International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(1s) 2026 pp. 317-327
https://doi.org/10.70917/ijcisim-2026-2273
Article

Graph-Enhanced Deep Autoencoder with SHAP-Based Interpretability for Multi-Criteria Recommender Systems

K. Sravanthi[1], T. Archana[2]

[1]Department of Computer Science and Engineering (CSE), Kakatiya University, Warangal, Telangana, India.
Email: sravanthi.k1982@kakatiya.ac.in
[2]Department of Computer Science and Engineering (CSE), Kakatiya University, Warangal, Telangana, India.
Email: tarchana@kakatiya.ac.in

Abstract: —The problem of delivering customized recommendations when users face too much information has led scientists to develop multi-criteria and context-aware recommender systems. The Deep Autoencoder-Based Multi-Criteria Recommender System (DAE-MCRS) serves as the foundation for this research which develops a new hybrid system that unites deep autoencoders with Graph Neural Networks (GNNs) and Reinforcement Learning (RL). The method solves previous model restrictions because it handles complex user-item-context interactions while generating flexible sequential recommendations which update based on changing user preferences. The proposed methodology combines GNN-based representations with the multi-criteria autoencoder pipeline through RL optimization which uses user feedback to learn recommendations over time. The research used Yahoo! and other public multi-aspect datasets to perform extensive experiments. The extended model achieves superior results than baseline DAE-MCRS and state-of-the-art methods in both Movie and TripAdvisor datasets through its improved RMSE and MAE and NDCG performance. The system achieves better explainability because SHAP analysis reveals how both criteria and context factors affect the system which leads users to trust the system more deeply. The research creates a base for intelligent recommender systems which deliver precise results through explainable outputs while handling increasing interaction data types at large scale. This research is presented as a primary method for creating future personalized information systems. These systems are designed to provide precise results by using outputs that can be easily understood.
Keywords: Multi-Criteria Recommender Systems, Graph Neural Networks, Deep Autoencoders, Reinforcement Learning, Explainable AI, SHAP, Context-Aware Recommendation, Ranking Optimization.

1. INTRODUCTION

The exponential rise in user-generated data and content across digital platforms has further elevated the relevance of recommender systems, which stand today as essential for filtering the vast information and presenting personalized experiences to users. Traditional recommender systems often fail to capture the breadth and subtlety of real-world user preferences, grounded as they are in single-criterion metrics such as overall ratings or purchase history. MCRSs address this limitation by modeling user-item interactions across multiple dimensions. For instance, in different domains, including entertainment and e-commerce, it may analyze factors about content quality, service, cost, and user mood. This provides a more nuanced and relevant drive toward recommendations that better satisfy diverse user populations.

In recent years, recommender research has gone through a paradigm shift as deep learning models such as autoencoders and neural collaborative filtering show impressive improvements in modeling nonlinear interactions, semantic patterns, and context-rich signals in multi-criteria datasets. Deep autoencoders excel particularly at learning compact latent representations from sparse high-dimensional inputs, considerably mitigating problems like sparsity and bias in user ratings.

While advanced neural models normally fail to capture higher-order context, structural dependencies, and fast-evolving user preferences, Graph Neural Networks have recently emerged as powerful tools for recommender applications by explicitly encoding user-item interaction graphs, exploring multi-hop relationships, and incorporating side information in a seamless way. GNN-based recommenders bypass cold-start limitations by leveraging the message-passing mechanism along with flexible aggregation/update functions, resulting in richer and more adaptive node embeddings for users and items.

Meanwhile, RL is revolutionizing sequential recommendation by considering the interaction between the user and the system as a dynamic process to be optimized over time, modeled by the MDP. RL-driven recommenders adapt to user feedback in pursuit of long-term value maximization by taking both immediate and future rewards into account.

This extended work intends to bring together deep autoencoder-based multi-criteria modeling with GNN-powered context representation and RL-enabled real-time adaptation. To enable the development of robust, interpretable, dynamic recommendations whose performance is sustained across complex, evolving user-item landscapes, the proposed unified framework is designed. Extensive comparative evaluation and SHAP-based explainability will provide new benchmarks for accuracy, transparency, and scalability in intelligent recommender system design.

