JTE
HCM-UTE
ISSN: 2615-9740

JOURNAL OF TECHNICAL EDUCATION SCIENCE
Ho Chi Minh City University of Technology and Engineering
Website: https://jte.edu.vn
Email: jte@hcmute.edu.vn

A Contextual-Enhanced LightGCN for Movie Recommendation Systems

Dinh-Quoc-Hoa Pham, Huyen-Trang Phan*
Ho Chi Minh City University of Technology and Engineering, Vietnam

*Corresponding author. *Email: trangpth@hcmute.edu.vn*

ARTICLE INFO

Received:	08/01/2026
Revised:	04/02/2026
Accepted:	09/02/2026
Published:	28/02/2026

KEYWORDS

Graph Convolutional Network;

Collaborative Filtering;

Contextual Awareness Recommendation;

Content Features;

Demographics.

ABSTRACT

In the context of the digital information explosion, recommender systems have been widely deployed to mitigate information overload through personalized information filtering. Traditional methods, such as collaborative filtering and content-based filtering, established the foundation for this field. Recently, advancements in deep learning particularly Graph Convolutional Network-based models such as LightGCN have demonstrated superior effectiveness in learning user and item representations from high-order interaction graph structures. To alleviate this limitation, this paper proposes a recommendation method titled Contextual-enhanced LightGCN[1]. This approach enhances the LightGCN model by simultaneously leveraging movie content features and user demographic information to aggregate information during the training process. Our ablation study further clarifies that while item content features enhance recommendation quality, the simple integration of user demographics introduces noise and degrades performance. Comprehensive experiments on MovieLens 100K and MovieLens 1M datasets, averaged over three independent runs, indicate that CF-LightGCN consistently outperforms the LightGCN baseline, achieving a Recall@20 improvement of up to 1.5%.

Doi: https://doi.org/10.54644/jte.2026.2069

Copyright © JTE. This is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License which permits unrestricted use, distribution, and reproduction in any medium for non-commercial purpose, provided the original work is properly cited.

1. Introduction

To deal with too much information on online entertainment sites, recommender systems have been made to help people find and filter personalized information [1]. ecommender systems figure out what content is most useful for users based on what they know about them, the items, and how they interact with each other. These systems are very important for movie streaming services because they make sure that each user has a unique experience on online entertainment services. At the same time, they increase user engagement and business revenue.

Currently, various methodologies exist for developing recommender systems. Most notably, three fundamental approaches include content-based filtering, collaborative filtering, and hybrid methods. [2]. Content-based filtering is an approach in which the recommender system is constructed solely based on item attributes (e.g., titles, descriptions, categories) or user profiles [3]. Consequently, this method neglects a critical feature: the interactions between users and items. In contrast, collaborative filtering [4], [5] focuses on historical user activities or preferences (e.g., user item ratings) without utilizing any explicit user or item information. Hybrid methods [6] seek to integrate the respective strengths of both aforementioned approaches.

Although the aforementioned methods have achieved significant success, they still encounter difficulties in capturing the complex and non-linear relationships within large-scale data. The emergence of Graph Convolutional Network (GCN), exemplified by LightGCN [7] , has optimized deep learning architectures to focus on information propagation across interaction graphs by eliminating feature transformations and non-linear activation functions during the layer-wise information aggregation process. In LightGCN, embedding vectors are randomly initialized, which means that the model

---

[1] https://gitlab.com/hoadaknong/cs-project/-/tree/test/LightGCN_CAR

refinement process takes a long time to converge and may be prone to overfitting. Also, the rise of Context-Aware Recommender Systems (CARS) is a more advanced method [8]. These systems use information about the situation, like the time, place, mood, and social situation, as part of the recommendation process. The CARS approach makes systems give users more useful recommendations and solves problems that come up with traditional methods, like the cold-start problem and data sparsity.

