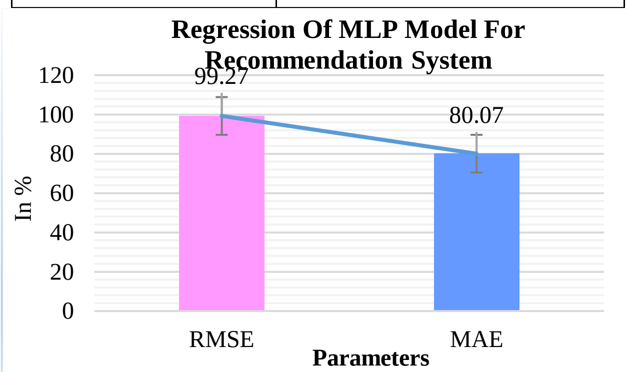

Fig. 5. Bar Graph of MLP regressor

Figure 5 shows the performance of the MLP model, revealing its predictive accuracy in terms of error metrics shown in Figure 6. The RMSE is 0.9927, indicating the average magnitude of prediction errors, with a higher emphasis on larger errors due to its quadratic nature. Meanwhile, the MAE is 0.8007, reflecting an average absolute deviation between forecasted and actual values. These metrics suggest that while the model demonstrates reasonable performance, there is scope for refinement, such as feature engineering or hyperparameter tuning, to reduce errors and improve prediction accuracy.

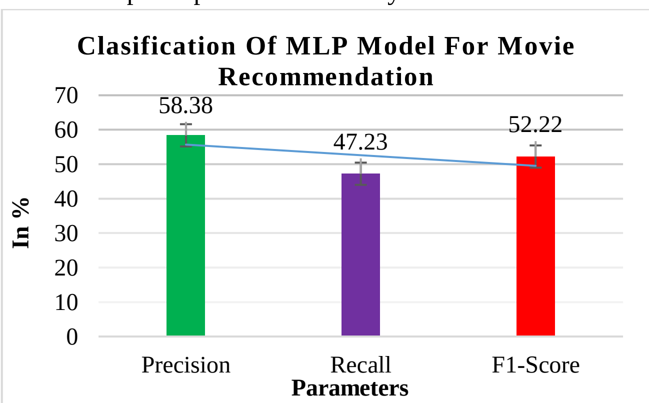

Fig. 6. Bar Graph for classification of MLP

Figure 6 shows performance of the MLP model indicates moderate efficacy in its classification task, as reflected in its performance metrics. With a precision of 0.5838, the model was able to accurately predict favorable outcomes in 58.38% of the cases. It seems that there were some constraints in collecting all relevant examples, since the recall value of 0.4723 indicates that the model correctly detected 47.23% of the real positive occurrences. The F1-score, which harmonizes precision and recall, stands at 0.5222, reflecting a balanced trade-off between these metrics. These results highlight areas for improvement to enhance the MLP's ability to generalize and perform effectively.

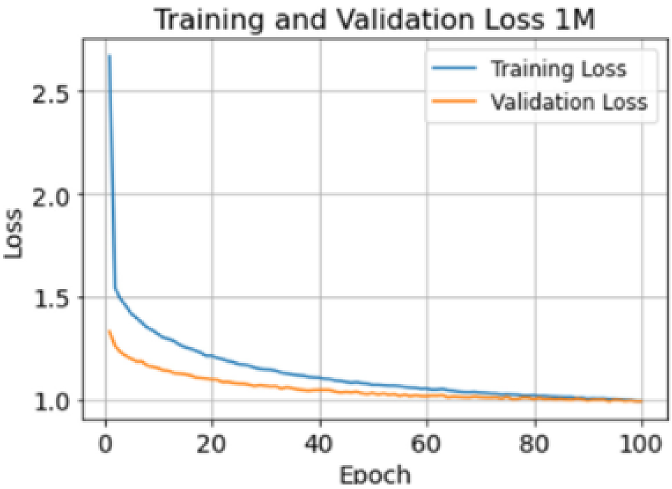

Fig. 7. Training and validation loss graph for MLP

The Figure 7 graph illustrates the training and validation loss over 100 epochs for the MLP model on the Movie Lens 1M dataset. The training loss (blue line) and validation loss (orange line) both demonstrate a downward trend, indicating consistent model optimization. Initially, the losses start at higher values but decline sharply within the first 20 epochs, suggesting rapid learning. Beyond 20 epochs, the rate of loss reduction slows, eventually stabilizing near 1.0 for both training and validation. The minimal gap between the two losses indicates that the model generalizes well without significant overfitting.

TABLE III. COMPARATIVE ANALYSIS FOR PERSONALIZED MOVIE RECOMMENDATIONS ON MOVIE LANCE 1M DATASET

Models	RMSE	MAE
DNN[16]	1.011	73.5
MLP	0.99	0.80

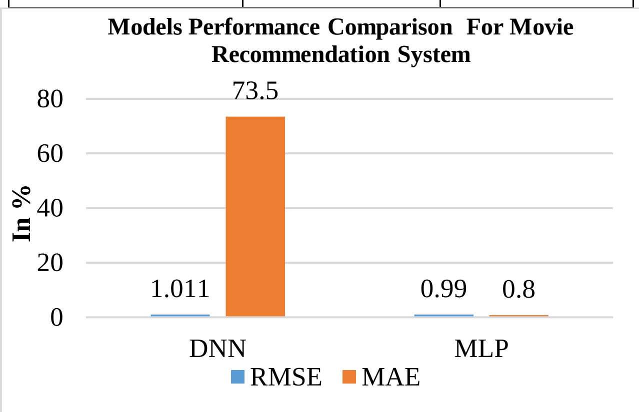

Fig. 8. Bar Graph for model performance

Figure 8 and Table III shows comparison of models reveals that the MLP outperforms the DNN in terms of both RMSE and MAE. The MLP achieves a lower RMSE of 0.99 compared to the DNN's 1.011, indicating better predictive accuracy. Furthermore, the MLP demonstrates superior error minimization with an MAE of 0.80, outperforming the DNN's MAE of 73.5, which suggests a significant disparity in performance. These results highlight the MLP's efficacy in reducing error metrics, making it a more robust choice for the analyzed application.

V. CONCLUSION AND FUTURE SCOPE

Recommendation solutions that operate based on user behavior can help users find websites for running along with applicable books, illustrations, TV shows, and music tracks. A movie recommender system provides movie suggestions to persons matching their interest patterns to reduce their time searching for movies across the vast options online. Results from the MovieLens 1M dataset study concluded that the personal recommendation system achieved better performance with Multilayer Perceptron (MLP) which showed RMSE of 0.99 and MAE of 0.80 while Deep Neural Network (DNN) achieved RMSE of 1.011 and MAE of 73.5. As promising performance shows limitations exist because the model struggles with complex user preferences and interactions along with the need for better handling of non-linear relationships and optimal feature selection. Further, including context information and collaboration filtering techniques can further enhance the recommendation system's ability by giving more personalized and accurate recommendations, thereby leading to development of stronger recommendation systems.

REFERENCES

1. Z. Cheng, Y. Ding, X. He, L. Zhu, X. Song, and M. Kankanhalli, "A3NCF: An adaptive aspect attention model for rating prediction," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2018-July, pp. 3748–3754, 2018, doi: 10.24963/ijcai.2018/521.
2. H. Chen, "Personalized learning resources recommendation model based on transfer learning," *Proc. - 2012 Int. Conf. Comput. Sci. Electron. Eng. ICCSEE 2012*, vol. 2, pp. 14–16, 2012, doi: 10.1109/ICCSEE.2012.501.
3. U. A. Khan, N. Ejaz, M. A. Martinez-Del-Amor, and H. Sparenberg, "Movies tags extraction using deep learning," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, AVSS 2017*, 2017. doi: 10.1109/AVSS.2017.8078459.
4. Y. Bao, H. Fang, and J. Zhang, "TopicMF: Simultaneously Exploiting Ratings and Reviews for Recommendation," *Proc. Natl. Conf. Artif. Intell.*, vol. 1, pp. 2–8, 2014, doi: 10.1609/aaai.v28i1.8715.
5. R. Xiong, J. Wang, N. Zhang, and Y. Ma, "Deep hybrid collaborative filtering for Web service recommendation," *Expert Syst. Appl.*, vol. 110, pp. 191–205, 2018, doi: https://doi.org/10.1016/j.eswa.2018.05.039.
6. R. Ahuja, A. Solanki, and A. Nayyar, "Movie recommender system using k-means clustering and k-nearest neighbor," in *Proceedings of the 9th International Conference On Cloud Computing, Data Science and Engineering, Confluence 2019*, 2019. doi: 10.1109/CONFLUENCE.2019.8776969.
7. E. Inan, F. Tekbacak, and C. Ozturk, "Moreopt: A goal programming based movie recommender system," *J. Comput. Sci.*, 2018, doi: 10.1016/j.jocs.2018.08.004.
8. N. Banik and M. Hasan Hafizur Rahman, "Evaluation of Naïve Bayes and Support Vector Machines on Bangla Textual Movie Reviews," in *2018 International Conference on Bangla Speech and Language Processing, ICBSLP 2018*, 2018. doi: 10.1109/ICBSLP.2018.8554497.
9. H. Wang and H. Zhang, "Movie genre preference prediction using machine learning for customer-based information," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference, CCWC 2018*, 2018. doi: 10.1109/CCWC.2018.8301647.
10. R. Rai and P. Mewada, "Sentiment Analysis of Movie Review using Machine Learning Approach," *IJOSTHE*, 2017, doi: 10.24113/ojssports.v5i1.83.
11. W. Serrano, "Neural Networks in Big Data and Web Search," *Data*, vol. 4, no. 1, 2019, doi: 10.3390/data4010007.
12. P. Li, S. Azman, M. Noah, and H. M. Sarim, "A Survey on Deep Neural Networks in Collaborative Filtering Recommendation Systems," 2018.
13. H. Yu, D. C. Samuels, Y. Zhao, and Y. Guo, "Architectures and accuracy of artificial neural network for disease classification from omics data," *BMC Genomics*, vol. 20, no. 1, p. 167, 2019, doi: 10.1186/s12864-019-5546-z.
14. T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)?– Arguments against avoiding RMSE in the literature," *Geosci. Model Dev.*, vol. 7, pp. 1247–1250, 2014, doi: 10.5194/gmd-7-1247-2014.
15. W. Wang and Y. Lu, "Analysis of the mean absolute error (MAE) and the root mean square error (RMSE) in assessing rounding model," in *IOP conference series: materials science and engineering*, 2018, p. 12049.
16. N. Liang, H. T. Zheng, J. Y. Chen, A. K. Sangaiah, and C. Z. Zhao, "TRSDL: Tag-aware recommender system based on deep learning-intelligent computing systems," *Appl. Sci.*, 2018, doi: 10.3390/app8050799..
17. Routhu, K., Bodepudi, V., Jha, K. M., & Chinta, P. C. R. (2020). A Deep Learning Architectures for Enhancing Cyber Security Protocols in Big Data Integrated ERP Systems. *Available at SSRN 5102662*.
18. Chinta, P. C. R., & Katnapally, N. (2021). Neural Network-Based Risk Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures. *Neural Network-Based Risk Assessment for Cybersecurity in Big Data-Oriented ERP Infrastructures*.
19. Katnapally, N., Chinta, P. C. R., Routhu, K. K., Velaga, V., Bodepudi, V., & Karaka, L. M. (2021). Leveraging Big Data Analytics and Machine Learning Techniques for Sentiment Analysis of Amazon Product Reviews in Business Insights. *American Journal of Computing and Engineering*, *4*(2), 35-51.
20. Karaka, L. M. (2021). Optimising Product Enhancements Strategic Approaches to Managing Complexity. *Available at SSRN 5147875*.
21. Chinta, P. C. R., & Karaka, L. M. AGENTIC AI AND REINFORCEMENT LEARNING: TOWARDS MORE AUTONOMOUS AND ADAPTIVE AI SYSTEMS.

22. Boppana, S. B., Moore, C. S., Bodepudi, V., Jha, K. M., Maka, S. R., & Sadaram, G. AI And ML Applications In Big Data Analytics: Transforming ERP Security Models For Modern Enterprises.
23. Chinta, P. C. R., Katnapally, N., Ja, K., Bodepudi, V., Babu, S., & Boppana, M. S. (2022). Exploring the role of neural networks in big data-driven ERP systems for proactive cybersecurity management. *Kurdish Studies*.
24. Chinta, P. C. R. (2022). Enhancing Supply Chain Efficiency and Performance Through ERP Optimisation Strategies. *Journal of Artificial Intelligence & Cloud Computing*, *1*(4), 10-47363.
25. Sadaram, G., Sakuru, M., Karaka, L. M., Reddy, M. S., Bodepudi, V., Boppana, S. B., & Maka, S. R. (2022). Internet of Things (IoT) Cybersecurity Enhancement through Artificial Intelligence: A Study on Intrusion Detection Systems. *Universal Library of Engineering Technology*, (2022).
26. Moore, C. (2023). AI-powered big data and ERP systems for autonomous detection of cybersecurity vulnerabilities. *Nanotechnology Perceptions*, *19*, 46-64.
27. Chinta, P. C. R. (2023). The Art of Business Analysis in Information Management Projects: Best Practices and Insights. *DOI*, *10*.
28. Chinta, P. C. R. (2023). Leveraging Machine Learning Techniques for Predictive Analysis in Merger and Acquisition (M&A). *Journal of Artificial Intelligence and Big Data*, *3*(1), 10-31586.
29. Krishna Madhav, J., Varun, B., Niharika, K., Srinivasa Rao, M., & Laxmana Murthy, K. (2023). Optimising Sales Forecasts in ERP Systems Using Machine Learning and Predictive Analytics. *J Contemp Edu Theo Artific Intel: JCETAI-104*.
30. Maka, S. R. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Available at SSRN 5116707*.
31. Routhu, KishanKumar & Katnapally, Niharika & Sakuru, Manikanth. (2023). Machine Learning for Cyber Defense: A Comparative Analysis of Supervised and Unsupervised Learning Approaches. Journal for ReAttach Therapy and Developmental Diversities. 6. 10.53555/jrtdd.v6i10s(2).3481.
32. Chinta, Purna Chandra Rao & Moore, Chethan Sriharsha. (2023). Cloud-Based AI and Big Data Analytics for Real-Time Business Decision-Making. 36. 96-123. 10.47363/JAICC/2023.
33. [33] Krishna Madhav, J., Varun, B., Niharika, K., Srinivasa Rao, M., & Laxmana Murthy, K. (2023). Optimising Sales Forecasts in ERP Systems Using Machine Learning and Predictive Analytics. *J Contemp Edu Theo Artific Intel: JCETAI-104*.
34. Bodepudi, V. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Journal of Artificial Intelligence and Big Data*, *3*(1), 10-31586.
35. Jha, K. M., Bodepudi, V., Boppana, S. B., Katnapally, N., Maka, S. R., & Sakuru, M. Deep Learning-Enabled Big Data Analytics for Cybersecurity Threat Detection in ERP Ecosystems.