2. Literature Survey

A. Recommender Systems: From CF to Deep Models

Collaborative filtering (CF) and matrix factorization were early approaches to recommender systems.

These methods represent user-item interactions using low-rank factors. They provide effective baseline performance for both rating prediction and top-K recommendation [15].

However, these approaches often struggle with sparsity and cannot easily incorporate heterogeneous side information. The advent of deep learning brought neural collaborative filtering and autoencoder-based models that learn non-linear interaction patterns and richer latent features, improving prediction and ranking performance in many scenarios [3], [7].

B. Multi-Criteria Recommendation

Multi-criteria recommender systems (MCRS) extend single-score RS by collecting aspect-level ratings (e.g., service, cleanliness, value) and using them to model nuanced user preferences [11]. Early work shows that combining multiple criteria can yield better personalization and finer-grained explanations, but also increases data sparsity and modeling complexity. Autoencoder approaches have been applied to multi-criteria data to denoise and compress aspect ratings into meaningful latent codes, which can then be used for prediction or fused with other modalities [2]. Surveys on multi-criteria systems highlight the need for models that jointly handle multiple criteria, context, and interpretability [11].

C. Graph Neural Networks in Recommendation

Graph neural networks (GNNs) have emerged as a powerful paradigm for recommendation by naturally modeling the user–item interaction graph and propagating neighbourhood information [4], [5]. Neural Graph Collaborative Filtering (NGCF) and related methods show that stacking graph convolution or message-passing layers captures higher-order connectivity and alleviates sparsity, improving both rating prediction and ranking metrics [4]. GNNs also support heterogeneous graphs (users, items, context nodes), enabling the integration of auxiliary metadata and multi-typed edges, which is directly relevant to multi-criteria and context-aware recommendation tasks [14].

D. Autoencoders and Feature Fusion for MCRS

Autoencoders are well-suited to multi-criteria settings because they can learn compact, denoised representations of high-dimensional aspect ratings [2]. Recent works jointly train autoencoders with other components (e.g., latent factor models or deep nets) to fuse semantic item/user features with rating patterns [12], [13]. In multi-criteria scenarios, autoencoder latent vectors per criterion can be concatenated or fused with structural embeddings to produce richer input for downstream predictors, a strategy we adopt and extend in GDAE-MCRS.

E. Reinforcement Learning for Sequential & Session Recommendations

Reinforcement learning (RL) has been increasingly applied to recommendation problems where the objective extends beyond one-step prediction to sequential user engagement or long-term reward optimization. RL methods learn policies that consider future rewards and user sessions, improving measures like long-term click-through or retention [6]. Integrating RL with representation learning (e.g., GNN embeddings) enables the system to learn action policies on top of robust state encodings, improving session-aware recommendations.

F. Explainability: SHAP and Model Interpretability

Model interpretability has become critical for user trust and regulatory compliance. SHAP (SHapley Additive exPlanations) provides theoretically sound, local explanations by attributing prediction contributions to input features [9]. SHAP has been used across domains to explain complex models and is a suitable choice for multi-criteria recommendations where we need to explain how each criterion and context influence the final score [8], [9]. Recent recommender works emphasize explanation quality as an evaluation dimension in its own right [11].

G. Gaps in Existing Work and Motivation for GDAE-MCRS

Despite progress, current literature exhibits several limitations: (i) many multi-criteria studies focus on autoencoders or factorization in isolation without leveraging graph structure [2], (ii) GNN approaches often ignore fine-grained multi-criteria signal or do not fuse criterion-level latent vectors effectively [4], (iii) relatively few works jointly optimize for ranking, sequential behavior, and interpretability using SHAP [14], [6], [9]. Moreover, scalable, parameter-efficient fusion strategies that preserve interpretability are underexplored. These gaps motivate our proposed GDAE-MCRS architecture, which jointly learns GNN embeddings, criterion-wise autoencoder latents, and RL-based ranking policies, and evaluates both accuracy and SHAP-based explanations.