Leveraging the graph representation learning capabilities of LightGCN, we propose an attribute-based embedding initialization method. Within the scope of this research, contextual information is defined as a set of fundamental features that shape the interaction environment, comprising (1) user context via demographics and (2) item context via movie metadata. Rather than starting with randomly initialized embedding vectors – a common approach that often suffers from performance degradation when interaction data is sparse—we construct the embedding foundation directly from the entities' auxiliary information. Regarding user information, the initial embedding vectors are constructed by combining demographic features such as age, gender, and occupation. This ensures the initial placement of users with similar traits closer together in the vector space. In terms of movie details, each movie is represented by a combination of its genre and title. This approach enables the layer-wise propagation of LightGCN to effectively exploit feature relationships, thereby enhancing personalization performance while more robustly addressing the user cold-start problem. The proposed method comprises the following primary stages: (1) Contextual feature extraction; (2) Feature-enriched embedding initialization; (3) Lightweight graph propagation; (4) Hybrid recommendation.

The primary contributions of this research are summarized as follows:

- Proposed an embedding initialization method derived from contextual information for recommender systems, termed CF-LightGCN; the model initiates from meaningful representations rather than random initialization.
- CF-LightGCN was developed based on ablation studies analyzing the individual and combined contributions of content features and user demographics, demonstrating that content features yield higher effectiveness than user demographics alone or the combination of both.
- Evaluate model performance and analyze the contributions of individual feature types.

This paper is structured as follows: Section 2 describes the proposed methodology in detail, including the model architecture, the training process, and the evaluation of the experiments. Finally, the paper ends with a summary and some ideas for future research.

2. Related work

In this section, we look at some recent improvements that have been made to recommender systems, with a focus on methods based on graph neural networks (GNNs) and those that are aware of the context.

Recent improvements in recommender systems have focused more and more on collaborative filtering models based on GNN. These models leverage not only user-item interactions but also contextual information. LighGCN [7], a simpler GNN architecture made for collaborative filtering, has become a strong base because it learns user-item representations well by combining input from nodes that are close to each other in the graph. This method gets rid of feature transformations and non-linear activation functions, which makes it a lot better than GCN-based methods that have been investigated before.

Hassanzadeh et al. put out a personalized knowledge-aware recommendation model called LightGCN (LGKAT) [9]. This model learns embeddings from both user-item interactions and knowledge graphs at the same time using an attention sub-network. This model encodes rich semantic information in a way that is unique to each user and gets better Recall and F1 scores on a number of datasets.

Fei Wang et al. have suggested the KLGCN model [10], which combines features from the interaction graph and the item-side knowledge graph separately. The model shows that dual-graph learning can greatly improve recommendation performance by improving embeddings for both users and entities.

In addition to these methods, BIKAGCN [11] adds a dual-layer architecture. The LightGCN representation learning process is followed by a knowledge-aware GCN layer that captures deeper entity relationships inside the knowledge graph. This means that adding different types of graph convolutional layers on top of each other can improve the ability to extract features. Other studies focus on multi-view knowledge graph convolutional networks that learn item representations from different relational views in the knowledge graph, which makes it easier to simulate long-term user preferences.

In addition to static graph embedding models, researchers have extended LightGCN research toward contextual dimensions. Temporal variants integrate dynamic interaction models to encapsulate periodic or time-dependent user preferences. Authors in [12] utilized LightGCN inside a seasonality-aware framework, explicitly modeling temporal trends to mitigate the cold-start issue and enhance suggestion quality. Authors in [13] also proposed a causal graph convolution mechanism that preserves the chronological order of interactions for time-sensitive recommendation tasks, demonstrating that temporal context can be critical in recommender systems.

Integrating auxiliary information with the LightGCN model represents a approach to enhancing the effectiveness of recommender systems. Consequently, we aim to leverage this potential to conduct deeper research into recommender systems by employing LightGCN to utilize the contextual information available within datasets.

3. Methodology

The Contextual-enhanced LightGCN model is constructed based on item content information. Item content features serve as the model input, whereas nodes representing users are initialized as random embeddings, following the original LightGCN approach [7]. This allows the model's input embeddings to be more informative compared to a completely random initialization. The detailed architecture of the model is presented in Figure 1.

