LSTM-GNOG: A New Paradigm to Address Cold Start Movie Recommendation System using LSTM with Gaussian Nesterov's Optimal Gradient

Ravikumar R N[1], Sanjay Jain[2], Manash Sarkar[3]
Dept. of Computer Science and Engineering, Amity University, Rajasthan, Jaipur, India[1, 2]
Dept. of Computer Science and Engineering, Atria Institute of Technology, Bangalore, India[3]

Abstract—**In this modern streaming platform, the movie recommendation system is an important tool for enabling the users to find new content specialized to their interests. To address the cold start problem prevalent in movie recommendation systems, we introduce the Long Short-Term Memory-Gaussian Nesterov's Optimal Gradient (LSTM-GNOG) approach. This model utilizes both implicit and explicit feedback to effectively manage sparse rating data. By integrating Bayesian Personalized Ranking (BPR) and Probabilistic Matrix Factorization (PMF) algorithms with preprocessing via Singular Value Decomposition (SVD), our system enhances data robustness. Our empirical results on the MovieLens 100K, MovieLens 1M, FilmTrust, and Ciao datasets demonstrate significant improvements, with Mean Absolute Error (MAE) values of 0.4962, 0.5249, 0.4625, and 0.5341, respectively. Compared to traditional methods such as Unsupervised Boltzmann Machine-based Time-aware Recommendation (UBMTR) and Efficient Gowers-Jaccard-Sigmoid Measure (EGJSM), LSTM-GNOG shows better improvement in prediction accuracy. These results underscore the effectiveness of LSTM-GNOG in overcoming data sparsity issues in movie recommendations.**

***Keywords*—Cold start; Gaussian Nesterov's optimal gradient; long short-term memory; movie recommendation system; probabilistic matrix factorization**

I. Introduction

In the last few years, the growth of digital platforms like Netflix, Amazon Prime, and Hulu caused an explosion in the amount of available content [1]. Due to extensive varieties of movies and TV shows, the users find challenges frequently to determine new content that corresponds to the preferences [2]. Recommendation system is a data tool that helps users to discover what they want from an extensive range of accessible items [3]. Nowadays, movie recommender systems gain popularity among users which provide personalized recommendations based on their preferences, viewing history, and relevant information of likes and dislikes of the users [4], [5]. Nowadays, movie recommendation system is significant for allowing the users to find new content based on their interests [6]. The MRS is classified into three main types they are: Content-based (CB) filtering, Collaborative Filtering (CF) and Hybrid filtering. The CB filtering concentrate in the features of the contents itself, like director, actors, genre to generate recommendations [7], [8].

The CF is a type of recommender systems which against to CB recommendation methods and produce predictions according to previously estimated items through other users [9]. The CF exhibits the problems of user, and generates reliable ideas that learn from the user preferences. In addition, it displays the new item issues, which denote a new item that has been sufficiently reviewed by users [10], [11]. Moreover, total rating data are accessible to decide the achievement of a CF system. In movie recommendations, the user behaviors like preferences, interactions and view history are efficiently modelled through time which enables the model to understand user preferences [12], [13]. The cold start problem raised because of inadequate user-item data particularly for new user or item. The cold start problem has two different kinds such as item based and user-based problems [14]. This has been challenging for the system to recommend information to a new user in which the information is not saved in the system [15]. To overcome this issue, this paper proposed a LSTM-GNOG that is able to handle inadequate data by capturing long-term dependencies in user-item interactions and producing predictions and recommendations according to the patterns extracted from data.

The primary contribution is given as follows:

- The LSTM-GNOG is proposed in movie recommendation system which is able to handle inadequate data from preferences to address the cold start problem and producing predictions and recommendations according to the patterns.
- Implicit and explicit feedback data features are extracted by using BPR and PMF because the historical user ratings are scarce in cold start conditions. The rating data of explicit is inadequate, hence the SVD is used as pre-processing that improves the number of ratings.
- Finally, the extracted implicit and explicit features are fused and its user-item feature matrix are attained and unmarked items ranking scores are predicted by LSTM.

The rest of the article is structured as follows: Section II elaborates literature survey, Section III explains problem statement, Section IV shows proposed methodology process, Section V summarizes results and discussion and Section VI concludes the manuscript.

II. Related Works

In this section, the recent related works based on new user cold start problem in movie recommendation system are analyzed briefly. These techniques are automatically predicting the unknown ratings of user interest or items through analyzing known items or similar user preferences.

GM Harshvardhan et al. [16] implemented an UBMTR that detects movie-rating data in hidden features in connection with time. It considers time and ratings as two-input and output binary scores through contrastive divergence technique samples from Monte Carlo Markov Chain. It occurs a correlation among content intreated and temporal situations. It was seldom in a recommendation field through Boltzmann machines which are adoptable in pattern completion to overcome missing values and deals with imbalanced and unstructured data through encoding raw data into latent variables. However, model struggled to generate relevant movies because of the lack of historical data and preferences which outputs in poor user experience. Gourav Jain et al. [17] introduced a Cognitive Similarity-based Measure to improve the CF filtering performance in recommendation system. An EGJSM was developed which included nonlinear sigmoid function to punish bad ratings. The effectiveness of EGJSM based on similarity calculation in which cognitive or traditional approach are determine. In cognitive method, a cognitive similarity was developed, in that similarity was calculated through taking cognitive features with rating data. However, the recommendation system relies on explicit feedback data which unable to consider the implicit feedbacks about user performance which affected the recommendation performance.

Sophort Siet et al. [18] presented an enhancing sequential movie recommendation system through K-means and DL. Primarily, user behavior sequence was created which predict the potential target of movie users. Then, user data are integrated into movie sequence embeddings as input features for dimensionality reducing. The developed model incorporated transformer layer with positional encoding for user behavior sequences and multi-head self-attention to improve the prediction performance. Moreover, it applied into K-means clustering which was incorporated with cluster data and forecasted ratings for target users. It recommended movies based on user preferences which minimized decision exhaustion and improved the revenue generation. However, it unable to understand user preferences due to data sparsity which affected the performance. Junmei Feng et al. [19] implemented a combination of Probabilistic Matrix Factorization (PMF) and Bayesian Personalized Ranking (BPR) for cold start problem in recommendation system. It enabled use of implicit and explicit feedback information and overcome cold start problems to achieve precise data to items and users. The BPR was applied to extract user and item features from rating data. The rating data of user are preprocessed to confirm explicit feature extraction accuracy by normalizing the rating data. However, recommendation system relied on user-item interactions and unable to consider contextual data like location, time which impact the user preferences.