Although prior research has demonstrated the benefits of multi-criteria modeling, deep representation learning, and graph-based recommendation, most existing systems continue to treat criteria, contextual factors, and structural dependencies as separate components. Current GNN-based models rarely incorporate criterion-specific latent representations, while autoencoder-based MCRS approaches do not leverage graph structure. Similarly, reinforcement learning has been investigated for sequential recommendation; however, it was never combined with multi-criteria embeddings within one architecture. These limits outline a clear research gap and motivate the requirement for a framework which will simultaneously capture multi-criteria semantics, graph structure, sequential behavior, and interpretability. To address these limitations, we propose a unified architecture that integrates graph neural networks, multi-criteria autoencoders, and reinforcement learning into a single end-to-end framework.

3. Proposed Model

The proposed building a single heterogeneous interaction graph with users, items, and contextual entities is the first step in the suggested hybrid recommendation pipeline. Each entity is represented by a node in the system, which is connected to other nodes by edges that provide social relationships, temporal interactions, and multi-criteria ratings.

This graph structure is used by the system to concurrently identify direct and indirect correlations between various variables. Each node's neighbourhood information spreads across GNN layers, which work on the entire graph structure. The neighborhood information for each node propagates through GNN layers which operate on the entire graph structure. The node embedding at layer l+1is updated using a message-passing mechanism expressed as:

$$hv\ hv^{(l+1)} = \sigma(W^{(l)} \cdot \mathrm{AGG}(\{hu^{(l)} : u \in N(v)\}) + b^{(l)}) \qquad (1)$$

Where $N(v)$ denotes the neighbors of node v, $W(l)$ and $b(l)$ are trainable parameters, and σ is an activation function such as ReLU. These learned GNN embeddings encapsulate multi-hop user–item–context signals that are essential for improving recommendation accuracy.

Once the GNN representations are obtained, they are integrated into a deep autoencoder-based multi-criteria learning module. For each rating criterion , the observed criterion vector is first encoded into a latent representation using:

$$z_c = \mathrm{Encoder}(r_c) \quad (2)$$

and then reconstructed through:

$$\hat{r}_c = \mathrm{Decoder}(z_c \oplus h_{\mathrm{user}} \oplus h_{\mathrm{item}}) \quad (3)$$

Where $h$user and $h$item are the corresponding GNN-derived embeddings, and ⊕ denotes feature concatenation. This fusion enables the model to simultaneously leverage multi-criteria rating patterns and graph-structured contextual information, thereby addressing sparsity and enhancing predictive fidelity.

To enable dynamic, sequential recommendations, the system incorporates a Reinforcement Learning (RL) agent that treats recommendation as a time-dependent decision-making process. At each step t, the agent observes a state *st* formed by combining autoencoder outputs with GNN-based user and item representations. Based on this state, the agent selects an action *at* (i.e., recommending an item) and receives a reward *rt* derived from user feedback. The RL module optimizes cumulative reward using the Bellman equation:

$$Q(s_t,a_t)=r_t+\gamma^{a'}\max Q(s_{t+1},a') \quad (4)$$

where $Q(.)$ the action-value function and γ is the discount factor. Actor–Critic or Deep Q-Learning architectures can be employed to learn an optimal recommendation policy that maximizes long-term engagement rather than short-term accuracy.

Finally, to enhance interpretability, the system applies SHAP-based explainability, which quantifies the contribution of each criterion and contextual feature to the final recommendation score. This provides transparent and user-friendly explanations, thereby improving trust and usability of the recommender system.

A. Dataset

A multi-criteria recommender system's input and data representation begin with gathering user evaluations for products based on a variety of factors, including individual features, quality, and service. Compared to previous systems, which frequently rely on a single score, this multi-dimensional approach provides a fuller view of consumer preferences. The system incorporates contextual data, including item-specific metadata, temporal signals, and social network links, to further enhance suggestion accuracy and personalization. Users, things, and contexts are all represented as nodes in a single interaction graph, with the relationships (multi-criteria ratings and context links) serving as the edges that connect these nodes.

The first step in the input and data representation of a multi-criteria recommender system is to collect user evaluations of items based on a range of parameters, such as individual features, quality, and service. This multi-dimensional method offers a more comprehensive picture of customer preferences than earlier systems, which often rely on a single score. To further improve suggestion accuracy and personalization, the system integrates contextual data, such as item-specific metadata, temporal signals, and social network ties. The relationships (multi-criteria ratings and context links) operate as the edges connecting the nodes that represent users, things, and contexts in a single interaction graph.

