An Intelligent Recommendation System Utilizing a Hybrid Deep Learning Method

Abu Tholib
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia | Informatics Engineering, Faculty of Engineering, Universitas Nurul Jadid, Indonesia
abu.tholib.2305349@students.um.ac.id

Triyanna Widiyaningtyas
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia
triyannaw.ft@um.ac.id (corresponding author)

Didik Dwi Prasetya
Department of Electrical Engineering and Informatics, Universitas Negeri Malang, Indonesia
didikdwi@um.ac.id

Received: 18 May 2025 | Revised: 9 June 2025 | Accepted: 21 June 2025

 DOI: https://doi.org/10.48084/etasr.12230

ABSTRACT

Recommender systems play a crucial role in enhancing user experience by providing personalized product suggestions, attempting to increase sales and company profitability. However, current methodologies encounter two significant limitations: i) unidirectional models for review information extraction are unable to capture complex contextual semantics effectively, and ii) inefficiency when applied to large-scale datasets. The aim of this research is to develop a novel Hybrid Deep Learning and Probabilistic Matrix Factorization (HD-PMF) model to address the data sparsity problem in recommender systems. This research method is a combination of Bidirectional Long Short-Term Memory (BiLSTM) to capture contextual semantics from user reviews, Stacked Denoising Autoencoder (SDAE) to extract robust latent features from user data, and PMF optimized using Stochastic Gradient Descent (SGD) for accurate rating prediction. The results of this research are based on experiments conducted on two benchmark datasets with high sparsity levels: MovieLens 1M (95.35%) and Amazon Information Video (AIV) (99.98%). The HD-PMF model achieves a Root Mean Square Error (RMSE) of 0.4864, significantly outperforming baseline models such as PMF, Collaborative Deep Learning (CDL), LSTM-PMF, and Dual Deep Learning (DDL)-PMF. These results demonstrate that HD-PMF is an effective and promising approach for improving recommendation accuracy.

Keywords-recommender system; hybrid deep learning; probabilistic matrix factorization

I. INTRODUCTION

In recent years, recommender systems have developed rapidly and have been utilized in various applications such as recommendations for fashion [1], music [2], online shopping, travel [3, 4], dining [5], clothing [6], movies [7, 8], education [9], and news [10]. Major companies such as Alibaba, eBay, Amazon, Flipkart, and Netflix now rely on recommendation engines to increase user engagement, boost sales, and improve profitability by delivering highly relevant product suggestions [11, 12]. These systems assist users in making informed decisions when buying products, based on historical data [13]. Recommendation systems are commonly categorized into collaborative filtering, content-based filtering, and knowledge-based approaches. However, despite significant progress, several issues persist, including the cold-start problem, data sparsity, and model performance [14]. Among these, data sparsity remains a critical obstacle, often resulting in degraded recommendation quality and reduced user satisfaction. In highly sparse environments, where only a small subset of items receives user ratings, models struggle to generate accurate predictions. This negatively impacts user engagement, discovery of relevant content, and ultimately customer retention and platform profitability.

To address the sparsity issue, some studies have incorporated auxiliary information such as user demographics (e.g., gender, age) and item metadata (e.g., brand, category) to enrich feature representations and alleviate sparsity-related limitations [15, 16]. However, the core challenge lies in the limited availability of rating data, which reduces the efficacy of conventional recommendation algorithms. Authors in [17] used Probabilistic Matrix Factorization (PMF), which models user-item interactions probabilistically; however, PMF alone struggles with learning complex feature relationships in sparse data. To improve PMF's effectiveness, authors in [18] proposed using autoencoders to capture complex features and overcome PMF's weaknesses. Later, authors in [19] developed the PHD-PMF by integrating the Dirichlet process to capture more complex variations in user preferences, although this model was still not optimal in handling changes in user preferences. To address such changes, authors in [20] developed the Long Short-Term Memory (LSTM)-PMF approach to model user interactions over time. However, this method still faced challenges related to interpretability and computational efficiency. More recently, authors in [21] introduced Dual Deep Learning (DDL)-PMF as a traditional latent factor model to represent product documents, which was effective in addressing data sparsity in ratings. Moreover, authors in [22] proposed Hybrid Content Fuzzy (HCF) recommender, a cloud model-based approach that predicts unrated items by enhancing item similarity, thereby increasing the number of co-rated items among users and improving recommendation performance under sparse conditions. While these models offer improved handling of sparse ratings, two critical gaps remain unaddressed: i) existing models often process reviews sequentially in a single direction, limiting their ability to capture nuanced contextual semantics; and ii) optimization via Maximum A Posteriori (MAP) estimation in PMF incurs high computational complexity, rendering it inefficient for large-scale applications [23].