Yuyu Yuan et al. [20] suggested a multi-dimensional model named as UITrust depends on classification and entropy for recommendation system. It enhanced the recommendation quality through employing entropy information of item-user ratings. To minimize the prediction computational complexity and sparsity of weight matrix are compared with traditional techniques. The entropy was utilized to reflect the global behavior of item and user. It enhances the user experience by improving selection process and providing personalized recommendations. However, the number of movies and users enhanced, the personalized recommendations also enhanced which created scalability issues. G. Parthasarathy and S. Sathiya Devi et al. [21] introduced an Ensemble Learning based Collaborative Filtering with instance selection and improved clustering. The Classification and Regression Tree-Balanced Iterative Reducing and Clustering using Hierarchies (CART-BIRCH) for movie recommendation system. The hyper parameter tuning was included in BIRCH for improving cluster formation and it achieve movie recommendation to new users through Gradient Boost classification with coverage. However, the user has rated only available movies that lead sparse user-item matrices and difficult to find user or item recommendations. In Table I, from [22-28] shows the literature survey of the most recent existing work and its demerits.

TABLE I. LITERATURE SURVEY

Authors	Techniques	Merits	Demerits
Ravikumar et al.	K-Means, KNN, CF, CBF, TF-IDF, Cosine Similarity, Weighted Average, Min-Max Scaler	Personalized Recommendation, Improved quality recommendations and efficiency	Cold start, Scalability Concerns
Tain et al.	CBF, Feature extraction, Weighted rating	Enhanced recommendation, Comprehensive results	Dependent on browse history, Optimization techniques not applied.
Lee et al.	CF, Weighted rating	Personalized movie recommendation	Hybrid techniques can be applied, Optimization required.
Huang et al.	LLM-Interaction Simulator	Simulate vivid interactions for each cold item	Hyperparameter and optimization can be explored.
Ziaee et al.	MoRGH: Movie Recommender System using GNNs on Heterogeneous Graphs	GNN with hybrid CF and CBF based approach.	Huge dataset is not explored.
Hasan et al.	Alternating Least Squares (ALS) algorithm	Fusion of text to number and cosine similarity	Optimization techniques not applied.
Peng et al.	Deep RL with CF, Deep Deterministic Policy Gradient (DDPG) algorithm	Addressed sparsity and improved accuracy	Optimization algorithms not explored.

III. Problem Statement

Cold start problem is one of the most important concerns in the recommendation system. The domains such as E-commerce, Entertainment, social media, etc. will not progress if its wont address new user/item i.e. cold start problem [29]. From the above analysis, the existing recommendation systems relies on explicit feedback data which unable to consider the implicit feedbacks about user performance which affected the recommendation performance. Unable to understand user preferences due to data sparsity and the user has rated only available movies that lead sparse user-item matrices and difficult to find user or item recommendations [30]. Also, in terms of hybrid models more focus is not given to optimization techniques which is very useful in terms of reducing the computation time and increase the response time with minimal efforts [31]. So, to address all these issues we propose a LSTM-GNOG for prediction and recommendation which enables user feedback data of score and implicit data to attain precise data presenting user and items characteristics. Hence, the proposed LSTM-GNOG overcomes the cold start problem and enhances the prediction and recommendation.

IV. Proposed Methodology

The LSTM-GNOG movie recommendation system quickly solves the cold start problem and makes predictions and suggestions based on data patterns. The suggested movie recommendation model understands user preferences by effectively modelling user behaviours including preferences, interactions, and watch history across time. Insufficient user-item data, especially for new users or items, caused the cold start problem. However, LSTM-GNOG can manage limited preferences data to overcome the cold start problem and provide patterns-based predictions and recommendations.

- Initially, four datasets such as MovieLens 100K, MovieLens 1M, FilmTrust and Ciao are considered in this work for movie recommendation system.
- Then, the implicit and explicit feedback data are exploited because the historical user ratings are scarce in cold start conditions.
- To extract implicit features, the BPR is used and extract explicit features PMF is used. The rating information is inadequate for explicit data, therefore the pre-processing such as SVD is included before extracting features which enhances the number of ratings and predicts the unrated user items based on historical ratings.
- Lastly, extracted implicit and explicit features are fused and its user-item feature matrix are attained and unmarked items raking scores are predicted and recommended by LSTM-GNOG.

The process of proposed methodology is shown in the Fig. 1 with implicit and explicit feedback, rating data with SVD, BPR and PMF feature extraction, feature fusion and prediction and recommendation using the proposed LSTM-GNOG model.

A. Dataset

In this work, publicly accessible four datasets such as MovieLens 100K [32], MovieLens 1M [33], FilmTrust [34] and Ciao [35] are considered to establish performance of proposed model. The dataset is divided into 80% of training and 20% of testing.

1) MovieLens 100K: It includes 100,000 ratings for 1682 movies given by 943 users in which every user has rated minimum 20 movies. These scores are integer and ranges between 1-5 in that 1 denotes bad feedback and 5 denotes best feedback[1].

2) MovieLens 1M: It comprises 1,000,209 unidentified ratings for 3706 movies given by 6040 users, these scores are integer and ranges between 1-5.

3) FilmTrust: It includes 35,497 ratings for 2071 movies given by 1508 users and these scores are multiplied by 0.5 and ranges between 0.5-4.0.

4) Ciao: It includes 72,665 ratings for 12,121 items given by 17,615 users and the ratings are ranges between 1 and 5.

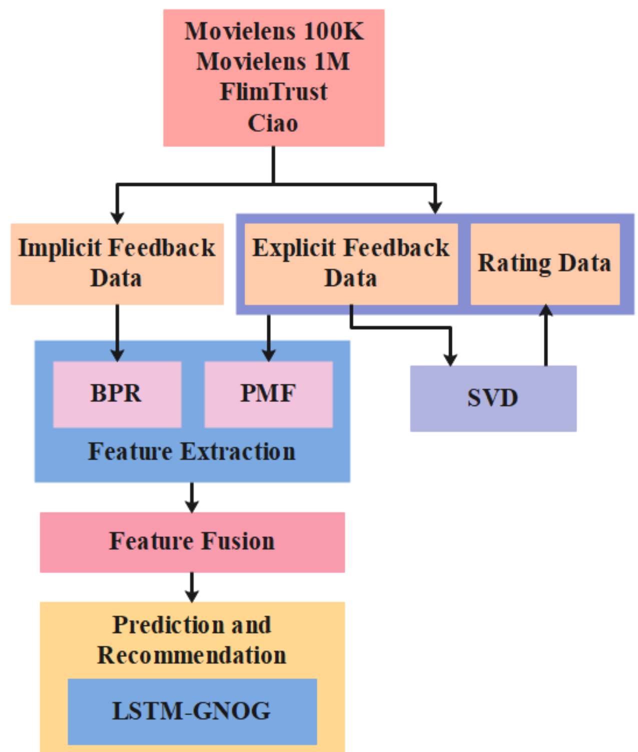

Fig. 1. Process of proposed methodology

B. Feature Extraction

The dataset is dividing into implicit and explicit feedback data which are exploited because the historical user ratings are scarce in cold start conditions. To extract implicit features, the BPR is used and extract explicit features PMF is used. The rating information is inadequate for explicit features, therefore the pre-processing such as SVD is included before extracting features which enhances the number of ratings and predicts the unrated user items based on historical ratings and extracted BPR and PMF features are fused which are clearly explained in the below section.