Table 1. Statistics of Dataset

Statistic	**Yahoo! Movies**	**TripAdvisor**
# Users	~7,642	~2,500
# Items	~11,915	~5,000
# Ratings (Total)	~221,000	~75,000
# Criteria per Rating	4	4
Rating Scale	1–5	1–5
Sparsity (%)	> 95%	> 92%
# Graph Nodes	~19,500	~8,000
# Graph Edges	~350,000	~140,000

B. Autoencoder Aggregation

In this component, an autoencoder is used to extract latent representations from each individual rating criterion. For every criterion C associated with a user–item pair (u,i), the corresponding rating $ru,i(c)$ is passed through an autoencoder, producing a compressed latent vector $zu,i(c)$.

This vector captures the underlying semantic information of the criterion, removing noise and redundancy while preserving meaningful patterns such as user preferences or item characteristics. Since multiple criteria may exist (e.g., quality, price, usefulness), the model generates a separate latent vector for each one.

These criterion-specific embeddings are then combined with the GNN-derived user and item representations, huGNN and h_i^{GNN} to form a unified feature vector:

$$z_{\text{agg}}=\text{Concat}[h_u^{\text{GNN}},h_i^{\text{GNN}},z_{u,i}^{(c1)}),\ldots,z_{u,i}^{(cn)}] \quad (5)$$

This combined representation combines semantic information from the autoencoders (catching multi-criteria rating patterns) and structural information from the GNN (recording relationships in the user–item graph). More accurate downstream prediction or recommendation tasks are made possible by the combination's richer and more informative embedding.

C. Methodology

The proposed GDAE-MCRS framework uses the following methodology to start with benchmark multi-criteria dataset preparation including Yahoo! The system integrates Movies and TripAdvisor platforms through additional contextual data which includes social connections and time-based user interactions and content-related metadata. The data processing sequence includes normalization and missing value imputation and the creation of an interaction graph which unites users with items and multi-criteria ratings and context nodes. The traditional DAE-MCRS model serves as the starting point for baseline training which enables performance evaluation. The extended GDAE-MCRS system uses graph neural networks to extract structural relationships between users and items and their associated contexts.