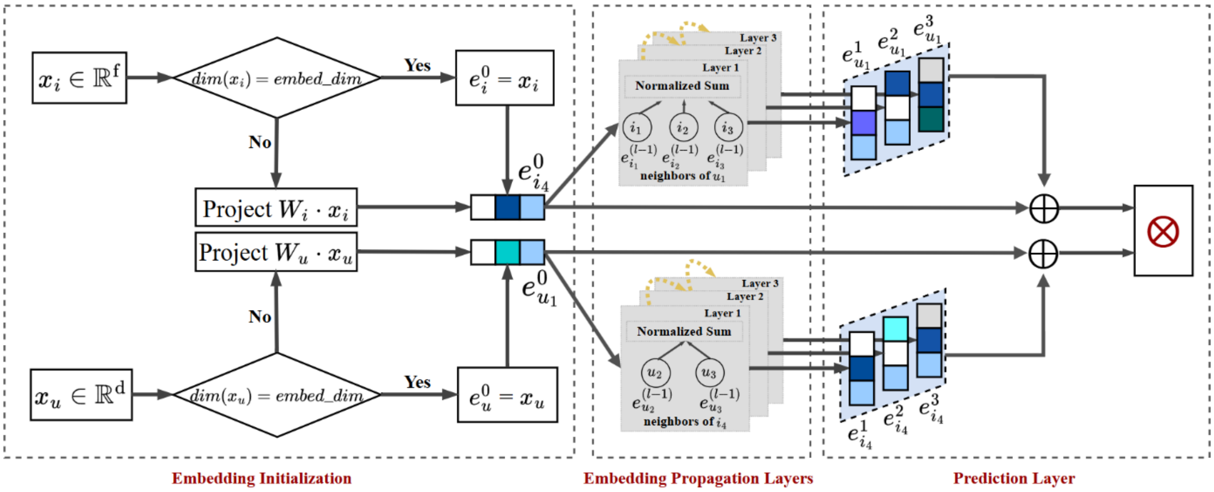


Figure 1. *An illustration of the complete pipeline from feature extraction through graph propagation to final prediction.*

3.1. Contextual feature extraction

This step transforms the input data, including user information x_u and movie information x_i into feature vectors. This process serves to encode contextual information, establishing the foundation for subsequent processing layers.

In recommender systems, contextual information includes auxiliary data that provides semantic meaning beyond user-item interactions. According to our definition, contextual information includes:

- Content Features: Intrinsic attributes of the items (e.g., movie genre, title).
- Auxiliary Information: Demographic attributes of the users (e.g., gender, age, occupation)

Integrating such information addresses the cold-start problem and enhances the semantic richness of the representations. This process consists of the following sub-steps:

3.1.1. Item content feature extraction

Genre encoding

For each item i, we extract genre information using Multi-Label Binarization. Given the input set of genres $\mathcal{G} = g_1, g_2, \dots, g_m$, the feature vector for item i is defined as $\boldsymbol{g_i} \in 0,1^{|\mathcal{G}|}$, where each dimension represents the presence (1) or absence (0) of a specific movie genre.

Title feature extraction

We employ Term Frequency-Inverse Document Frequency (TF-IDF) to extract semantic features from the titles of the input movie data as Eq.(1):

$$\text{TF-IDF}(t, d, D) = \text{TF}(t, d) \times \text{IDF}(t, D) \tag{1}$$

Where $\text{TF}(t, d)$ is the frequency of term t in title d, and $\text{IDF}(t, D) = \log \frac{|D|}{|d \in D : t \in d|}$ measures the importance of term t across the entire corpus D.

The resulting title feature vector $t_i \in R^{\mathbb{100}}$ captures the semantic content of the titles for each movie in the input dataset.

Combined item feature vector

This is the final task following the extraction of auxiliary movie information features. The final feature representation is synthesized by concatenating the genre and title features obtained in the previous two steps as Eq.(2):

$$x_i = [g_i | t_i] \in R^f \tag{2}$$

Where $f = |\mathcal{G}| + 100$ represents the total number of feature dimensions, and symbol | denotes the concatenation operator.

3.1.2. User demographic feature extraction

Categorical encoding

User demographic attributes are encoded using Label Encoding followed by Embedding Layers, as shown in Table 1.

Table 1. *Feature specification.*

Attribute	Encoding Method	Embedding Dimension
Gender	Label Encoding → Embedding	d_g
Age	Label Encoding → Embedding	d_a
Occupation	Label Encoding → Embedding	d_o

For each user u with gender g_u, age a_u and occupation o_u, we define $x_u = [\, e_g \mid e_a \mid e_o \,] \in R^d$, where $d = d_g + d_a + d_o$ is the total dimensionality of the user embedding vector.

3.2. Feature-enriched embedding initialization

After transforming the features of user and movie information, the next step involves unifying their representation dimensions using projection matrices W_u and W_i when the dimensions are unequal. This process ensures that all context feature embeddings are structurally compatible prior to graph propagation.

Traditional collaborative filtering methods initialize user and item embedding vectors randomly from a normal distribution $\mathcal{N}(0, \sigma^2)$. This approach (1) ignores existing semantic information; (2) requires longer training time to reach convergence; (3) may lead to suboptimal user-item representation learning for sparse datasets.

Our approach leverages contextual features to warm-start the embedding representation learning process.

To address the dimensional discrepancy between the feature vectors and the target embedding dimension d_{embed}, we propose a conditional projection or adjustment mechanism for the input vectors of both user and item embeddings.

For each item i, if the input feature vector x_i has a dimensionality equal to d_{embed}, this vector is directly used as the initial embedding $e_i^{(0)}$ (Eq.(3)). Conversely, if the dimensionality of x_i differs from d_{embed}, a linear projection via a learnable matrix W_i is applied to map x_i into the embedding space with the correct d_{embed} dimensionality.

Similarly, for each user u, the initial embedding $e_u^{(0)}$ (Eq.(4)) follows the same mechanism: the feature vector x_u is used directly when the dimensions are already aligned, or a linear projection via matrix W_u is applied when dimensional adjustment to d_{embed} is required.

$$e_i^{(0)} = \begin{cases} x_i & \text{if } dim(x_i) = d_{embed} \\ W_i \cdot x_i & \text{if } dim(x_i) \neq d_{embed} \end{cases} \tag{3}$$

$$e_u^{(0)} = \begin{cases} x_u & \text{if } dim(x_u) = d_{embed} \\ W_u \cdot x_u & \text{if } dim(x_u) \neq d_{embed} \end{cases} \tag{4}$$

Where $W_i, W_u \in R^{d_{embed} \times f}$ represent the projection matrices that are learnable during the training process.

This approach guarantees that the input characteristics possess consistent dimensions prior to the training phase and that both user and item representations exist within the same embedding space.

Feature-enriched initialization presents several notable advantages over random initialization. Specifically, while random initialization fails to preserve semantic information and results in poor performance in cold-start scenarios, feature-enriched initialization maintains the original semantic information, thereby improving effectiveness in data-sparse situations. Due to these semantically rich initial embeddings, the model's training process achieves faster convergence compared to random initialization, which starts from unstructured representations. This provides a favorable foundation for learning and optimizing the model in subsequent training stages.

3.3. Lightweight graph propagation

The objective of this step is to exploit the relationships between users and items through a multi-level neighborhood information aggregation mechanism. Utilizing a “Lightweight” structure minimizes computational overhead while still allowing embeddings to capture collaborative signals from indirect connections within the graph.

This step consists of the following sub-steps:

3.3.1. User-Item bipartite graph

First, a user-item bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is constructed, where $\mathcal{V} = \mathcal{U} \cup \mathcal{I}$ is the set containing user and item nodes, and $\mathcal{E} = \{(u,i): u \in \mathcal{U}, i \in \mathcal{I}, r_{ui} \text{ exists}\}$ is the set of edges connecting users and items in the graph.

Next, a symmetric normalized matrix is constructed as Eq.(5).

$$\tilde{A} = D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{5}$$

In this formulation, $A \in R^{(|\mathbb{U}|+|\mathbb{I}|)\times(|\mathbb{U}|+|\mathbb{I}|)}$ is the adjacency matrix of the bipartite graph, where $|\mathbb{U}|$ and $|\mathbb{I}|$ represent the number of users and items, respectively. D is the diagonal degree matrix with $D_{ii} = \sum_j A_{ij}$.

This operation ensures (1) Connectivity scale invariance, it reduces dependency on connection scale, maintaining the graph’s scaling properties; (2) Gradient stability, repeated information passing through an adjacency matrix can lead to exploding or vanishing gradients; normalization keeps the signal and gradient magnitudes within a stable range; (3) Symmetry in information propagation, the use of $D^{-1/2}$

ensures that the influence between adjacent nodes is balanced, resulting in a symmetric and equitable message passing mechanism.

3.3.2. Light graph convolution

Unlike the traditional GCN model, LightGCN removes feature transformations and non-linear activation functions, performing only the aggregation of neighboring nodes as Eq.(6) and Eq.(7):

$$e_u^{(k)} = \sum_{i \in \mathcal{N}_u} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} e_i^{(k-1)} \tag{6}$$

$$e_i^{(k)} = \sum_{u \in \mathcal{N}_i} \frac{1}{\sqrt{|\mathcal{N}_u|}\sqrt{|\mathcal{N}_i|}} e_u^{(k-1)} \tag{7}$$

Where $\mathcal{N}_u$ denotes the set of items interacted with by user u, $\mathcal{N}_i$ represents the set of users who have interacted with item i and k indicates the layer index. Another representation of the LightGCN aggregation process is expressed as Eq.(8):

$$E^{(k)} = \tilde{A}E^{(k-1)} \tag{8}$$

3.3.3. Multi-layer embedding fusion

After the embedding vectors have been propagated through the layers, these representation vectors are aggregated using the average of all layers as Eq.(9) and Eq.(10).

$$e_u = \frac{1}{K+1}\sum_{k=0}^{K} e_u^{(k)} \tag{9}$$

$$e_i = \frac{1}{K+1}\sum_{k=0}^{K} e_i^{(k)} \tag{10}$$

Where K denotes the total number of propagation layers.

This layer integrates three primary tasks: (1) capturing the aggregated neighborhood information; (2) balancing the local and global graph structures; and (3) preventing over-smoothing by incorporating the initial representation embedding vectors into the final aggregation.

3.4. Hybrid recommendation

This stage synthesizes the feature representations obtained from the initialization and graph propagation processes to calculate prediction scores. The procedure consists of the following sub-steps:

First, the predicted preference score for user u toward item i is calculated via the inner product as Eq.(11):

$$\hat{y}_{\mathrm{ui}} = e_u^{\top} e_i \tag{11}$$

Next, the model is evaluated using the Bayesian Personalized Ranking (BPR) loss, which is specifically designed for implicit feedback interaction data as Eq.(12):

$$\mathcal{L}_{\mathcal{BPR}} = \sum_{(u,i,j) \in \mathcal{D}} -\ln \sigma\left(\hat{y}_{ui} - \hat{y}_{uj}\right) \tag{12}$$

In this formulation, (u, i, j) represents a triplet consisting of user u, positive item i, positive item j and $\mathcal{D}$ denotes the training dataset of these triplets.

To prevent overfitting, L2 regularization is applied to the initial embedding vectors as Eq.(13):

$$\mathcal{L}_{reg} = \frac{1}{2|\mathcal{B}|}\sum (u,i,j) \in \mathcal{B} \left(\left|e_u^{(0)}\right|^2 + \left|e_i^{(0)}\right|^2 + \left|e_j^{(0)}\right|^2\right) \tag{13}$$

Finally, the total loss function is defined as Eq.(14):

$$\mathcal{L} = \mathcal{L}_{\mathcal{BPR}} + \lambda\mathcal{L}_{reg} \tag{14}$$

Where λ serves as the regularization coefficient.

4. Experimental Results

4.1. Experimental data

In this paper, we utilize two datasets, MovieLens 100K [14] and MovieLens 1M [14] , for experimentation and evaluation. MovieLens 100K consists of 100,000 ratings from 943 users across 1,682 movies. MovieLens 1M contains 1,000,209 ratings from 6,040 users on 3,706 movies, using a rating scale from 1 to 5 stars. User data includes UserID, age, gender, occupation, and zip code. Movie data includes MovieID, title, and genre (19 genres for MovieLens 100K and 18 genres for MovieLens 1M). Interaction data records user movie ratings along with the corresponding scores and timestamps.

Movie features are represented by concatenating the one-hot encoding of genres with a 100-dimensional TF-IDF vector derived from the titles, resulting in a 119-dimensional feature vector for MovieLens 100K and a 118-dimensional feature vector for MovieLens 1M.

The data density is quite sparse, reaching only 6.3% for MovieLens 100K and 4.47% for MovieLens 1M. The data is divided utilizing an 80-20 random allocation for each user. The 80-20 ratio was chosen due to its prevalent use in recommendation system research, providing substantial training data while preserving a suitable sample size for dependable evaluation.

4.2. Experimental Methodology

4.2.1. Experimental evaluation metrics

In this paper, the recommendation quality is evaluated using standard top-*K* metrics, where *K* is set to 20 to ensure a fair comparison with the LightGCN baseline. Model performance is assessed through Precision@K, Recall@K, and NDCG@K. To monitor the model across training epochs, NDCG@K is utilized as the primary comparison metric for checkpointing the optimal parameter set. To mitigate the impact of randomness and ensure statistical reliability, all experiments were repeated three times with different random seeds. The final reported metrics are the average values of these independent runs.

4.2.2. Parameter settings

To ensure a fair comparison and isolate the contribution of our contextual initialization strategy, we strictly adhered to the hyperparameters recommended in the original LightGCN paper [7], setting the dimensionality of the vector space to 64 and utilizing 3 graph propagation layers. The Adam optimizer is employed with a learning rate of 0.001, a batch size of 2048, and an L2 regularization coefficient (λ) of $1e-4$.

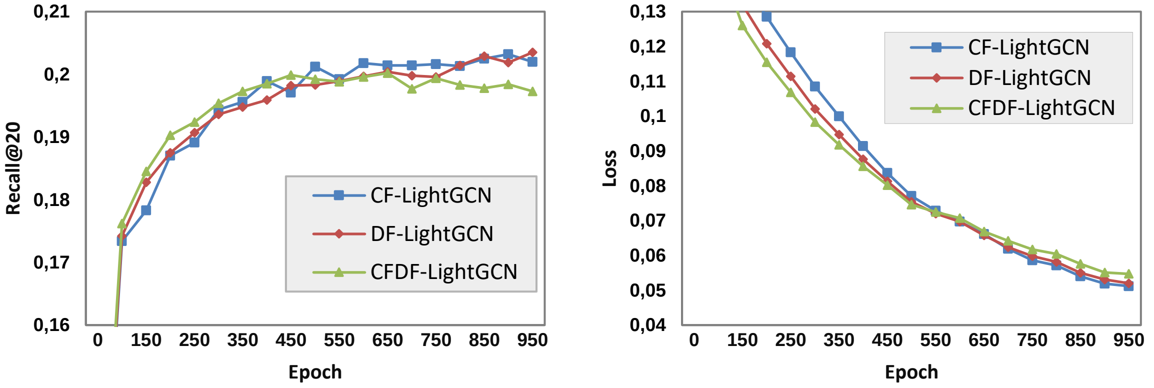
4.2.3. Ablation methods

To evaluate the contribution of each type of contextual feature, we conducted a systematic ablation study as detailed in Table 2.

Table 2. *Ablation methods design.*

Method	Item Features	User Demographics
CF-LightGCN	✓	✗
DF-LightGCN	✗	✓
CFDF-LightGCN	✓	✓

4.3. Experimental results and discussion



(a) (b)

Figure 2. *Training curves of LightGCN's three variants on MovieLens 100K: (a) Recall@20; (b) Loss.*

Figure 2.a demonstrates that CF-LightGCN attains superior metrics compared to the other two variations. The richness and diversity of movie elements can elucidate this; specifically, data on genres and titles provide robust semantic signals for differentiating films, allowing the model to acquire item representations with enhanced generalization skills. CFDF-LightGCN reaches approximately 0.2, failing to outperform CF-LightGCN despite incorporating both data sources, which suggests potential signal redundancy or conflict.

As shown in Figure 2.b, all three variants decrease steadily from 0.13 to approximately 0.05, demonstrating effective optimization. The loss curves for CF-LightGCN and CFDF-LightGCN almost overlap, indicating that the addition of demographic features does not significantly alter the optimization process. There are no signs of overfitting under the current regularization configuration.

Table 3. *Performance comparison of ablation variants.*

Dataset	Method	Precision	Recall	NDCG
MovieLens 100K	LightGCN	0.1475	0.2008	0.2145
	CF-LightGCN	**0.1492**	**0.2025**	**0.2163**
	DF-LightGCN	0.1475	0.2002	0.2138
	CFDF-LightGCN	0.1485	0.2015	0.2154
MovieLens 1M	LightGCN	0.2852	0.2459	0.3725
	CF-LightGCN	**0.2883**	**0.2496**	**0.3769**
	DF-LightGCN	0.2840	0.2463	0.3714
	CFDF-LightGCN	0.2857	0.2475	0.3730

The experimental results of the ablation study on both MovieLens 100K and MovieLens 1M are presented in Table 3.

For MovieLens 100K, the CF-LightGCN model obtains the best performance, improving Recall by 0.85% and Precision by 1.15% compared to LightGCN. Nevertheless, DF-LightGCN performs worse than LightGCN, this indicates that demographic features act as noise rather than informative signals in the current architecture. We attribute this to the "semantic gap" between static demographic data and dynamic user preferences. Furthermore, CFDF-LightGCN shows lower performance than CF-LightGCN. This suggests that our simple concatenation-based fusion strategy treats all features equally, allowing noisy demographic signals to dilute the high-quality content embeddings.

For MovieLens 1M, CF-LightGCN improves Recall by 1.5% over the baseline. Although DF-LightGCN demonstrates improved results relative to its performance on the smaller dataset, it continues to fall short of the baseline. CFDF-LightGCN surpasses DF-LightGCN but remains inferior than CF-LightGCN.

These quantitative results indice that the performance gain of CF-LightGCN increases with dataset size from 0.85% to 1.5%, suggesting that the feature-enriched approach requires larger volumes of data to reach its full potential.

Table 4. *Comparison performance.*

Dataset	Method	Precision	Recall	NDCG
MovieLens 100K	MF	0.1409	0.1924	0.2066
	NGCF	0.1365	0.1820	0.1970
	LightGCN	0.1475	0.2008	0.2145
	CF-LightGCN	**0.1492**	**0.2025**	**0.2163**
MovieLens 1M	MF	0.2646	0.2227	0.3419
	NGCF	0.2765	0.2316	0.3576
	LightGCN	0.2852	0.2459	0.3725
	CF-LightGCN	**0.2883**	**0.2496**	**0.3769**

Looking at Table 4, it can be seen that the paper compares experimental results with the MF, NGCF, and LightGCN models. With an NDCG of **0.2163** on MovieLens 100K and **0.3769** on MovieLens 1M, the results demonstrate that our proposed model outperforms the other three methods. These findings highlight the effectiveness of graph-based approaches in exploiting user-item relational structures.

The experimental results confirm two significant findings: (1) the integration of content features provides a distinct performance improvement over both the original LightGCN and more complex GNN architectures such as NGCF; (2) the streamlined structure of LightGCN, when combined with feature enrichment, outperforms sophisticated architectures like NGCF that rely on multiple layers of transformation and nonlinearity.

5. Conclusion

This research proposes Contextual-enhanced LightGCN to enhance the performance of recommendation systems by replacing random embedding initialization with an information-rich initialization mechanism based on input data.

Through experiments on two datasets, this research has demonstrated that integrating movie content features helps the model overcome the limitations of data sparsity and the cold-start problem. Experimental results on the MovieLens 100K and 1M datasets show that the variant utilizing content features performs most effectively. This approach enables the model to converge faster and achieve superior accuracy compared to the original LightGCN model as well as more complex architectures like NGCF.

In addition to the positive findings, the paper highlights several drawbacks regarding feature integration and selection. Compared to utilizing only movie content features, integrating user demographic data or combining both feature types simultaneously does not yield high efficiency and can even lead to performance degradation. The current techniques of vector concatenation and linear projection are not yet advanced enough to effectively handle the noise present in demographic data.

Future research directions will focus on two primary aspects. First, more advanced feature integration mechanisms, such as attention-based neural networks [9], will be investigated to achieve more effective weighting and noise filtering when combining multi-source data, replacing simple linear projections. Second, the model will be extended to handle dynamic graphs [12], [13] or incorporate temporal and seasonal factors. This expansion aims to capture real-time transitions in user preferences, thereby further enhancing the practical utility and predictive accuracy of the system.

Acknowledgements

This research was completed with the active support and facilitation provided by the Editorial Board of the Journal of Technical Education Science at Ho Chi Minh City University of Technology and Engineering. We would like to express our sincere gratitude for your valuable contributions, which were instrumental in the execution and publication of this article.

Conflict of Interest

The authors declare no conflict of interest.

REFERENCES

[1] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 974-983.

[2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering,* vol. 17, no. 6, pp. 734-749, 2005.

[3] K. Lang, "NewsWeeder: Learning to Filter Netnews," in *Machine Learning Proceedings 1995,* A. Prieditis and S. Russell, Eds. San Francisco (CA): Morgan Kaufmann, 1995, pp. 331-339.

[4] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," *Advances in neural information processing systems,* vol. 20, 2007.

[5] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv:1205.2618,* 2012.

[6] R. Burke, "Hybrid recommender systems: Survey and experiments," *User modeling and user-adapted interaction,* vol. 12, no. 4, pp. 331-370, 2002.

[7] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, "Lightgcn: Simplifying and powering graph convolution network for recommendation," in *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, 2020, pp. 639-648.

[8] S. Raza *et al.*, "A comprehensive review of recommender systems: Transitioning from theory to practice," *Computer Science Review,* vol. 59, p. 100849, 2026.

[9] R. Hassanzadeh, V. Majidnezhad, and B. Arasteh, "A novel recommender system using light graph convolutional network and personalized knowledge-aware attention sub-network," *Scientific Reports,* vol. 15, 05/05 2025.

[10] F. Wang, Y. Li, Y. Zhang, and D. Wei, "KLGCN: Knowledge graph-aware Light Graph Convolutional Network for recommender systems," *Expert Systems with Applications,* vol. 195, p. 116513, 2022/06/01/ 2022.

[11] L. Guoshu, Y. Li, and B. Sichang, "BIKAGCN: Knowledge-Aware Recommendations Under Bi-layer Graph Convolutional Networks [J]," *Neural Processing Letters,* vol. 56, p. 20, 2024.

[12] A. Drif, M. Tabti, M. Tamhachet, and H. Cherifi, "LightGCN with Season Filtering for Recommender System," 2025, pp. 158-169.

[13] A. Ghiye, B. Barreau, L. Carlier, and M. Vazirgiannis, "Rolling Forward: Enhancing LightGCN with Causal Graph Convolution for Credit Bond Recommendation," in *Proceedings of the 5th ACM International Conference on AI in Finance*, 2024, pp. 231-238.

[14] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm transactions on interactive intelligent systems (tiis),* vol. 5, no. 4, pp. 1-19, 2015.

Dinh-Quoc-Hoa Pham was born in Vietnam in 2001. In 2023, he graduated with a Engineer’s degree in Faculty of Information Technology from Ho Chi Minh City University of Technology and Education (currently Ho Chi Minh City University of Technology and Engineering). He is currently pursuing a Master’s degree in Computer Science at the same university. His research interest is recommendation systems**.**

Email: hoadaknong101@gmail.com, 2531307@student.hcmute.edu.vn. ORCID: https://orcid.org/0009-0000-9856-3239

Huyen-Trang Phan received the M.S. degree in computer science from the University of Science and Technology - The University of Da Nang, Vietnam, in 2015, Ph.D. degree and Postdoctoral in Computer Science at the Department of Computer Engineering from Yeungnam University, South Korea in 2020 and 2021. She worked as a research professor at the Department of Computer Engineering, Yeungnam University, South Korea, from 2021 to 2024. She is currently a lecturer at the Faculty of Information Technology, Ho Chi Minh City University of Technology and Engineering, Vietnam. She is the author of 10 journal papers and 15 conference papers. Her research interestsinclude sentiment analysis, fake news detection, text summarization, and decision support systems.

Email: trangpth@hcmute.edu.vn. ORCID: https://orcid.org/0000-0002-7466-9562