[1] https://www.kaggle.com/code/ravikumarrn/movie-recsys-datasets

1) Probabilistic matrix factorization (PMF): The PMF is employed to extract explicit features from data that enables personalized recommendations by learning latent features which captures individual user preferences. It able to mitigate cold-start problems and able to handle user-item sparsity in recommendation system. By leveraging latent features, it recommended to user and items with sparse data with existing user-items. It is a matrix decomposition method that used in CF recommendations which establishes the possibility ideas based on matrix decomposition [29]. The PMF uses dual low-rank matrices U and V for denoting user-item rating matrix R. Consider matrices R, U and I as following a Gaussian distribution, which are indicated in Eq. (1-3),

$$p(R|U,I,\sigma^2) = \prod_{u=1}^{m}\prod_{i=1}^{n}[N(r_{ui}|p_u^T q_i,\sigma^2)]^{I_{ui}} \tag{1}$$

$$p(U|\sigma_U^2) = \prod_{u=1}^{m} N(p_u|0,\sigma_U^2 I) \tag{2}$$

$$p(I|\sigma_I^2) = \prod_{i=1}^{n} N(q_i|0,\sigma_I^2 I) \tag{3}$$

Where, $p(R|U,I,\sigma^2)$ is a probability of rating matrix R for user U and item I, $N(\sigma^2)$ is a Gaussian distribution with a variance σ^2, I_{ui} is an indicator function. If user U rated item I, the I_{ui} is accurate else 0. The I is an identity matrix with dimension f, m and n are a total number of user and items, $p_u^T q_i$ is a predicted rating for user u and item I, σ_U^2 is a covariance matrix, p_u and q_i are latent feature vector for each user U and item I. The possibilities of U and I obtained by Bayesian formula which is indicated in Eq. (4),

$$E = \underset{u,i}{\arg min} \frac{1}{2}\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}\,(r_{ui}|p_u^T q_i)^2 + \frac{\lambda_U}{2}\|p_u\|^2 + \frac{\lambda_I}{2}\|q_i\|^2 \tag{4}$$

Where, r_{ui} is an observed rating, I_{ui} is an indicator function, $\lambda_U = \frac{\sigma^2}{\sigma_U^2}$ and $\lambda_I = \frac{\sigma^2}{\sigma_I^2}$ are regularization coefficients that is employed to diminishing the over-fitting issues.

2) Bayesian personalized ranking (BPR): The Non-negative Matrix Factorization (NMF) is a matrix factorization technique which suffered from scalability issues when dealing with huge-dimensional data so, in this research BPR is employed to extract implicit features from data which efficiently handles implicit feedback data in which user preferences are contingent from actions like views, clicks and purchases than explicit ratings. The BPR uses matrix factorization techniques for user-item interactions. By decaying user-item interactions into feature vector, it learns low-dimensional illustrations of users and items. It delivers a user by personalized ranking list according to the implicit feedback like transaction, click activity and view history [30]. It is formed based on assumption of user preferences on item i to j, if the user U selects a valued item i to un-valued item j, $\left(i \in N(u)\; and\; j \in \bar{N}(u)\right)$. Furthermore, consider relative order among items pair, rather than only user-item pairs. The BPR of common optimization criterion for personalized ranking is indicated in Eq. (5),

$$BPR - Opt = \sum_{(u,i,j)\in D_S} In(\sigma(\hat{r}_{uij})) - \lambda_\Theta\|\Theta\|^2 \tag{5}$$

Where, $D_S = \{(u,i,j)|i \in N(u)\; and\; j \in \bar{N}(u)\}$, and D_S is a dataset, $\hat{r}_{uij}$ captures special connection among two items and user that is demarcated as $\hat{r}_{uij} = \hat{r}_{ui} - \hat{r}_{uj}$. The predicted ranking score $\hat{r}_{ui} = q_i^T p_u + b_i.\sigma(x)$ utilizes logistic sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$. The constant λ_Θ manages model regularization, σ is a sigmoid function. The model's parameter vector is depicted by $\Theta = \{b_u, b_j, p_u, q_i, q_j\}$ and it is learned by LSTM-GNOG.

3) Integrating PMF-BPR: The integration of PMF and BPR helps to address the sparsity problem in recommendation system. The PMF captures explicit preferences integrated by users through movie feedback or ratings whereas BPR deliberates implicit feedback data to capture preferences that user unable to express explicitly. The explicit feedback provides user preferences directly like how much users dislike and like a particular item but, the implicit feedback provides indirect user preferences and behaviours. By integrating both feedbacks, the recommendation system produces accurate prediction and recommendation which leads to enhance the recommendation quality. The BPR and PMF are linear technique since amplitude score of forecasted ranking value in Eq. (5) is unidentified and predicted score in Eq. (4) required to inadequate to user score range. Then, forecasted scores are measured as $\hat{r}_{ui} = q_i^T p_u$ and sigmoid function is used to process $\hat{r}_{ui}$ and normalize it into (0, 1). After restricting the amplitude of PMF is indicated in Eq. (6),

$$E = \underset{u,i}{\arg min} \frac{1}{2}\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}\left(r_{ui} - \omega\sigma(\hat{r}_{ui})\right)^2 + \frac{\lambda_U}{2}\|p_u\|^2 + \frac{\lambda_I}{2}\|q_i\|^2 \tag{6}$$

Where, $\sigma(x)$ is a sigmoid function and ω is an amplitude parameter which is used to change predicted value into users rating range. Hence, $\omega = r_{max}$ in which r_{max} is a highest user rating on dataset. The Eq. (6) is modified and indicated in Eq. (7),

$$E = \underset{u,i}{\arg min} \frac{1}{2}\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}\left(r_{ui} - r_{max}\sigma(\hat{r}_{ui})\right)^2 + \frac{\lambda_U}{2}\|p_u\|^2 + \frac{\lambda_I}{2}\|q_i\|^2 \tag{7}$$

The number of user rating is minimum in cold start, PMF unable to extract the explicit features according to historical ratings. Hence, before feature extraction, the SVD is used as pre-processing to predict the unrated user items according to historical ratings that is efficiently reduce sparsity issues. Then, predicted and historical scores are applied as PMF input data. Therefore, after pre-processing the PMF loss function is indicated in Eq. (8).

$$E = \frac{1}{2}\left(\sum_{(u,i)\in T}\left(r_{ui} - r_{max}\sigma(\hat{r}_{ui})\right)^2 + \sum_{(u,i)\in T'}\left(\tilde{r}_{ui} - r_{max}\sigma(\hat{r}_{ui})\right)^2\right) + \frac{\lambda_U}{2}\|p_u\|^2 + \frac{\lambda_I}{2}\|q_i\|^2 \tag{8}$$

Where, T is a historical rating set, T' is a set of predicted rating attained by LSTM-GNOG model, $\tilde{r}_{ui}$ is a predicted score of users U on unrated item i according to LSTM-GNOG model.