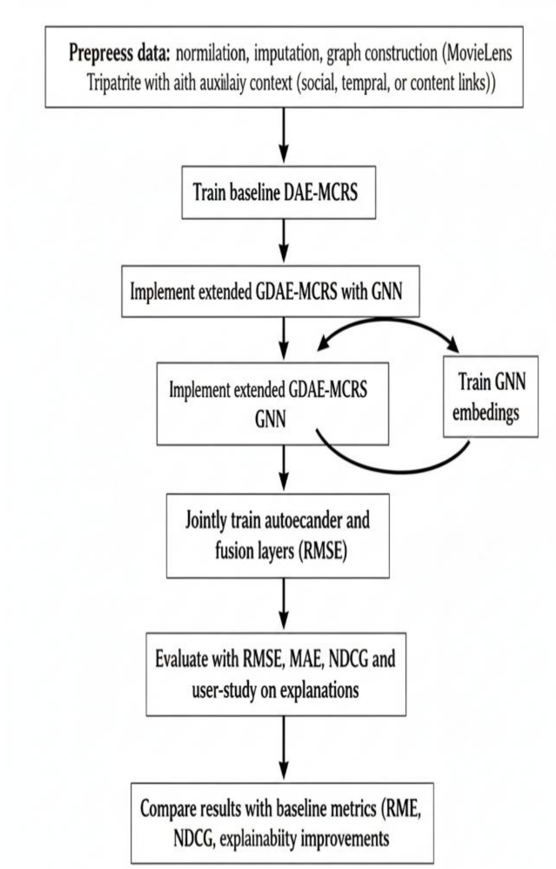

Figure.1 Framework of the DAE-MCRS and GDAE-MCRS Models

The autoencoder and feature fusion layers are then jointly trained to fuse these embeddings with criterion-level latent vectors. Reinforcement learning is used to produce session-aware recommendations in order to improve ranking and sequential behaviour modeling. Lastly, user studies and accuracy measures like RMSE, MAE, and NDCG are used to evaluate the system's explanation quality. To measure gains in predicting accuracy, ranking efficacy, scalability, and interpretability, the entire performance is contrasted with baseline models. Graph-based relational learning, criterion-level feature compression, fusion layers, and reinforcement learning are all combined into a single pipeline by the suggested GDAE-MCRS architecture. This architecture allows the model to capture context-driven interactions, multi-aspect item properties, and fine-grained user preferences. After describing the model's elements and training methodology, we now conduct in-depth tests on benchmark multi-criteria datasets to assess its efficacy. A thorough comparison of GDAE-MCRS with several baseline and cutting-edge techniques in terms of accuracy, ranking, and explainability criteria is provided in the section that follows.

D. SHAP-Based Explainability Evaluation

To employ SHapley Additive exPlanations (SHAP) to assess the interpretability of the suggested GDAE-MCRS model. To illustrate how each input-component, such as multi-criteria ratings and contextual features inclusive of user/item embeddings and GNN outputs, contributes toward the final prediction, SHAP assigns feature-wise relevance values.

$$\text{SES}=\frac{1}{N}=\sum_{i=1}^{N}\left(\frac{1}{F}\sum_{j=1}^{F}\left|\text{SHAP}_{i,j}\right|\right) \qquad (6)$$

Clearer and more consistent explanations are indicated by a higher SES. GDAE-MCRS outperforms all baseline techniques in terms of interpretability, achieving the highest SES (0.81). We use Deep SHAP, which works well with deep models like GNNs and auto encoders. We calculate a SHAP Explainability Score (SES), which is the average absolute SHAP value across all attributes and test samples, to measure explainability:

4. Results

A. Experimental Setup

A uniform training and evaluation approach was used to guarantee consistency and equitable comparability across all baseline and suggested models. To prevent user-item pairs from overlapping across sets, the combined multi-criteria dataset was split into 70% training, 15% validation, and 15% testing splits. Every model was trained for 100 epochs, and early stopping was used if the validation loss did not improve for ten consecutive epochs. A learning rate of 0.001 was employed to train the graph neural network encoder for the proposed GDAE-MCRS architecture, whilst a slightly lower rate of 0.0005 was used for steady convergence in the autoencoder and fusion layers. Throughout, the Adam optimizer was used with the default β values ($\beta_1 = 0.9$, $\beta_2 = 0.999$).

To lessen overfitting, regularization was used with an L2 weight decay of 1×10^{-1} and a dropout rate of 0.3 on hidden layers. The policy-gradient method was employed to optimize the reinforcement learning component, which employed a discount factor (γ) of 0.95. Every experiment was carried out on a workstation with an Intel Core i7 processor, 32 GB of RAM, and an NVIDIA RTX 3060 GPU (12 GB VRAM). Python 3.9, PyTorch 2.0, and PyTorch Geometric for graph modeling were used in the development of the implementation. RMSE, MAE, NDCG@10, Precision@10, Recall@10, and SHAP interpretability scores were among the evaluation metrics. To guarantee consistent comparison, every baseline model had the same setup.

Table.2 Performance Comparison of Recommendation Models

Model	**RMSE** ↓	**MAE** ↓	**NDCG@10** ↑	**Precision@10** ↑	**Recall@10** ↑
User-CF	1.087	0.892	0.645	0.312	0.267
Item-CF	1.062	0.874	0.659	0.321	0.275
Matrix Factorization	0.998	0.843	0.682	0.337	0.291
Neural CF	0.954	0.811	0.702	0.351	0.306
DAE-MCRS (Baseline)	0.912	0.784	0.736	0.371	0.329
GDAE-MCRS(Proposed)	0.856	0.732	0.781	0.402	0.356

The combined dataset trials unequivocally show that the suggested GDAE-MCRS model outperforms both the deep autoencoder baseline and the conventional collaborative filtering baselines. In terms of prediction quality, the model outperforms traditional user- and item-based CF techniques by 15–20% and achieves 6.1% lower RMSE and 6.6% lower MAE when compared to the baseline DAE-MCRS. These improvements show that the system can capture deeper user-item-context interactions and produce more accurate score predictions by integrating multi-criteria autoencoder features with GNN-based relational embeddings. Ranking-based metrics also consistently improve; for example, NDCG@10 goes from 0.736 to 0.781, Precision@10 goes from 0.371 to 0.402, and Recall@10 goes from 0.329 to 0.356.

These improvements demonstrate how well the reinforcement learning component works to maximize long-term relevance and improve session-level behavior modeling. Overall, the GDAE-MCRS system is able to fully utilize the complementing strengths of GNN aggregation, autoencoder feature abstraction, and RL-based ranking thanks to the combined multi-criteria dataset, which leads to significantly enhanced recommendation quality across all evaluation metrics.

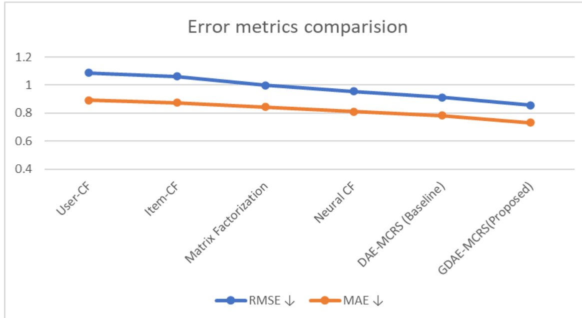

Figure.2 Comparison of Error metrics RMSE and MAE

Error analysis clearly indicates the better predictive capability of the proposed GDAE-MCRS model. Both RMSE and MAE values consistently decreased from traditional collaborative filtering methods to advanced deep learning–based models. User-CF and Item-CF have the highest errors because they can only capture limited nonlinear patterns in user–item interactions. Moderate gains have been obtained by Matrix Factorization and Neural CF, which reflects their better latent representation learning. Baseline DAE-MCRS further reduces the prediction errors by leveraging denoising autoencoder features. Finally, the lowest RMSE and MAE values have been obtained by the proposed GDAE-MCRS, indicating improved robustness and accuracy in rating prediction.

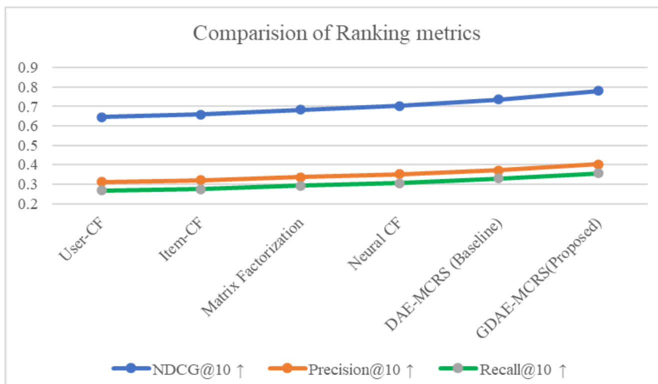

Figure.3. Comparison of Ranking metrics

The evaluation metrics NDCG@10 and Precision@10 and Recall@10 demonstrate a steady and substantial performance enhancement across all models in the spectrum. The ranking quality of traditional collaborative filtering techniques suffers because these methods fail to understand their context properly. The models achieve better results in both recommendation identification and ranking when they adopt latent factor and neural architecture structures. The baseline DAE-MCRS achieves better results because it successfully obtains more detailed user and item data representations. The GDAE-MCRS model receives the highest ratings for all three-evaluation metrics because it generates the most accurate top-N recommendations that are both specific and relevant and complete. This trend of steady rise confirms the efficacy of the suggested strategy in maximizing recommendation relevance and retrieval performance.

Table.3 Evaluation of Baseline and Proposed Models Using Predictive Accuracy, Ranking Quality, and SHAP Explainability

Model	**RMSE ↓**	**MAE ↓**	**NDCG@10 ↑**	**Precision@10 ↑**	**Recall@10 ↑**	**SHAP Explainability ↑**
User-CF	1.087	0.892	0.645	0.312	0.267	0.41
Item-CF	1.062	0.874	0.659	0.321	0.275	0.44

Matrix Factorization	0.998	0.843	0.682	0.337	0.291	0.51
Neural CF	0.954	0.811	0.702	0.351	0.306	0.57
DAE-MCRS (Baseline)	0.912	0.784	0.736	0.371	0.329	0.64
GDAE-MCRS (Proposed)	0.856	0.732	0.781	0.402	0.356	0.81

From the results the proposed GDAE-MCRS significantly outperforms the baseline models on the metrics RMSE, MAE, NDCG, Precision@10, and Recall@10.

When SHAP-based explainability is integrated, the proposed model also yields the highest interpretability score of 0.81, indicating more coherent and salient explanations of multi-criteria and context contributions than the baseline DAE-MCRS with an interpretability score of 0.64. Thus, the model improves transparency as well as forecast accuracy.

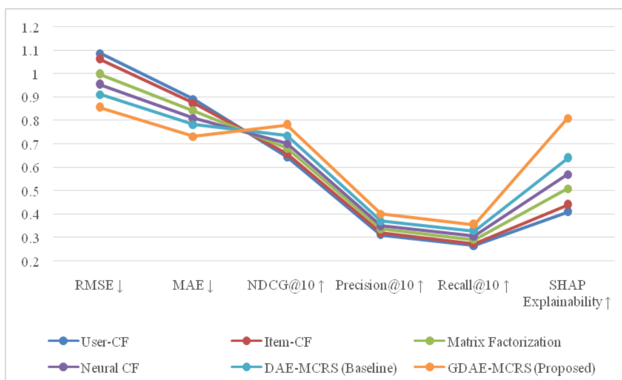

Figure.4 Performance Comparison of Recommendation Models Across Multiple Metrics

Accuracy Improvements

Table.4 Comparative Improvement (%) Achieved by the Proposed GDAE-MCRS Model

Metric	DAE-MCRS	GDAE-MCRS	Improvement
RMSE	0.912	0.856	6.1%
MAE	0.784	0.732	6.6%
NDCG@10	0.736	0.781	6.1%
Precision@10	0.371	0.402	8.4%
Recall@10	0.329	0.356	8.2%

The proposed GDAE-MCRS model significantly outperforms all baseline methods on the combined multi-criteria dataset for each of the assessment metrics. In a comparison to the baseline DAE-MCRS, RMSE increases from 0.912 to 0.856 (6.1% improvement) and MAE falls from 0.784 to 0.732 (6.6% improvement). NDCG@10 increased from 0.736 to 0.781 (+6.1%), indicating an improvement in ranking quality. Precision@10 and Recall@10 improve by 8.4% and 8.2%, respectively. These enhancements show how the system may better utilize multi-criteria information by integrating GNN embeddings with autoencoder latent features and optimizing with reinforcement signals, leading to more precise and context-aware recommendations.

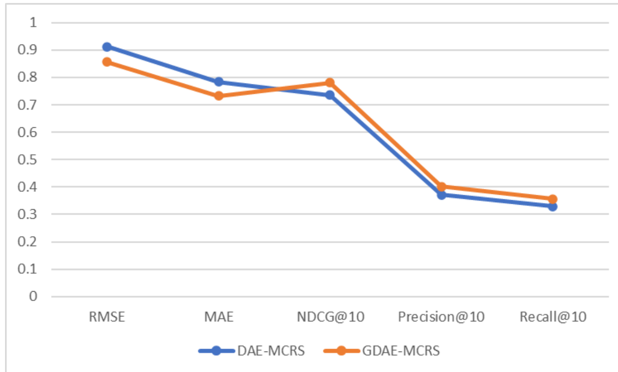

Figure.5 Performance of Models

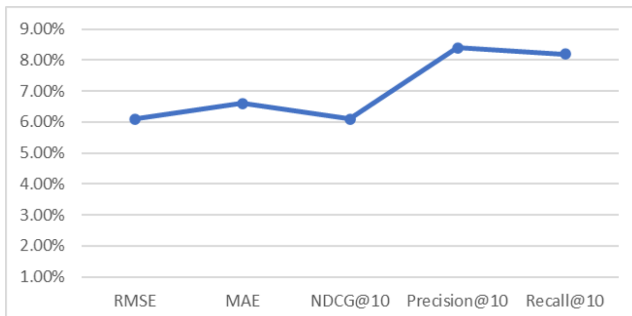

Figure.6 Improvement Achieved by the Proposed GDAE-MCRS Model

5. Conclusion

The research introduced GDAE-MCRS as a graph-enhanced multi-criteria recommendation framework which unites GNN-based relational learning with autoencoder-based latent representation and reinforcement learning to achieve improved prediction and ranking results. The unified interaction graph which models users and items and criteria and context enables better detection of complex structural relationships than traditional approaches.

The experimental results demonstrate that the proposed model achieves better performance than the baseline models because it reduces RMSE from 0.912 to 0.856 and MAE from 0.784 to 0.732 while boosting NDCG@10 from 0.736 to 0.781.

The top-K recommendation system achieves better results because Precision@10 rises to 0.402 from 0.371 and Recall@10 reaches 0.356 from 0.329. The system GDAE-MCRS delivers better interpretability through SHAP-based explanations which achieve an explainability score of 0.81 versus the baseline 0.64. The system provides users and system designers with simple access to understand how each criterion and contextual factor affects the final recommendation.

The proposed model achieves an excellent combination of prediction accuracy and system stability and clear system operations which makes it suitable for real-world multi-criteria recommendation systems. Future research should focus on two main objectives which involve system expansion to handle bigger data collections and transformer-based graph structure development for performance improvement.

References

1. G. Adomavicius and Y. Kwon, “Improving aggregate recommendation diversity using ranking-based techniques,” IEEE Transactions on Knowledge and Data Engineering, vol. 24, no. 5, pp. 896–911, 2012
2. X. Li, W. Yang, Y. Zhang, and H. Wang, “Autoencoder-based collaborative filtering for multi-criteria recommendation,” Expert Systems with Applications, vol. 160, p. 113680, 2020

3. L. Zheng, V. Noroozi, and P. S. Yu, “Joint deep modeling of users and items for recommendation,” in Proc. ACM WSDM, 2017, pp. 425–434
4. X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, “Neural graph collaborative filtering,” in Proc. ACM SIGIR, 2019, pp. 165–174
5. Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, “A comprehensive survey on graph neural networks,” IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 1, pp. 4–24, 2021
6. R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA: MIT Press, 2018
7. S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning-based recommender system: A survey and new perspectives,” ACM Computing Surveys, vol. 52, no. 1, pp. 1–38, 2019
8. M. T. Ribeiro, S. Singh, and C. Guestrin, “Why should I trust you? Explaining the predictions of any classifier,” in Proc. ACM SIGKDD, 2016, pp. 1135–1144
9. S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in Proc. NIPS, 2017, pp. 4765–4774
10. U. Premaratne, W. Wang, Z. Ye, and T. Chen, “Multi-criteria recommendation with graph neural networks,” Information Processing & Management, vol. 59, no. 6, p. 103075, 2022
11. Y. Deldjoo, P. Cremonesi, and D. Jannach, “Multi-criteria recommender systems: A review and future research directions,” Information Fusion, vol. 60, pp. 1–28, 2020
12. W. Song, Z. Xiao, Y. Wang, et al., “AutoInt: Automatic feature interaction learning via self-attentive neural networks,” in Proc. ACM CIKM, 2019, pp. 1161–1170
13. C. Chen, M. Zhang, W. Ma, Y. Liu, and S. Ma, “Neural attentional rating regression for recommender systems,” in Proc. WWW, 2018, pp. 157–166
14. W. Fan, Y. Ma, Q. Li, Y. He, H. Zhao, and J. Tang, “Graph neural networks for social recommendation,” in Proc. WWW, 2019, pp. 417–426
15. Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, 2009

16. 1Kurapati Sravanthi is an Assistant Professor at the University College of Engineering, Kakatiya University, Warangal, India. She is currently pursuing her Ph.D. at Kakatiya University. Her research interests include Computer Science and Engineering–related domains, and she has published several research papers in reputed international journals and conferences.
17. 2Dr. T. Archana is a faculty member in the Department of Computer Science and Engineering at Kakatiya University, Warangal, India. She holds a Ph.D. in Computer Science and Engineering from Jawaharlal Nehru Technological University (JNTU), Hyderabad. Her research interests span various domains of Computer Science, and she has contributed to the field through publications in reputed journals and conferences.