iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: III Month of publication: March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78453

A Comprehensive Review of Mobile App Recommendation Systems Using Machine Learning and Deep Learning Techniques

Akhila V K[1], Anjitha V K[2], Hanna R P[3], Devika S[4], MS. Ramya K V[5]
Department of Computer Science and Engineering (CSE), Sree Narayana Guru College of Engineering and Technology Korom, Payyanur, Kannur

Abstract: The exponential growth of Mobileapplications has created a challenge for users in discovering relevant and high-quality apps that meet their preferences. Mobile app recommendation systems address this issue by providing personalized suggestions based on user behavior, preferences, and interaction history. This review paper presents a comprehensive study of mobile app recommendation systems with an emphasis on machine learning and deep learning techniques. Various approaches including content-based filtering, collaborative filtering, hybrid recommendation models, and deep learning-based methods are analyzed.

The role of crowdsourced data and sequential user behavior modeling using deep learning architectures such as GRU is discussed. Furthermore, key challenges, limitations, and future research directions in mobile app recommendation systems are highlighted.

I. INTRODUCTION

Mobile applications have become an integral part of everyday life, supporting domains such as education, entertainment, healthcare, and productivity. With millions of applications available on platforms like Google Play Store and Apple App Store, users face difficulty in identifying apps that align with their interests and requirements. This problem has led to the increasing importance of intelligent recommendation systems. Mobile app recommendation systems aim to analyze user behavior, app features, and historical interactions to generate personalized recommendations. Traditional recommendation methods often suffer from issues such as data sparsity and cold-start problems. To overcome these limitations, machine learning and deep learning approaches have been widely adopted. This paper reviews existing research and techniques used in mobile app recommendation systems and explores recent trends and challenges.

II. RELATED WORK

Several studies have explored recommendation systems for mobile applications using different computational techniques. Early research primarily focused on collaborative filtering and content-based methods. With the advancement of machine learning, matrix factorization and clustering-based approaches gained popularity. Recent studies emphasize deep learning models that capture complex user behavior patterns and sequential interactions. Recurrent Neural Networks (RNNs), LSTM, and GRU-based models have shown promising results in modeling temporal user preferences. Hybrid approaches combining traditional and deep learning techniques are increasingly adopted to improve recommendation accuracy and scalability.

III. LITERATURE REVIEW

A. “Recommender Systems Survey,” Knowledge-Based Systems

The study presented a comprehensive survey of recommender systems, covering collaborative filtering, content-based filtering, and hybrid approaches. It analyzed traditional machine learning techniques and highlighted the growing importance of deep learning models in recommendation tasks. The authors examined key methodologies and compared their effectiveness in handling different types of data and user interactions.

Furthermore, the study discussed major challenges such as data sparsity, cold- start problems, scalability, and the limitations of existing evaluation metrics. It emphasized how these issues affect the performance and reliability of recommendation systems, especially in dynamic and large-scale environments.

In addition, the survey explored recent advancements in deep learning techniques, including neural networks and sequence- based models, which enable better understanding of user behavior and temporal patterns. The authors also highlighted the importance of incorporating contextual information, user feedback, and sentiment analysis to enhance recommendation accuracy.

B. Deep Learning–Based Mobile App Recommendation

A deep learning-based framework for mobile app recommendation using user behavioral data. The system employed neural networks to model complex relationships between users and applications. Experimental results showed that deep learning models significantly outperform traditional collaborative filtering methods.

The study demonstrated the effectiveness of learning latent representations from user interactions, making it suitable for dynamic mobile app ecosystems.

C. Neural Collaborative Filtering for Recommendation Systems

Neural Collaborative Filtering (NCF), which replaces the linear inner product of matrix factorization with a neural network architecture.

The model effectively captures non- linear user–item interactions and improves recommendation accuracy. This approach has been widely adopted in recommendation systems, including mobile app recommendation, due to its flexibility and improved performance over traditional machine learning techniques.

D. Context-Aware Mobile App Recommendation Using Deep Learning

Focused on incorporating contextual information such as time, location, and usage patterns into mobile app recommendation systems.

The proposed deep learning model dynamically adapts recommendations based on changing user contexts. The results showed enhanced personalization and higher user satisfaction compared to context-free recommendation approaches. This study highlights the importance of context-aware learning in mobile environments.

E. Sequential Mobile App Recommendation Using GRU Networks

Proposed a GRU-based sequential recommendation model to capture temporal user behavior in mobile app usage. The model effectively learns user interest evolution over time while maintaining lower computational complexity compared to LSTM. Experimental evaluation demonstrated superior performance in terms of recommendation accuracy and efficiency. This work confirms the suitability of GRU models for real-time mobile app recommendation systems.