The extracted implicit and explicit features are fused and given to prediction and recommendation.

C. Prediction and Recommendation

The fused features are predicted and recommended by using LSTM-GNOG which is helpful for handling historical data from user preferences and producing prediction based on extracted data patterns. In movie recommendations, the user behaviours like preferences, interactions and view history are efficiently modelled through time which enables the model to understand user preferences. In movie recommendation system, the movie features and user behaviours are denoted as sequential information. The LSTM gates such as input, output and forgot gates are used to manage the memory retention and flow of data which adopt to learn long-term dependences and user preferences. The cold start problem raised because of inadequate user-item data particularly for new user or item. However, LSTM-GNOG is able to handle inadequate data from preferences to address the cold start problem and producing predictions and recommendations according to the patterns. The LSTM gates and GNOG facilitates the model to retaining significant data through long sequences which mitigates vanishing gradient issues by adapting learning rate. The LSTM-GNOG enables the model for making prediction according to user interaction with movies. It is helpful for mitigating cold start problems where there is no historical data for new items or users. Each input that is user U and corresponding items I are attained from the recommendations which is indicated in Eq. (9),

$$X_t \rightarrow \sum_{i,j=1}^{m,n} U_i I_j \tag{9}$$

In Eq. (9), X_t is an input attained from dataset and saved in an input layer at time t, m is a number of users U, n is a number of items I, i is a valued item, j is an un-valued item then, user-item matrix is indicated in Eq. (10),

$$X_t = \begin{bmatrix} u_1 i_1 & u_1 i_2 & u_1 i_3 & \cdots & u_1 i_n \\ u_2 i_1 & u_2 i_2 & u_2 i_3 & \cdots & u_2 i_4 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ u_m i_1 & u_m i_2 & u_m i_3 & \cdots & u_m i_n \end{bmatrix} \tag{10}$$

The user-item matrix attained at various time and the input is transferred to next layer. In initial hidden layer, the optimizes classification of recommendations are created by GNOG model. Consider that, $U = u_1, u_2, u_3, \ldots, u_n$ according to user and $I = i_1, i_2, i_3, \ldots, i_n$ are items created by corresponding user. The ranks assigned through user on respective items are presented as user-item ranking matrix $R \in R^{u \times i}$ where $R^{u \times i}$ is a ranking of item allocated by user. In this work, ranking is presented by 1-5. In a ranking social network of user and items are created through every user and set a proximate P_i and $C_{u,i}$ is a confidence score. If an output range is zero it denotes confidence exist between user i and j. In the proposed model, contingent collinear with Gaussian function is used which is indicated in Eq. (11),

$$Res = Prob(R|U, I, \sigma^2) = \prod_{u=1}^{m} \prod_{i=1}^{n} \left[N(r_{ui} | p_u^T q_i, \sigma^2) C_{u,i} \right] \tag{11}$$

$N(\sigma^2)$ is a Gaussian distribution with a variance σ^2, I_{ui} is an indicator function. User U valued the item i, the I_{ui} is true else 0. The I is an identity matrix with dimension f, the possibilities of U and V obtained by Bayesian formula. Then, Nesterov accelerated gradient in U and I are used to reduce objective function presented in Eq. (9). This process is employed to train neural network for optimizing contingent function. The objective remains two various parameters such as ϑ and l and its distance according to learning rate $\varepsilon > 0$ with coefficient momentum of $\mu \in [0,1]$. Then, updated formula for two various users U and I are indicated in Eq. (12), (13), and its momentum of every user U and I is indicated in Eq. (14), (15):

$$U[\theta^l] = U_i^{(l+1)}[Res^l] - \varepsilon^l \nabla f(Res^l) \tag{12}$$

$$I[\theta^l] = I_i^{(l+1)}[Res^l] - \varepsilon^l \nabla f(Res^l) \tag{13}$$

$$Res^{l+1} = U[\theta^l] + \mu^l (U[\theta^l] - U[\theta^{l-1}]) \tag{14}$$

$$Res^{l+1} = I[\theta^l] + \mu^l (I[\theta^l] - I[\theta^{l-1}]) \tag{15}$$

In input layer for every user by items are attained as input and user-item matrix is attained based on confidence score. Then, input is passed to first hidden layer here, possibility collinear through Gaussian function and gradient descent for dual users are calculated. Then, two various users' momentum is estimated and to address missing values. The LSTM is used to efficiently model the sequential patterns of movie features and user behaviour. The LSTM holds past likes and dislikes of users in memory cell state. It recollects likes and dislikes over arbitrary time and its three gates like input, output and forget gate tunes item flow into out of cell state which contributes high recall rate. In LSTM, cell state is developed to run various types of sentiment chains which includes both positive and negative reviews with various interactions. Every memory cell generates a positive and negative feedbacks of movies to cell state. The gates include pointwise multiplication and sigmoid operations. The forget gate contains forgetting coefficient attained by input layer X_t and past hidden layer H_{t-1} for respective cell state C_{t-1}. The forget gate F_t helps the cell to smooth the items of internal states which is indicated in Eq. (16),

$$F_t = \sigma(W_F \times [H_{t-1}, X_t] + B_F) \tag{16}$$

Here, followed by activation from input X_t is attained and past hidden layer H_{t-1} using $tanh$ function into the aggregated weight input model which is formulated in Eq. (17),

$$G_t = tanh(W_G \times [H_t, X_t] + B_G) \tag{17}$$

The input gate defines the items to updated in respective cell state C_{t-1} and its output accumulates the resultant input node score to produce new cell state into respective cell state C_t. The input gate I_t is used to manage how much items are entered into the cell which is indicated in Eq. (18),

$$I_t = \sigma(W_I \times [H_{t-1}, X_t] + B_I) \tag{18}$$

Where, σ is a sigmoid function, W_I is a weight matrix, X_t is a present cell input, H_{t-1} is a hidden state, B_I is a bias factor of input gate. Then, the internal or memory cell state C_t is indicated in Eq. (19),

$$C_t = F_t \times C_{t-1} + I_t \times G_t \tag{19}$$

Where, F_t is a forget gate, I_t is an input gate, G_t is an input node. Lastly, the output cell state O_t creates a block of predictive recommendations which is indicated in Eq. (20), and (21),

$$O_t = \sigma(W_O \times [H_{t-1}, X_t] + B_O) \quad (20)$$

$$H_t(RT) = O_t \times \tanh(C_t) \quad (21)$$

From Eq. (20) and (21), σ is a sigmoid function, o is a weight matrix, X_t is a present cell input, B_o is a bias factor of output gate, $tanh$ is an activation function, the output of hidden layer H_t are attained through internal state C_t and output items from output gate O_t accordingly. With the past hidden layer result attained from Nesterov accelerated gradient optimized classification, the vanishing gradient and cold start problems are addressed and new layer is produced for every time of input processed through the network. The proposed LSTM-GNOG pseudocode is given below which enables user feedback data of score and implicit data to attain precise data presenting user and items characteristics.