To address these issues, this study proposes a Hybrid Deep Learning (HD)-PMF model aimed at mitigating data sparsity in recommender systems. The HD-PMF framework combines Bidirectional LSTM (BiLSTM) networks to capture comprehensive contextual semantics from product reviews, a Stacked Denoising Autoencoder (SDAE) to learn robust latent features, and PMF for accurate rating prediction, optimized using Stochastic Gradient Descent (SGD), which allows for efficient, scalable training through iterative parameter updates. These contributions aim to advance the performance of recommender systems under sparse data conditions, improving both prediction accuracy and computational feasibility.

II. RESEARCH METHODOLOGY

A. *Bidirectional Long Short-Term Memory (Bi-LSTM)*

A Bi-LSTM network is a neural architecture commonly used in Natural Language Processing (NLP) to capture sentence representations by considering contextual information from both directions, left-to-right and right-to-left [24, 25]. This architecture consists of two independent LSTM layers: the forward LSTM processes the input sequence from beginning to end, while the backward LSTM processes it in reverse. Accordingly, for each word, two vectors are generated based on reading direction, namely $\overleftarrow{h}_t$ and $\overrightarrow{h}_t$. These are then concatenated to form the final hidden representation:

$$h_t = \overleftarrow{h}_t \oplus \overrightarrow{h}_t \qquad (1)$$

The resulting vector h_t captures contextual dependencies from both directions, providing a richer semantic representation of each word. The output from the first Bi-LSTM layer can be passed to a second Bi-LSTM layer for deeper contextual encoding. Figure 1 illustrates the Bi-LSTM architecture. The sequence $\{\mathrm{h}_0 \rightarrow \mathrm{h}_1 \rightarrow \mathrm{h}_2 \ldots . \rightarrow \mathrm{h}_n\}$ represents the sequence of hidden states produced by the forward LSTM, while $\{\mathrm{h}_n \rightarrow \ldots . \mathrm{h}_2 \rightarrow \mathrm{h}_1 \rightarrow \mathrm{h}_0\}$ represents the sequence of hidden states produced by the backward LSTM.

B. *Stacked Denoising Autoencoder (SDAE)*

SDAE, first introduced in [26], is employed in this study, utilizing its stacked autoencoder architecture, which can extract important features from user information and convert them into more informative latent factors. It achieves this by corrupting the input data X with random noise, producing $\bar{X}$, and training the model to reconstruct the original input, thereby learning robust hidden patterns. $\bar{X}$ is then passed through the encoder layer:

$$h_1 = g\,(C_1 \bar{X} + b_1) \qquad (2)$$

The decoding process reconstructs the original input and predicts the target using the following expressions:

$$\hat{R} = f\,(C_L h_L + b_{\hat{R}\,i}), \hat{X} = f\,(Q_L h_L + b_{\hat{X}}) \qquad (3)$$

where function $f()$ represents the second nonlinear activation function used in the decoding stage, and the outputs $\hat{R}$ and $\hat{X}$ correspond to the predicted target and the reconstructed input, respectively. The SDAE architecture comprises two symmetric parts: an encoder and a decoder, each consisting of $L/2$ layers. The encoder transforms the corrupted input into a latent representation, while the decoder reconstructs the original input from this latent space. The overall architecture is illustrated in Figure 2.

C. *Hybrid Deep Learning and Probabilistic Matrix Factorization (HD-PMF)*

Figure 3 presents the architecture of the proposed HD-PMF approach. This method leverages multiple data sources, including user ratings, demographic attributes, and product reviews, to enhance latent feature learning and personalization. The overall process of the proposed HD-PMF approach is given in Algorithm 1.

The PMF method is widely used in recommendation systems to predict user ratings for items based on existing rating patterns. It represents users and items as vectors in a low-dimensional latent space, allowing the model to identify hidden patterns in user-item interactions, even when the data is very sparse. The PMF algorithm decomposes the rating matrix R into two latent factor matrices: a user matrix U and an item matrix V. These latent factors are learned by minimizing the difference between observed and predicted ratings, while SGD is used to optimize the parameters. Notably, the user and item vectors are updated during training based on the error between actual and predicted ratings.

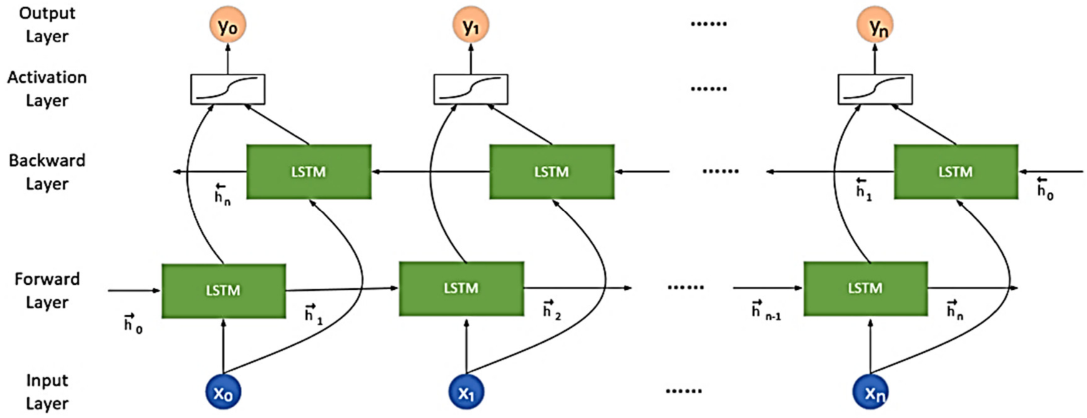

Fig. 1. Architecture of Bi-LSTM.