IV. CONCLUSION FROM LITERATURE REVIEW

Based on the analysis of the five selected studies, it can be concluded that mobile app recommendation systems have evolved significantly with the adoption of machine learning and deep learning techniques. Early approaches relying on collaborative filtering and content-based methods face limitations such as cold-start issues and data sparsity.

To address these challenges, researchers have increasingly focused on hybrid and deep learning-based solutions. The reviewed literature demonstrates that deep learning models are capable of capturing complex and non- linear user–item relationships, resulting in improved recommendation accuracy. Context-aware models further enhance personalization by considering dynamic factors such as time and location.

Moreover, GRU-based sequential recommendation models have proven effective in modeling temporal user behavior while maintaining computational efficiency. Despite these advancements, challenges related to privacy, scalability, and explainability remain open research areas.

Future work should focus on developing privacy-preserving, explainable, and context-aware deep learning models to further improve the effectiveness of mobile app recommendation systems.

V. PROPOSED SYSTEM ARCHITECTURE

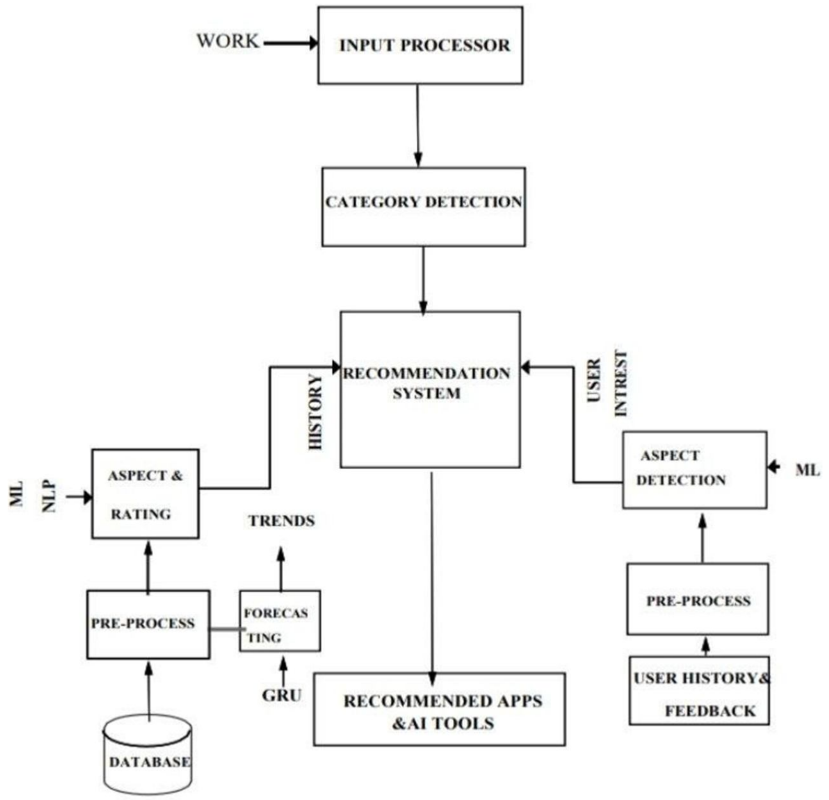


Fig.1 Proposed System Architecture of Mobile App Recommendation System

The overall architecture of the proposed system consists of multiple modules that work together to provide personalized recommendations. The system begins with an input processor that collects and pre-processes user data, followed by a category detection module for classification. The core recommendation system generates suggestions based on user interests, historical data, and trends.Machine Learning and NLP techniques are used to analyze user behavior, feedback, and ratings, while a GRU-based forecasting module identifies emerging trends. All data is stored in a centralized database, and the backend server manages communication between modules. The system finally outputs recommended applications and AI tools.The final output of the system is a set of personalized recommendations, including relevant applications and AI tools. Thus, the proposed architecture ensures scalability, adaptability, and improved user experience through continuous learning and data-driven decision-making.

The proposed mobile app recommendation system architecture consists of the following components:

1) Data Collection Module: Collects user interaction data such as app downloads, ratings, reviews, and usage logs.
2) Preprocessing Module: Cleans and filters data, removes noise, and handles missing values.
3) Feature Extraction Module: Extracts user features, app metadata, and behavioral patterns.
4) Recommendation Engine: Uses machine learning and deep learning models (collaborative filtering, hybrid models, GRU-based networks).
5) Recommendation Output Module: Generates personalized app recommendations for users.

This architecture ensures scalability, adaptability, and improved recommendation accuracy.

VI. WORKING METHODOLOGY

The proposed mobile app recommendation system integrates machine learning, deep learning, and hybrid techniques to provide accurate and personalized recommendations. The working methodology consists of the following sequential steps:

A. User Interaction Data Collection

The first step involves collecting raw data from mobile app platforms such as Google Play Store or Apple App Store. The data includes:

- App usage patterns: frequency, duration, and session times.
- User interactions: app downloads, ratings, likes/ dislikes, and reviews.
- Metadata: app categories, descriptions, keywords, and version history.