Algorithm: Proposed Pseudocode for LSTM-GNOG

```
1. Preprocess and extract features
2. Initialize LSTM model and GNOG optimizer
3. Define loss function for model training
4. Training Process
   | TrainModel Function
   | Train for specified epochs
   |            | Batch training
   |            | Forward pass
   |            | Backward pass and optimize
   |            |   | Evaluate on validation data
   | ReturnModel |   | PrintLoss
5. LSTM-GNOG model training
6. MakePrediction() function to perform prediction
7. GenerateRecommendation() function to generate
   recommendation
8. Usage of trained model to make predictions and
   recommendations
9. End
```

V. Experimental Result

The result achievements of proposed LSTM-GNOG are estimated against the known measure for recommendation and prediction. The performance measures of MAE and RMSE are used for prediction, the precision, recall and f1score are used for recommendation. The mathematical validation of these metrics is indicated in Eq. (22) to (26),

$$MAE = \frac{1}{N}\sum_{u,i}|p_{u,i} - r_{u,i}| \quad (22)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{u,i}|p_{u,i} - r_{u,i}|^2} \quad (23)$$

$$Precision = \frac{No.of\ correct\ recomm.relevant\ to\ total\ query}{No.of\ recommendations} \quad (24)$$

$$Recall = \frac{No.of\ correct\ recommendation}{Total\ no.of\ relevant\ recommendations} \quad (25)$$

$$F1score = \frac{2}{\frac{1}{Precision}+\frac{1}{Recall}} \quad (26)$$

Where, N is a total number of user-item pairs, $p_{u,i}$ and $r_{u,i}$ are the predicted and actual rating of user.

A. Quantitative and Qualitative Analysis

The quantitative result achievement of proposed LSTM-GNOG is examined on four various datasets. In movie recommendations, the user behaviours like preferences, interactions and view history are efficiently modelled through time which enables the model to understand user preferences. However, the LSTM-GNOG is used to manage insufficient data from user preferences and making predictions and recommendations based on patterns. The GNOG learning rate mechanism enables the model efficiently employs the available data and enables learning rate based on data sparsity which allows to learn efficiently sparse or incomplete user-item interactions. The result achievements are compared with other prediction and recommendation techniques such as LSTM with Batch Gradient Descent (BGD), Conjugate Gradient Descent (CGD), Stochastic Gradient Descent (SGD) and Nesterov Accelerated Gradient (NAG). By examining Tables III to VI, the LSTM-GNOG reached better result for all four datasets. The Table II shows the performance of PMF-BPR.

TABLE II. Result Achievement of PMF-BPR on All Four Datasets

Method	MAE	RMSE	Precision	Recall	F1score
NMF	0.6738	0.5316	0.6429	0.6142	0.6728
PMF	0.6582	0.5035	0.6714	0.6483	0.7051
BPR	0.5937	0.4728	0.7037	0.7951	0.7534
PMF-BPR	0.5261	0.4537	0.7369	0.8124	0.7958

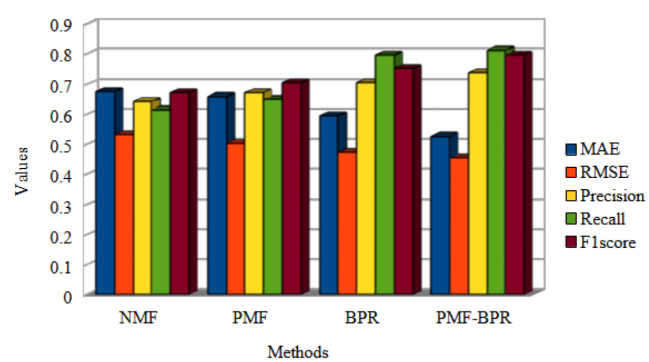

Fig. 2. Result achievement of PMF-BPR on all four datasets

In Table II and Fig. 2, the PMF-BPR result achievement with MAE, RMSE, precision, recall and f1score for all four datasets is presented. By integrating both feedbacks, the recommendation system produces accurate prediction and recommendation which leads to enhance the recommendation quality. By identifying relevant items, gathering user preferences and creating personalized recommendations, the recommendation system mitigates the cold start problem in implicit and explicit feedback data. The NMF, PMF and BPR result achievements are compared with PMF-BPR. The PMF-BPR achieves better result with MAE of 0.5261, RMSE of 0.4537, precision of 0.7369, recall of 0.8124 and f1score of 0.7958 which is higher when compared other prediction and recommendation techniques.

TABLE III. RESULT ACHIEVEMENT OF LSTM-GNOG ON MOVIELENS 100K DATASET

Method	MAE	RMSE	Precision	Recall	F1score
LSTM-BGD	0.6617	0.6135	0.7526	0.7363	0.7249
LSTM-CGD	0.6354	0.5546	0.7854	0.7986	0.7735
LSTM-SGD	0.5761	0.4963	0.8137	0.8341	0.8258
LSTM-NAG	0.5276	0.4581	0.8465	0.8532	0.8671
LSTM-GNOG	**0.4962**	**0.4157**	**0.8731**	**0.8965**	**0.8984**

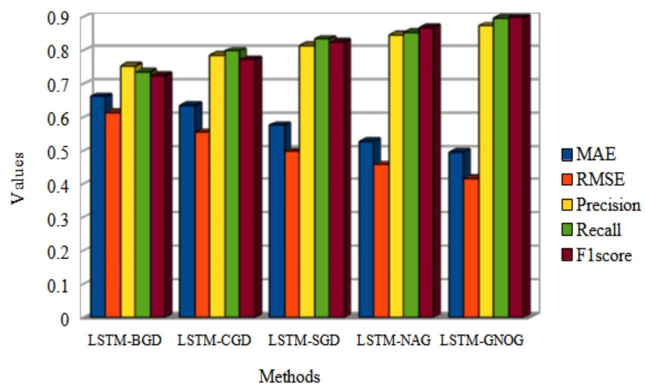

Fig. 3. Result achievement of LSTM-GNOG on MovieLens 100K dataset

In Table III and Fig. 3, the LSTM-GNOG result achievement with MAE, RMSE, precision, recall and f1score for MovieLens 100K dataset is presented. The LSTM with BGD, CGD, SGD and NAG result achievements are compared with LSTM-GNOG. The LSTM-GNOG achieves better result with MAE of 0.4962, RMSE of 0.4157, precision of 0.8731, recall of 0.8965 and f1score of 0.8984 which is higher when compared other prediction and recommendation techniques.

TABLE IV. RESULT ACHIEVEMENT OF LSTM-GNOG ON MOVIELENS 1M DATASET

Method	MAE	RMSE	Precision	Recall	F1score
LSTM-BGD	0.6724	0.5941	0.7168	0.7638	0.6919
LSTM-CGD	0.6255	0.5563	0.7517	0.7945	0.6643
LSTM-SGD	0.5936	0.5046	0.7865	0.8269	0.7081
LSTM-NAG	0.5671	0.4752	0.8054	0.8571	0.7365
LSTM-GNOG	**0.5249**	**0.4328**	**0.8336**	**0.8754**	**0.7841**

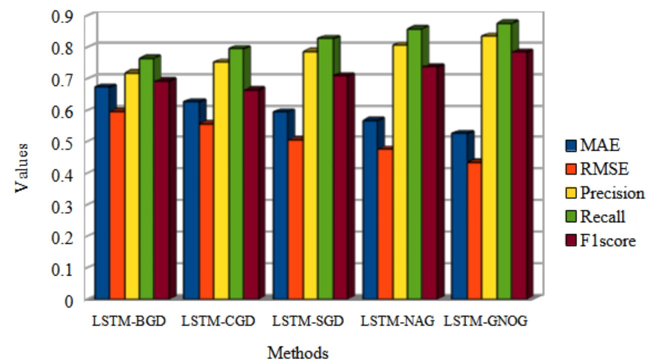

Fig. 4. Result achievement of LSTM-GNOG on MovieLens 1M dataset

In Table IV and Fig. 4, the LSTM-GNOG result achievement with MAE, RMSE, precision, recall and f1score for MovieLens 1M dataset is presented. The LSTM with BGD, CGD, SGD and NAG result achievements are compared with LSTM-GNOG. The LSTM-GNOG achieves better result with MAE of 0.5249, RMSE of 0.4328, precision of 0.8336, recall of 0.8754 and f1score of 0.7841 which is higher when compared other prediction and recommendation techniques.

TABLE V. RESULT ACHIEVEMENT OF LSTM-GNOG ON FILMTRUST DATASET

Method	MAE	RMSE	Precision	Recall	F1score
LSTM-BGD	0.6831	0.6744	0.3592	0.4793	0.3413
LSTM-CGD	0.6649	0.6221	0.3857	0.5171	0.3864
LSTM-SGD	0.6234	0.5869	0.4594	0.5857	0.4356
LSTM-NAG	0.5118	0.5153	0.4945	0.6116	0.4948
LSTM-GNOG	**0.4625**	**0.4461**	**0.5166**	**0.6749**	**0.5652**

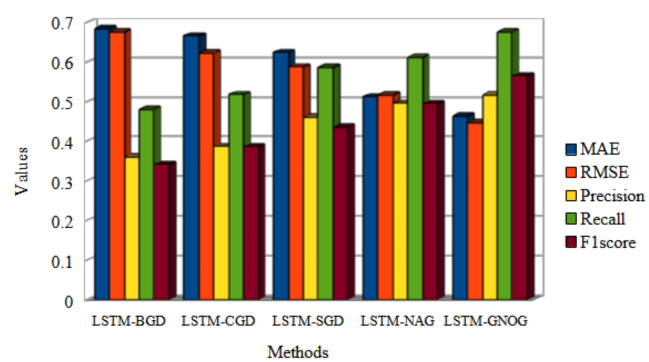

Fig. 5. Result achievement of LSTM-GNOG on FilmTrust dataset

In Table V and Fig. 5, the LSTM-GNOG result achievement with MAE, RMSE, precision, recall and f1score for FilmTrust dataset is presented. The LSTM with BGD, CGD, SGD and NAG result achievements are compared with LSTM-GNOG. The LSTM-GNOG achieves better result with MAE of 0.4625, RMSE of 0.4461, precision of 0.5166, recall of 0.6749 and f1score of 0.5652 which is higher when compared other prediction and recommendation techniques.

TABLE VI. RESULT ACHIEVEMENT OF LSTM-GNOG ON CIAO DATASET

Method	MAE	RMSE	Precision	Recall	F1score
LSTM-BGD	0.6738	0.5931	0.3136	0.3847	0.3565
LSTM-CGD	0.6352	0.5617	0.4195	0.4316	0.4671
LSTM-SGD	0.5927	0.5359	0.4257	0.5179	0.5916
LSTM-NAG	0.5765	0.4961	0.5845	0.6385	0.6348
LSTM-GNOG	**0.5341**	**0.4583**	**0.6341**	**0.7276**	**0.7152**

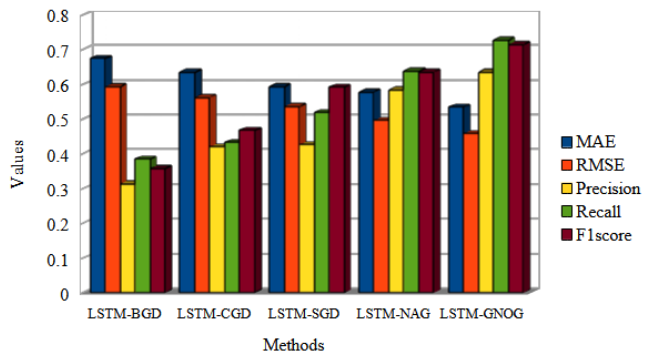

Fig. 6. Result achievement of LSTM-GNOG on Ciao dataset

In Table VI and Fig. 6, the LSTM-GNOG result achievement with MAE, RMSE, precision, recall and f1score for Ciao dataset is presented. The LSTM with BGD, CGD, SGD and NAG result achievements are compared with LSTM-GNOG. The LSTM-GNOG achieves better result with MAE of 0.5341, RMSE of 0.4583, precision of 0.6341, recall of 0.7276 and f1score of 0.7152 which is higher when compared other prediction and recommendation techniques.

B. Comparative Analysis

The comparative result achievement of proposed LSTM-GNOG is examined on four various datasets. In Table VII, the result achievements are compared with other prediction and recommendation techniques such as UBMTR [16], EGJSM [17], K-means [18], RBPR [19], UITrust_C [20] and CART-BIRCH [21]. To overcome the cold start issue, the LSTM-GNOG can accept minimal preference data and provide patterns-based predictions and suggestions. The proposed model uses score and implicit data to accurately represent user and item attributes.