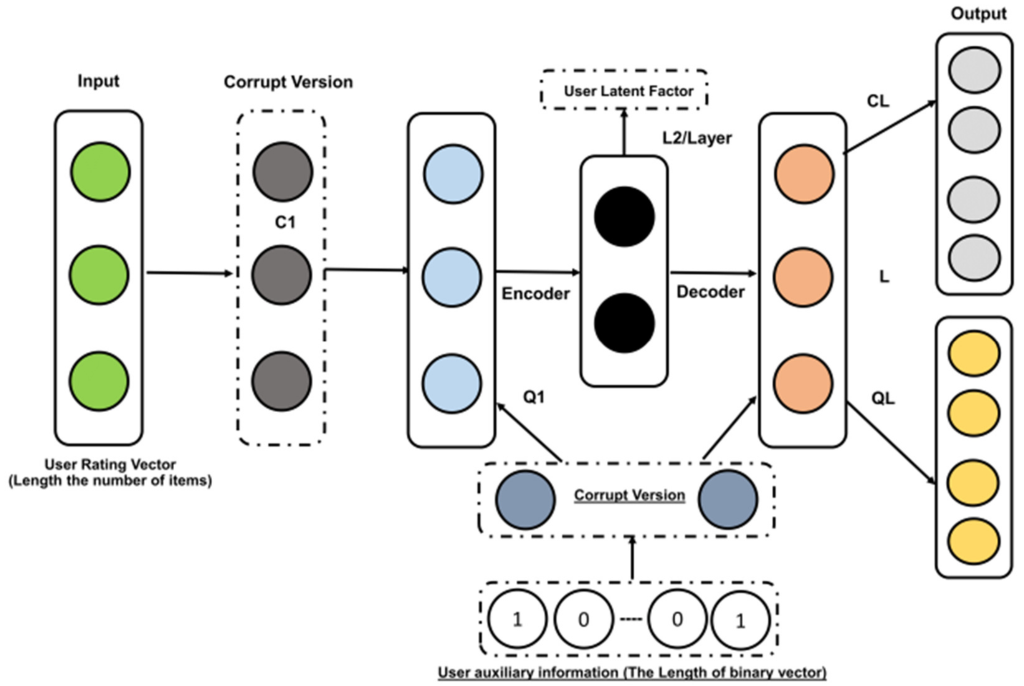

Fig. 2. The architecture of SDAE.

Algorithm 1: Proposed approach using HD-PMF

Input:

User latent factor matrix $\{U_i\}$

Item latent factor matrix $\{V_i\}$

Observed rating matrix $\{R_{u,i}\}$

Output:

Predicted rating $\hat{R}_{u,i}$ for each user-item pair

Method

Randomly initialize user latent factors $\{U_i\}$ and item latent factors $\{V_i\}$ a normal distribution scaled by the number of latent factors k, using (4):

$$U_{i,k} \sim N\,(0, 1/\mathrm{k}), V_{j,k} \sim N\,(0, 1/\mathrm{k}) \quad (4)$$

Calculate the predicted rating by performing a dot product between the user and item latent vectors using (5):

$$\hat{r}_{u,i} = U_u . V_i^T \quad (5)$$

Calculate the prediction error:

$$e_{u,i} = r_{u,i} . \hat{r}_{u,i} \quad (6)$$

Update the user and item vectors using SGD:

$$U_u \leftarrow U_u + \alpha\left(e_{u,i}\, V_i - \lambda U_u\right)$$

$$V_i \leftarrow V_i + \alpha\,(e_{u,i}\, U_u - \lambda V_i) \quad (7)$$

Generate the final predicted rating:

$$\hat{R}_{u,i} = U_u . V_i^T \quad (8)$$

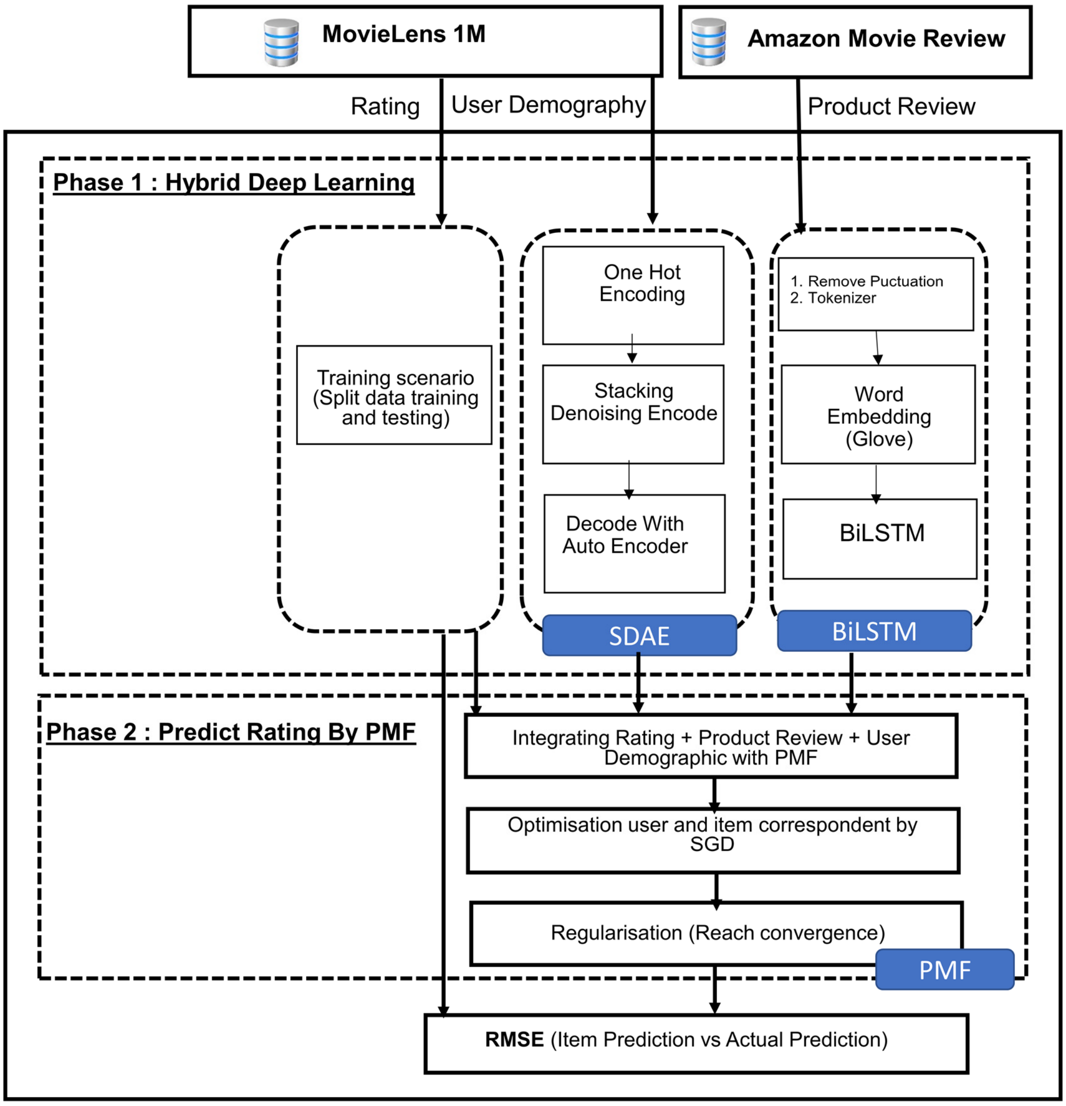

Fig. 3. Proposed method using HD-PMF.

D. Dataset

In this study, two widely used datasets were employed to evaluate the effectiveness of the proposed hybrid model for recommender systems: the MovieLens (ML-1M) dataset [27] and the Amazon Information Video (AIV) dataset [28]. Detailed characteristics of the datasets are shown below in Table I.

TABLE I. DATASET CHARACTERISTICS

Dataset	Number of users	Number of items	Rating & review	Sparse level (%)
AIV	81,339	18,203	238,352	99.98%
MovieLens	6,040	3,544	993,482	95.35%

The ML-1M dataset serves as a benchmark for collaborative filtering and contains explicit user ratings on movies. It includes various user attributes such as gender, age, occupation, and zip code. The dataset also provides item-related information, including movie titles and genres. The rating matrix consists of explicit ratings on a scale from 1 to 5 stars, reflecting user preferences for specific movies. However, the dataset exhibits a sparsity level of 95.35%, indicating that a substantial portion of possible user-item interactions remain unrecorded. To complement the structured ratings from ML-1M, the AIV dataset was utilized, which provides detailed textual reviews of movies. This dataset contains user demographic characteristics along with item-specific information in the form of user reviews and product ratings. Unlike ML-1M, the AIV dataset exhibits an extremely high sparsity level of 99.98%.

E. Experimental Settings

In this study, the ML-1M dataset was partitioned into training data (60%), cross-validation (20%), and test (20%) subsets. Additionally, to evaluate the model's performance under different scenarios, several alternative ratios were tested: 70:30%, 80:20%, and 90:10%, where the first percentage corresponds to the training data portion, while the remaining portion is used for cross-validation and testing. The train_test_split function from the scikit-learn library was employed for data partitioning, using a random_state of 42 to ensure reproducibility and consistency across experiments.

The model is initialized with key hyperparameters: the number of latent factors was set to 50; the learning rate was fixed at 0.001 to control the step size during updates via the SGD optimizer; and the regularization parameter was set to 0.02 to mitigate overfitting by penalizing large latent factor values. User and item latent factor matrices were initialized with values sampled from a scaled normal distribution to promote numerical stability during training. The training process consists of 20 epochs, where in each iteration, latent factors are updated based on observed interactions in the training dataset.

F. Evaluation Metrics

Performance metrics are critical for evaluating the effectiveness of recommender systems in delivering relevant suggestions to users [29]. One widely adopted metric is the Root Mean Square Error (RMSE), which is particularly useful for addressing the challenges of data sparsity [30]. RMSE quantifies the deviation between predicted and actual ratings; a lower RMSE indicates higher predictive accuracy [15]. The RMSE is computed using (9):

$$RMSE = \sqrt{\frac{1}{N}} \sum_{i,j} Z_{i,j}^{p} (R_{u,i} - \hat{R}_{u,i}) \qquad (9)$$

where N denotes the total number of predicted user–item pairs in the test set, $R_{u,i}$ is the actual rating provided by user u for item i, and $\hat{R}_{u,i}$ ic corresponding predicted rating.

III. RESULTS AND DISCUSSION

The progression of validation loss of the proposed model across the 20 epochs is illustrated in Figure 4. In the initial 5 epochs, this loss value is relatively stable around 0.7030, indicating minimal improvement in the initial training phase. Between epochs 6 and 10, a slight decrease is observed, indicating that the model starts to generalize better. A significant decrease occurs from epoch 11 onwards, reaching 0.4775 at epoch 20, indicating improved learning and better representation of the underlying data patterns. This consistent decrease in validation loss indicates a well-converged training process, with the potential for further improvement in prediction accuracy.

Table III presents a comparative analysis of RMSE values across various models under different training data ratios (60%–90%). The baseline PMF model consistently exhibits the highest RMSE, ranging from 0.9589 (60%) to 0.9045 (90%), demonstrating limited ability to handle data sparsity. Collaborative Deep Learning (CDL) and PHD-MF offer moderate improvements, reducing RMSE to 0.8776 (90%). Deep learning-based models such as LSTM-PMF and DDL-PMF achieve further enhancements with RMSE values of 0.8607 and 0.8583 (90%), respectively. In contrast, the proposed HD-PMF model achieves the best performance, with RMSE decreasing from 0.6791 (60%) to 0.4864 (90%). This substantial improvement highlights the model's superior capability in learning from sparse data and delivering accurate predictions. The consistent decline in RMSE with increasing training data demonstrates the model’s strong generalization ability.

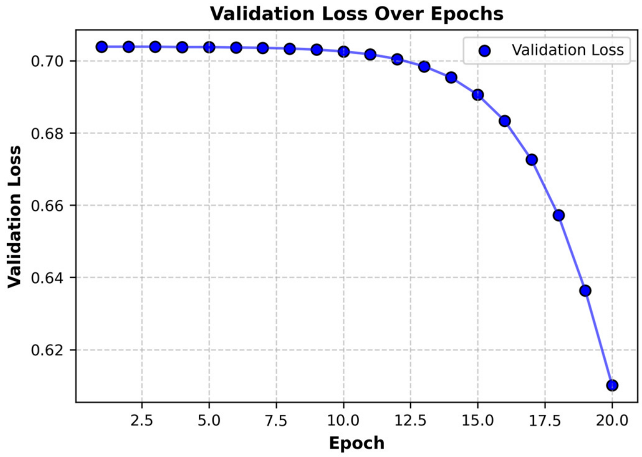

Fig. 4. Validation loss.

TABLE II. COMPARISON RESULTS OF HD-PMF OVER PMF, CDL, PHD-PMF, LSTM-PMF, DDL-PMF

Method	Ratio 60:40%	Ratio 70:30%	Ratio 80:20%	Ratio 90:10%
PMF [17]	0.9589	0.9336	0.9157	0.9045
CDL [18]	0.8996	0.8948	0.8915	0.8776
PHD-PMF [15]	0.8893	0.8814	0.8751	0.8691
LSTM-PMF [17]	0.8871	0.8763	0.8687	0.8607
DDL-PMF [21]	0.8809	0.8727	0.8687	0.8583
Proposed HD-PMF	0.6791	0.6095	0.5731	0.4864

The results presented in Figure 5 compare the RMSE values across various recommender models and train-test ratios. Traditional PMF demonstrated the highest RMSE values across all training ratios, consistent with previous studies highlighting its limitations in effectively handling sparse data. CDL and PHD-MF offer slight improvements over PMF, but their RMSE values remain high, reflecting limited capability in modeling complex user-item interactions. Deep learning-based methods such as LSTM-PMF and DDL-PMF perform better, yet still yield RMSE values above 0.85, highlighting the need for improved predictive accuracy. The proposed HD-PMF outperforms all baseline methods across all training ratios, achieving an RMSE as low as 0.4864 (90:10%). This significant reduction highlights the advantage of hybrid deep learning techniques in mitigating data sparsity and enhancing prediction accuracy. Despite these promising results, the HD-PMF model has several limitations. These include high computational complexity on large-scale datasets, reliance on extensive hyperparameter tuning, and slower convergence rates, particularly with small learning rates.

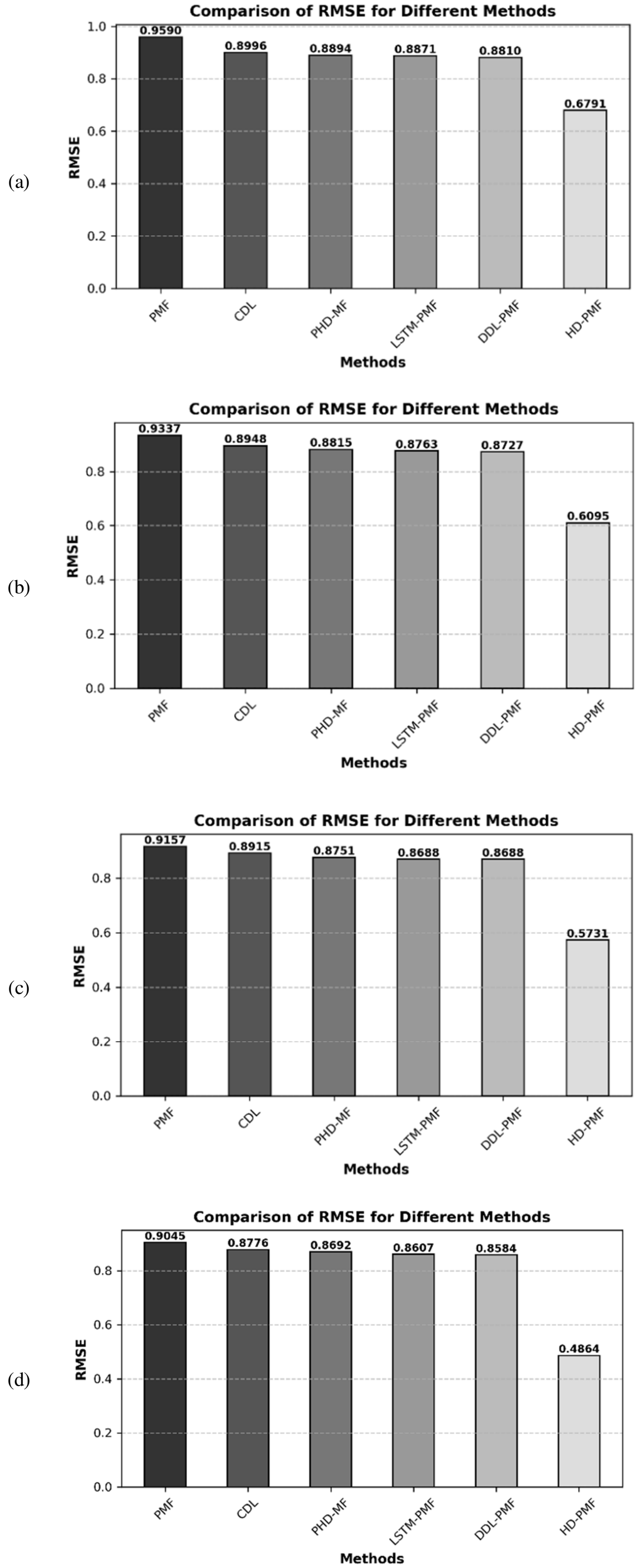

Fig. 5. Comparison of RMSE with ratio (a) 60:40%, (b) 70:30%, (c) 80:20%, and (d) 90:10%.

IV. CONCLUSION

This study introduced the Hybrid Deep Learning and Probabilistic Matrix Factorization (HD-PMF) model to address data sparsity in recommender systems. The MovieLens (ML-1M) and the Amazon Information Video (AIV) datasets were employed, which in total comprise 21,747 items and 1,321,834 reviews. The proposed model was compared in performance against the standalone PMF, Collaborative Deep Learning (CDL), PHD-PMF, Long Short-Term Memory (LSTM)-PMF, and Dual Deep Learning (DDL)-PMF, across different train-test ratios (60:40%, 70:30%, 80:20%, and 90:10%). Experimental results show that HD-PMF outperforms baseline methods, achieving the lowest Root Mean Square Error (RMSE) of 0.4864 (90:10% train-test ratio), compared to PMF (0.90452) and CDL (0.8776).

Future research will explore the incorporation of Variational Autoencoders (VAEs) to further improve scalability and convergence. VAE's probabilistic latent variable modeling may reduce sensitivity to hyperparameter tuning and offer better robustness in handling extreme sparsity in large-scale datasets. Additionally, future work will aim to optimize the model architecture and extend its application to diverse domains, including education, e-commerce, and healthcare, to evaluate its scalability and generalizability across heterogeneous data distributions and real-world scenarios.

ACKNOWLEDGMENT

This research was supported by Universitas Negeri Malang, Research Grant: "Penelitian Internal UM", scheme: "Kompetitif Topik KBK" with contract number 4.4.843/UN32.14.1/LT/2024.

REFERENCES

[1] L. De Divitiis, F. Becattini, C. Baecchi, and A. Del Bimbo, "Disentangling Features for Fashion Recommendation," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 19, no. 1s, pp. 1–21, Feb. 2023, https://doi.org/10.1145/3531017.

[2] Z. Liu, W. Xu, W. Zhang, and Q. Jiang, "An emotion-based personalized music recommendation framework for emotion improvement," *Information Processing & Management*, vol. 60, no. 3, May 2023, Art. no. 103256, https://doi.org/10.1016/j.ipm.2022.103256.

[3] A. Panteli and B. Boutsinas, "Improvement of similarity–diversity trade-off in recommender systems based on a facility location model," *Neural Computing and Applications*, vol. 35, no. 1, pp. 177–189, Jan. 2023, https://doi.org/10.1007/s00521-020-05613-z.

[4] N. Chekkai and H. Kheddouci, "TOP-Key Influential Nodes for Opinion Leaders Identification in Travel Recommender Systems," in *Advances in Model and Data Engineering in the Digitalization Era*, vol. 1751, P. Fournier-Viger, A. Hassan, L. Bellatreche, A. Awad, A. Ait Wakrime, Y. Ouhammou, and I. Ait Sadoune, Eds. Cham: Springer Nature Switzerland, 2022, pp. 149–161.

[5] X. Guo, L. Wang, M. Zhang, and G. Chen, "First Things First? Order Effects in Online Product Recommender Systems," *ACM Transactions on Computer-Human Interaction*, vol. 30, no. 1, pp. 1–35, Feb. 2023, https://doi.org/10.1145/3557886.

[6] S. Nair, K. Patil, H. Waghela, and S. Pansambal, "Outfit recommendation–using image processing," *Journal of Algebraic Statistics*, vol. 13, no. 2, pp. 1699–1706, 2022.

[7] S. Airen and J. Agrawal, "Movie Recommender System Using Parameter Tuning of User and Movie Neighbourhood via Co-

Clustering," *Procedia Computer Science*, vol. 218, pp. 1176–1183, 2023, https://doi.org/10.1016/j.procs.2023.01.096.

[8] Z. Z. Darban and M. H. Valipour, "GHRS: Graph-based hybrid recommendation system with application to movie recommendation," *Expert Systems with Applications*, vol. 200, Aug. 2022, Art. no. 116850, https://doi.org/10.1016/j.eswa.2022.116850.

[9] A. T. Khomeiny, T. Restu Kusuma, A. N. Handayani, A. Prasetya Wibawa, and A. H. Supadmi Irianti, "Grading System Recommendations for Students using Fuzzy Mamdani Logic," in *2020 4th International Conference on Vocational Education and Training (ICOVET)*, Malang, Indonesia, Sep. 2020, pp. 1–6, https://doi.org/10.1109/ICOVET50258.2020.9230299.

[10] Q. Zhao, X. Chen, H. Zhang, and X. Li, "Dynamic Hierarchical Attention Network for news recommendation," *Expert Systems with Applications*, vol. 255, Dec. 2024, Art. no. 124667, https://doi.org/10.1016/j.eswa.2024.124667.

[11] G. Behera and N. Nain, "DeepNNMF:deep nonlinear non-negative matrix factorization to address sparsity problem of collaborative recommender system," *International Journal of Information Technology*, vol. 14, no. 7, pp. 3637–3645, Dec. 2022, https://doi.org/10.1007/s41870-022-00982-1.

[12] T. Widiyaningtyas, D. D. Prasetya, and H. W. Herwanto, "Time Loss Function-based Collaborative Filtering in Movie Recommender System.," *International Journal of Intelligent Engineering & Systems*, vol. 16, no. 6, 2023, https://doi.org/10.22266/ijies2023.1231.84.

[13] S. Abinaya and K. U. Kumar, "Enhancing Multicriteria-Based Recommendations by Alleviating Scalability and Sparsity Issues Using Collaborative Denoising Autoencoder," *Computers, Materials & Continua*, vol. 78, no. 2, pp. 2269–2286, 2024, https://doi.org/10.32604/cmc.2024.047167.

[14] S. Gheewala, S. Xu, S. Yeom, and S. Maqsood, "Exploiting deep transformer models in textual review based recommender systems," *Expert Systems with Applications*, vol. 235, Jan. 2024, Art. no. 121120, https://doi.org/10.1016/j.eswa.2023.121120.

[15] L. Guo, L. Sun, R. Jiang, Y. Luo, and X. Zheng, "Recommendation based on attributes and social relationships," *Expert Systems with Applications*, vol. 234, Dec. 2023, Art. no. 121027, https://doi.org/10.1016/j.eswa.2023.121027.

[16] B. Kumar, N. Sharma, B. Sharma, N. Herencsar, and G. Srivastava, "Hybrid Recommendation Network Model with a Synthesis of Social Matrix Factorization and Link Probability Functions," *Sensors*, vol. 23, no. 5, Feb. 2023, Art. no. 2495, https://doi.org/10.3390/s23052495.

[17] R. Salakhutdinov and A. Mnih, "Probabilistic Matrix Factorization," in *Proceedings of the 21st International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2007, pp. 1257–1264.

[18] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative Deep Learning for Recommender Systems," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Sydney NSW Australia, Aug. 2015, pp. 1235–1244, https://doi.org/10.1145/2783258.2783273.

[19] J. Liu, D. Wang, and Y. Ding, "PHD: A Probabilistic Model of Hybrid Deep Collaborative Filtering for Recommender Systems," in *Proceedings of the Ninth Asian Conference on Machine Learning*, Yonsei University, Seoul, Republic of Korea, Nov. 2017, vol. 77, pp. 224–239.

[20] Hanafi and B. Mohd Aboobaider, "Word Sequential Using Deep LSTM and Matrix Factorization to Handle Rating Sparse Data for E-Commerce Recommender System," *Computational Intelligence and Neuroscience*, vol. 2021, no. 1, Jan. 2021, https://doi.org/10.1155/2021/8751173.

[21] Hanafi *et al.*, "Handling Sparse Rating Matrix for E-commerce Recommender System Using Hybrid Deep Learning Based on LSTM, SDAE and Latent Factor," *International Journal of Intelligent Engineering and Systems*, vol. 15, no. 2, pp. 379–393, Apr. 2022, https://doi.org/10.22266/ijies2022.0430.35.

[22] H. I. Alshbanat, H. Benhidour, and S. Kerrache, "A survey of latent factor models in recommender systems," *Information Fusion*, vol. 117, May 2025, Art. no. 102905, https://doi.org/10.1016/j.inffus.2024.102905.

[23] D. E. Cahyani, A. P. Wibawa, D. D. Prasetya, L. Gumilar, F. Akhbar, and E. R. Triyulinar, "Text-Based Emotion Detection using CNN-BiLSTM," in *2022 4th International Conference on Cybernetics and Intelligent System (ICORIS)*, Prapat, Indonesia, Oct. 2022, pp. 1–5, https://doi.org/10.1109/icoris56080.2022.10031370.

[24] A. Pranolo, Y. Mao, A. P. Wibawa, A. B. P. Utama, and F. A. Dwiyanto, "Robust LSTM With Tuned-PSO and Bifold-Attention Mechanism for Analyzing Multivariate Time-Series," *IEEE Access*, vol. 10, pp. 78423–78434, 2022, https://doi.org/10.1109/access.2022.3193643.

[25] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion," *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, Dec. 2010.

[26] X. Guo and X. Cui, "A Cloud Model-Based Collaborative Filtering Approach for Sparse Data Recommendation Systems," in *2024 5th International Symposium on Computer Engineering and Intelligent Communications (ISCEIC)*, Wuhan, China, Nov. 2024, pp. 149–152, https://doi.org/10.1109/isceic63613.2024.10810141.

[27] S. Forouzandeh, K. Berahmand, and M. Rostami, "Presentation of a recommender system with ensemble learning and graph embedding: a case on MovieLens," *Multimedia Tools and Applications*, vol. 80, no. 5, pp. 7805–7832, Feb. 2021, https://doi.org/10.1007/s11042-020-09949-5.

[28] *Amazon Product Reviews* (2018). J. Ni. [Online]. Available: https://cseweb.ucsd.edu/~jmcauley/datasets/amazon_v2/.

[29] N. Idrissi and A. Zellou, "A systematic literature review of sparsity issues in recommender systems," *Social Network Analysis and Mining*, vol. 10, no. 1, Dec. 2020, https://doi.org/10.1007/s13278-020-0626-2.

[30] T. Widiyaningtyas, I. Hidayah, and T. B. Adji, "User profile correlation-based similarity (UPCSim) algorithm in movie recommendation system," *Journal of Big Data*, vol. 8, no. 1, Dec. 2021, https://doi.org/10.1186/s40537-021-00425-x.