This step ensures a rich dataset capturing both explicit feedback (ratings, reviews) and implicit feedback (usage patterns), which is crucial for building accurate recommendation models. Data is typically collected using APIs, web scraping, or integrated logging mechanisms within the mobile apps.

B. Data Preprocessing

Raw data often contains noise, missing values, and inconsistencies. Preprocessing is therefore essential and includes:

- Data cleaning: Removing duplicate or irrelevant records, handling inconsistencies in app metadata.
- Handling missing values: Using techniques such as mean/mode imputation or predictive models to fill gaps.
- Normalization: Scaling numerical features (e.g., usage time, ratings) to a uniform range to improve model convergence.
- Noise filtering: Detecting and removing outlier interactions that could bias recommendations.

Preprocessing ensures high-quality input for machine learning and deep learning models, which improves predictive performance

C. Feature Extraction

After preprocessing, feature vectors are created to represent both users and applications. Key aspects include:

- User features: demographics, historical app usage, preferences, and feedback patterns.
- App features: categories, textual description embeddings (e.g., using Word2Vec, TF-IDF, or BERT embeddings), keywords, and popularity metrics.
- Behavioral features: temporal sequences of interactions, session patterns, and recency of app usage.

These features are critical for models to capture complex user–app relationships and serve as input for machine learning and deep learning models.

D. Machine Learning-Based Initial Recommendation

In this step, traditional machine learning models are applied to generate initial recommendations. Common models include:

- K-Nearest Neighbors (KNN): Identifies similar users or apps based on feature similarity.
- Decision Trees and Random Forests: Capture non-linear relationships between users and apps.
- Matrix Factorization: Decomposes the user–app interaction matrix to uncover latent factors.

The goal of this stage is to provide a baseline recommendation set that captures broad patterns in user preferences.

E. Deep Learning Models for Sequential Behavior Analysis

To account for the temporal dynamics of user behavior, deep learning models, particularly GRU (Gated Recurrent Unit) networks, are applied:

- Sequential modeling: GRUs process user interaction sequences to identify patterns in app usage over time.
- Temporal dependencies: The model captures evolving user preferences and predicts the next likely app of interest.
- Low computational cost: Compared to LSTMs, GRUs achieve similar accuracy with fewer parameters, enabling real-time recommendation.

This step refines recommendations by understanding not just static preferences, but also the evolution of user behavior.

F. Hybrid Recommendation Generation

To improve accuracy and robustness, recommendations from multiple models are combined in a hybrid approach:

- Weighted combination: Assigns weights to machine learning and deep learning model outputs to generate final scores.
- Feature augmentation: Integrates content-based and collaborative filtering features into deep learning models.
- Switching method: Dynamically selects the best model based on the availability of user data (e.g., cold-start users vs. active users).

This hybrid approach leverages the strengths of each model while minimizing their individual limitations.

G. Final Personalized App Recommendation Delivery

The final step involves presenting the personalized app recommendations to the user:

- Recommendations are ranked based on predicted user preference scores.
- Contextual factors such as current time, location, or device type can further refine the list.
- Feedback loops are incorporated: new user interactions update the dataset, allowing the system to adapt over time and improve recommendation quality continuously.

This step ensures that users receive highly relevant, timely, and actionable app suggestions, enhancing user engagement and satisfaction.

VII. CONCLUSION

This review paper presented a comprehensive and systematic analysis of mobile app recommendation systems with a focus on machine learning and deep learning techniques. Traditional recommendation approaches, including content- based filtering and collaborative filtering, were examined to establish their foundational role in personalized app recommendations. While these methods remain effective in certain scenarios, their limitations in handling dynamic user preferences, data sparsity, and cold-start situations were clearly identified. To address these limitations, advanced hybrid recommendation models that integrate multiple techniques were discussed. In addition, the growing adoption of deep learning-based approaches was highlighted, particularly for their ability to learn complex, non- linear relationships between users and applications. The review emphasized the significance of sequential modeling techniques, especially Gated Recurrent Units (GRU), in capturing temporal user behavior and evolving interests with lower computational complexity compared to traditional recurrent models.Furthermore, the role of crowdsourced data such as user ratings, reviews, and feedback was analyzed as a key factor in enhancing recommendation accuracy and system adaptability. Crowdsourced information provides real- world insights into user preferences and application quality, enabling recommendation systems to continuously improve through feedback-driven learning.Despite notable advancements, several challenges remain unresolved. Cold- start problems for new users and applications, privacy and security concerns related to user data, and scalability issues in large-scale mobile app platforms continue to limit system performance. These challenges highlight the need for more robust and adaptive solutions.