C. Discussion

The existing prediction and recommendation techniques like UBMTR [16] attained MAE of 0.76 for MovieLens100K dataset because it struggled to generate relevant movies because of the lack of historical data and preferences which outputs in poor user experience. EGJSM [17] attained MAE of 0.7278 for MovieLens100K because it relies on explicit feedback data which unable to consider the implicit feedbacks about user performance which affected the recommendation performance. The K-means [18] attained MAE of 0.8741 for MovieLens100K unable to understand user preferences due to data sparsity which affected the performance. The RBPR [19] attained precision of 0.0575 for MovieLens100K dataset because it relied on user-item interactions and unable to consider contextual data like location, time which impact the user preferences. The UITrust_C [20] attained MAE of 0.938 because the number of movies and users enhanced, the personalized recommendations also enhanced which created scalability issues. The CART-BIRCH [21] attained MAE of 0.520 for MovieLens100K dataset because user has rated only available movies that lead sparse user-item matrices and difficult to find user or item recommendations. However, to overcome these issues the LSTM-GNOG is proposed in this research which achieves MAE of 0.4962 for MovieLens100K dataset because it addresses cold start problem by handling scarce data and producing prediction and recommendation based on extracted data.

In future, the model may be compared to better Deep Learning methods. For more personalised recommendations, use real-time machine learning. Federated Learning and Gossip Learning can be researched since recommendation systems must protect privacy. Also, a hybrid optimisation method can be developed which may increase recommender system efficiency.

TABLE VII. RESULT ACHIEVEMENT OF LSTM-GNOG AND EXISTING TECHNIQUES COMPARISON

Dataset	Method	MAE	RMSE	Precision	Recall	F1 score
Movie Lens 100K	UBMTR [16]	0.76	0.88	N/A	N/A	N/A
	EGJSM [17]	0.727	1.019	N/A	N/A	N/A
	K-means [18]	0.874	1.075	0.551	0.326	0.409
	RBPR [19]	N/A	N/A	0.057	0.077	N/A
	UITrust_C [20]	0.938	0.942	N/A	N/A	N/A
	CART-BIRCH [21]	N/A	0.439	0.835	0.864	0.867
	LSTM-GNOG	**0.496**	**0.415**	**0.873**	**0.896**	**0.898**
Movie Lens 1M	EGJSM [17]	0.732	1.012	N/A	N/A	N/A
	K-means [18]	0.800	0.992	0.583	0.472	0.522
	RBPR [19]	N/A	N/A	0.107	0.141	N/A
	UITrust_C [20]	0.91	0.914	N/A	N/A	N/A
	CART-BIRCH [21]	0.571	0.450	0.795	0.857	0.666
	LSTM-GNOG	**0.524**	**0.432**	**0.833**	**0.875**	**0.784**
Film Trust	EGJSM [17]	0.583	0.872	N/A	N/A	N/A
	RBPR [19]	N/A	N/A	0.1517	0.3057	N/A
	LSTM-GNOG	**0.462**	**0.446**	**0.516**	**0.674**	**0.565**
Ciao	RBPR [19]	N/A	N/A	0.0113	0.035	N/A
	LSTM-GNOG	**0.534**	**0.458**	**0.634**	**0.727**	**0.715**

VI. Conclusion

The LSTM-GNOG is proposed in this article, which is helpful for capturing sequences and temporal dependences. It is able to handle inadequate data from preferences to address the cold start problem and producing predictions and recommendations according to the patterns. The implicit and explicit features are extracted by using BPR and PMF. The integration of PMF and BPR helps to address the sparsity problem in recommendation system. The PMF captures explicit preferences integrated by users through movie feedback or ratings whereas BPR deliberates implicit feedback data to capture preferences that user unable to express explicitly. By incorporating both feedbacks, the recommendation system produces accurate prediction and recommendation which leads to enhance the recommendation quality. The LSTM-GNOG enables user feedback data of score and implicit data to obtain precise data presenting user and items characteristics. The result achievement shows that LSTM-GNOG achieves MAE of 0.4962, 0.5249, 0.4625 and 0.5341 for MovieLens 100K, MovieLens 1M, FilmTrust and Ciao datasets.

References

[1] S. Sridhar, D. Dhanasekaran, and G. C. P. Latha, "Content-Based Movie Recommendation System Using MBO with DBN," Intell. Autom. Soft Comput., vol. 35, no. 3, pp. 3241–3257, 2023, doi: 10.32604/iasc.2023.030361.

[2] S. Liu, Y. Liu, X. Zhang, C. Xu, J. He, and Y. Qi, "Improving the Performance of Cold-Start Recommendation by Fusion of Attention Network and Meta-Learning," Electron., vol. 12, no. 2, pp. 1–14, 2023, doi: 10.3390/electronics12020376.

[3] A. Y. Mir, M. Zaman, S. M. K. Quadri, and S. A. Fayaz, "An Adaptive Classification Framework for Handling the Cold Start Problem in Case of News Items," Rev. d'Intelligence Artif., vol. 36, no. 6, pp. 889–896, 2022, doi: 10.18280/ria.360609.

[4] C. Y. Chen and J. J. Huang, "Temporal-Guided Knowledge Graph-Enhanced Graph Convolutional Network for Personalized Movie Recommendation Systems," Futur. Internet, vol. 15, no. 10, 2023, doi: 10.3390/fi15100323.

[5] M. Rahman, I. A. Shama, S. Rahman, and R. Nabil, "Hybrid Recommendation System To Solve Cold Start Problem," J. Theor. Appl. Inf. Technol., vol. 100, no. 11, pp. 3562–3580, 2022.

[6] Y. Ali et al., "A hybrid group-based movie recommendation framework with overlapping memberships," PLoS One, vol. 17, no. 3 March, pp. 1–28, 2022, doi: 10.1371/journal.pone.0266103.

[7] S. Airen and J. Agrawal, "Movie Recommender System Using K-Nearest Neighbors Variants," Natl. Acad. Sci. Lett., vol. 45, no. 1, pp. 75–82, 2022, doi: 10.1007/s40009-021-01051-0.

[8] E. Kannout, M. Grzegorowski, M. Grodzki, and H. S. Nguyen, "Clustering-Based Frequent Pattern Mining Framework for Solving Cold-Start Problem in Recommender Systems," IEEE Access, vol. 12, no. January, pp. 13678–13698, 2024, doi: 10.1109/ACCESS.2024.3355057.

[9] A. Almu, A. Ahmad, A. Roko, and M. Aliyu, "Incorporating Preference Changes through Users' Input in Collaborative Filtering Movie Recommender System," Int. J. Inf. Technol. Comput. Sci., vol. 14, no. 4, pp. 48–56, 2022, doi: 10.5815/ijitcs.2022.04.05.

[10] L. Li, H. Huang, Q. Li, and J. Man, "Personalized movie recommendations based on deep representation learning," PeerJ Comput. Sci., vol. 9, pp. 1–25, 2023, doi: 10.7717/peerj-cs.1448.

[11] Ayesha Siddique, M Kamran Abid, Muhammad Fuzail, and Naeem Aslam, "Movies Rating Prediction using Supervised Machine Learning Techniques," Int. J. Inf. Syst. Comput. Technol., vol. 3, no. 1, pp. 40–56, 2024, doi: 10.58325/ijisct.003.01.0062.

[12] A. Noorian, A. Harounabadi, and M. Hazratifard, "A sequential neural recommendation system exploiting BERT and LSTM on social media posts," Complex Intell. Syst., vol. 10, no. 1, pp. 721–744, 2024, doi: 10.1007/s40747-023-01191-4.

[13] Y. Yuan, L. Chen, and J. Yang, "A Multidimensional Model for Recommendation Systems Based on Classification and Entropy," Electronics, vol. 12, no. 2, p. 402, 2023, doi: 10.3390/electronics12020402.

[14] J. Herce-Zelaya, C. Porcel, Á. Tejeda-Lorente, J. Bernabé-Moreno, and E. Herrera-Viedma, "Introducing CSP Dataset: A Dataset Optimized for the Study of the Cold Start Problem in Recommender Systems," Inf., vol. 14, no. 1, pp. 1–17, 2023, doi: 10.3390/info14010019.

[15] A. Gonzalez, F. Ortega, D. Perez-Lopez, and S. Alonso, "Bias and Unfairness of Collaborative Filtering Based Recommender Systems in MovieLens Dataset," IEEE Access, vol. 10, no. July, pp. 68429–68439, 2022, doi: 10.1109/ACCESS.2022.3186719.

[16] G. M. Harshvardhan, M. K. Gourisaria, S. S. Rautaray, and M. Pandey, "UBMTR: Unsupervised Boltzmann machine-based time-aware recommendation system," J. King Saud Univ. - Comput. Inf. Sci., vol. 34, no. 8, pp. 6400–6413, 2022, doi: 10.1016/j.jksuci.2021.01.017.

[17] G. Jain, T. Mahara, S. C. Sharma, and A. K. Sangaiah, "A Cognitive Similarity-Based Measure to Enhance the Performance of Collaborative Filtering-Based Recommendation System," IEEE Trans. Comput. Soc. Syst., vol. 9, no. 6, pp. 1785–1793, 2022, doi: 10.1109/TCSS.2022.3187430.

[18] S. Siet, S. Peng, S. Ilkhomjon, M. Kang, and D. Park, "Enhancing Sequence Movie Recommendation System Using Deep Learning and KMeans," 2024.

[19] J. Feng, Z. Xia, X. Feng, and J. Peng, "RBPR: A hybrid model for the new user cold start problem in recommender systems," Knowledge-Based Syst., vol. 214, p. 106732, 2021, doi: 10.1016/j.knosys.2020.106732.

[20] Y. Mu and Y. Wu, "Multimodal Movie Recommendation System Using Deep Learning," Mathematics, vol. 11, no. 4, pp. 1–12, 2023, doi: 10.3390/math11040895.

[21] G. Parthasarathy and S. S. Devi, "Ensemble learning based collaborative filtering with instance selection and enhanced clustering," Comput. Mater. Contin., vol. 71, no. 2, pp. 2419–2434, 2022, doi: 10.32604/cmc.2022.019805.

[22] R. N. Ravikumar, S. Jain, and M. Sarkar, "Efficient Hybrid Movie Recommendation System Framework Based on A Sequential Model," Int. J. Intell. Syst. Appl. Eng., vol. 11, no. 9, pp. 145–155, 2023.

[23] X. Tian, "Content-based Filtering for Improving Movie Recommender System," no. Dai 2023, pp. 598–609, 2024, doi: 10.2991/978-94-6463-370-2_61.

[24] H.-C. Lee, Y.-S. Kim, and S.-W. Kim, "Real-Time Movie Recommendation: Integrating Persona-Based User Modeling with NMF and Deep Neural Networks," Appl. Sci., vol. 14, no. 3, p. 1014, 2024, doi: 10.3390/app14031014.

[25] F. Huang, Z. Yang, J. Jiang, Y. Bei, Y. Zhang, and H. Chen, "Large Language Model Interaction Simulator for Cold-Start Item Recommendation," 2024, [Online]. Available: http://arxiv.org/abs/2402.09176

[26] S. S. Ziaee, "MoRGH : Movie Recommender System using GNNs on Heterogeneous Graphs MoRGH : Movie Recommender System using GNNs on Heterogeneous Graphs," pp. 0–17, 2024.

[27] R. Hasan MBA and J. Ferdous, "Dominance of AI and Machine Learning Techniques in Hybrid Movie Recommendation System Applying Text-to-number Conversion and Cosine Similarity Approaches," J. Comput. Sci. Technol. Stud., pp. 94–102, 2024, doi: 10.32996/jcsts.

[28] S. Peng, S. Siet, S. Ilkhomjon, D.-Y. Kim, and D.-S. Park, "Integration of Deep Reinforcement Learning with Collaborative Filtering for Movie Recommendation Systems," Appl. Sci., vol. 14, no. 3, p. 1155, 2024, doi: 10.3390/app14031155.

[29] M. N. Noori, J. Ahamed, and M. Ahmed, "Matrix Factorization and Cosine Similarity Based Recommendation System For Cold Start Problem in E-Commerce Industries," Int. J. Comput. Digit. Syst., vol. 15, no. 1, pp. 775–787, 2024, doi: 10.12785/ijcds/150156.

[30] M. A. Abbas, S. Ajayi, M. Bilal, A. Oyegoke, M. Pasha, and H. T. Ali, "A deep learning approach for context-aware citation recommendation using rhetorical zone classification and similarity to overcome cold-start

problem," J. Ambient Intell. Humaniz. Comput., vol. 15, no. 1, pp. 419–433, 2024, doi: 10.1007/s12652-022-03899-6.

[31] R. N. Ravikumar, S. Jain, and M. Sarkar, "AdaptiLearn: real-time personalized course recommendation system using whale optimized recurrent neural network," Int. J. Syst. Assur. Eng. Manag., 2024, doi: 10.1007/s13198-024-02301-2.

[32] "Movielens-100k Dataset Link:" [Online]. Available: https://files.grouplens.org/datasets/movielens/ml-100k.zip

[33] "Movielens-1M Dataset Link:" [Online]. Available: https://files.grouplens.org/datasets/movielens/ml-1m.zip

[34] "FilmTrust Dataset Link:" [Online]. Available: https://guoguibing.github.io/librec/datasets/filmtrust.zip

[35] "Ciao Dataset Link:" [Online]. Available: https://guoguibing.github.io/librec/datasets/CiaoDVD.zip