REFERENCES

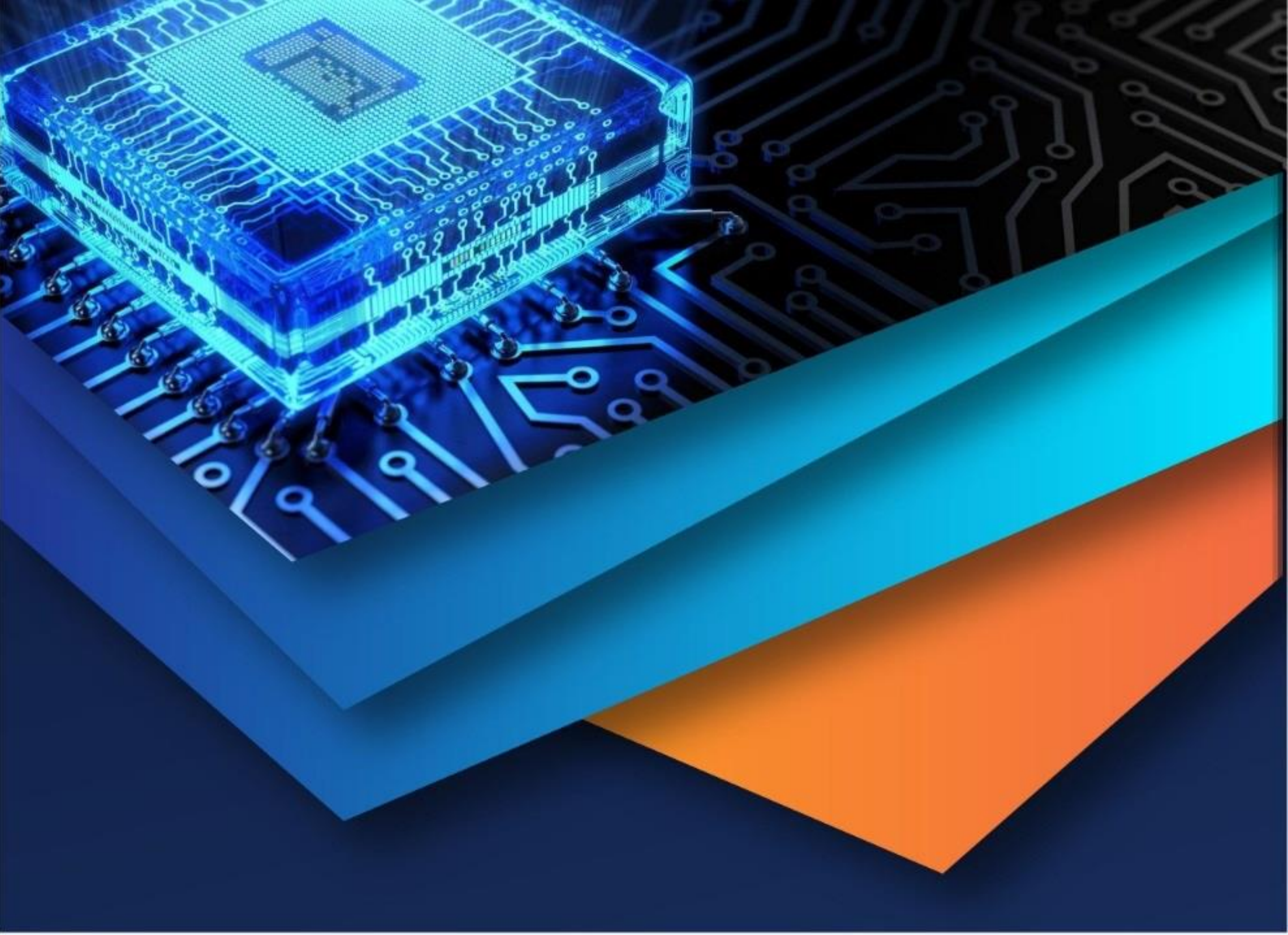
[1] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” Knowledge-Based Systems, vol. 173, pp. 109–128, 2019.

[2] X. Zhang, Y. Wang, and M. Li, “Deep learning–based mobile app recommendation systems,” IEEE Access, vol. 8, pp. 115–126, 2020

[3] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. Chua, “Neural collaborative filtering,” Proceedings of WWW, 2020.

[4] H. Wang, Y. Zhang, and Q. Liu, “Context-aware mobile app recommendation using deep learning,” IEEE Transactions on Mobile Computing, 2021.

[5] J. Li, K. Zhou, and Y. Chen, “Sequential mobile app recommendation using GRU networks,” IEEE Access, 2023.



10.22214/IJRASET

INDEX COPERNICUS
45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL
IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY