International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 10 Issue: 03 | March - 2026 SJIF Rating: 8.659 ISSN: 2582-3930

Optimizing Mobile App Recommendations Using Crowdsourced Educational Data

Kethavath Vijaya, department of Computer Science and Engineering, GNITC, 23-517, 23wj5a0517@gniindia.org
Lenkalapally Sravani, department of Computer Science and Engineering, GNITC, 22-5G7, 22wj1a05g7@gniindia.org
Katta Rahul, department of Computer Science and Engineering, GNITC, 23-516, 23wj5a0516@gniindia.org
MD Sirajul Huq, Assistant Professor, department of Computer Science and Engineering, GNITC

Abstract –With the rapid growth of mobile learning applications, students often face difficulty selecting the most relevant educational apps that suit their academic needs. This research proposes an intelligent recommendation system that optimizes mobile app suggestions using crowdsourced educational data. The system analyses application usage patterns and textual descriptions to recommend suitable apps for Undergraduate (UG), Postgraduate (PG), and Graduate students. Natural Language Processing (NLP) techniques such as text cleaning, tokenization, and feature extraction are applied to preprocess app metadata and user interaction data. A Random Forest Classifier is employed as the primary machine learning model due to its robustness, scalability, and strong performance in classification tasks. The model analyses behavioural patterns and content features to generate personalized recommendations. Experimental evaluation using metrics such as accuracy, precision, recall, and F1-score demonstrates that the proposed model produces reliable and interpretable results compared with traditional recommendation approaches. The system enhances personalization, reduces irrelevant suggestions, and improves user engagement in educational environments. This research contributes to the development of intelligent, AI-driven educational recommendation systems that support effective digital learning.

Key Words: Mobile App, Recommendation System, Crowdsourced Educational Data, Machine Learning, Random Forest Classifier, Natural Language Processing (NLP), Educational Technology, Personalized Learning, Data Mining.

1.INTRODUCTION

With the rapid advancement of mobile technology, educational applications have become an important tool for students to enhance their learning experience. Smartphones provide easy access to a wide range of educational apps that support activities such as online learning, exam preparation, and skill development. The large number of available applications makes it difficult for students to identify the most suitable apps for their academic needs.

Traditional app selection methods based on popularity or ratings often fail to provide personalized recommendations. To address this issue, intelligent recommendation systems can analyse user behaviour and application data to suggest relevant apps. This research focuses on developing a mobile app recommendation system using crowdsourced educational data.

The proposed system applies Natural Language Processing (NLP) techniques to preprocess app descriptions and extract useful features. A Random Forest Classifier is used to analyse the data and recommend suitable educational apps for Undergraduate (UG), Postgraduate (PG), and Graduate students. The system aims to improve personalization, enhance learning efficiency, and help students discover relevant educational applications more effectively.

2. LITERATURE REVIEW

Several studies have explored the use of recommendation systems to assist users in discovering relevant mobile applications and educational resources. With the rapid growth of digital learning platforms, researchers have focused on applying machine learning and data analysis techniques to improve the accuracy and personalization of app recommendations.

Maqbool et al. (2023) introduced MobileRec, a large-scale dataset designed for mobile application recommendation research. The dataset contains extensive user–app interaction records along with app metadata such as descriptions, categories, and ratings. Their study demonstrated that combining textual features from app descriptions with user interaction data significantly improves recommendation performance and helps generate more personalized suggestions.

Dave et al. (2023) proposed SAppKG, a knowledge graph–based recommendation framework that integrates app attributes, permissions, and user-side information to provide secure and accurate recommendations. The system uses additional contextual information to enhance prediction quality and ensures privacy-aware recommendation mechanisms. Their results showed that incorporating side information improves the precision and recall of recommendation systems compared to traditional collaborative filtering methods.

Badier et al. (2023) developed a recommendation model for an after-school e-learning mobile application. Their system analyses user interaction logs and educational content descriptions using Natural Language Processing (NLP) techniques. The study found that personalized recommendations significantly increase student engagement and learning efficiency compared to generic resource suggestions.

Barnes (2024) explored the application of the Random Forest Classifier in higher education for predicting academic outcomes and providing personalized recommendations. The research highlighted the advantages of Random Forest, including its ability to handle mixed data types, manage missing values, and provide interpretable feature importance. The study demonstrated that ensemble learning models can achieve higher prediction accuracy and reliability in educational data analysis.

Islam (2025) proposed a multi-model machine learning framework for academic course recommendation systems. The research combined behavioural data and textual features extracted using NLP techniques to generate personalized suggestions. The study concluded that hybrid machine learning models improve recommendation accuracy and help overcome challenges such as data sparsity and cold-start problems.

These studies highlight the growing importance of machine learning and NLP techniques in developing intelligent recommendation systems. However, there is still a need for scalable and interpretable models that effectively combine crowdsourced educational data with application metadata. The proposed system addresses this need by utilizing NLP-based feature extraction and a Random Forest Classifier to provide personalized educational app recommendations for university students.

3. RELATED WORK

Recommendation systems have become an essential tool for helping users discover relevant information in large digital environments. In recent years, many researchers have explored different techniques to improve the accuracy and personalization of mobile application recommendations, particularly in educational contexts.

One significant contribution in this field is the MobileRec dataset, introduced by Maqbool et al. (2023). This dataset contains millions of user–app interactions and rich metadata collected from the Google Play Store, including app descriptions, categories, ratings, and user activity. The dataset supports research in mobile app recommendation systems by enabling models to analyze user behavior and app characteristics for more accurate predictions.

Another notable work is SAppKG, a knowledge graph–based mobile app recommendation framework proposed by Dave et al. (2023). This model integrates app attributes, permissions, and interaction data to generate personalized recommendations while maintaining user privacy. The study demonstrated that using knowledge graphs and side information improves recommendation accuracy compared with traditional methods.

Educational recommender systems have also been widely studied. These systems use machine learning and information filtering techniques to guide students toward suitable learning resources such as courses, exercises, or digital learning tools. Studies show that such systems help improve learning outcomes and assist students in finding relevant educational content more efficiently.

Researchers have also explored conversational and sequential recommendation models. For example, the MobileConvRec dataset supports conversational mobile app recommendation systems that analyze user interactions and dialogues to understand user needs more effectively. These approaches combine sequential user behavior with natural language processing to generate more context-aware recommendations.

Despite these advancements, many existing recommendation systems rely on complex deep learning architectures or require very large datasets, making them difficult to interpret and computationally expensive. Therefore, machine learning algorithms such as Random Forest provide a practical alternative due to their robustness, interpretability, and ability to handle high-dimensional data efficiently. Based on these observations, the proposed system utilizes NLP techniques and a Random Forest Classifier to generate personalized educational app recommendations using crowdsourced data.

4. PROPOSED METHODOLOGY

Our Project focuses on developing an intelligent recommendation system that suggests suitable educational mobile applications to students using crowdsourced educational data. The system combines Natural Language Processing (NLP) techniques and machine learning algorithms to analyse application information and user interaction patterns. The goal of the system is to generate personalized app recommendations that match the academic level and learning preferences of students.

The first stage of the methodology is **data collection**. In this stage, data related to educational mobile applications and user interactions is gathered from various sources such as mobile app platforms, surveys, and usage logs. The collected dataset includes information such as application names, categories, textual descriptions, ratings, download counts, and user interaction data. Additionally, student information such as academic level (Undergraduate, Postgraduate, or Graduate) is included to help the model understand the type of applications suitable for each category of learners. The crowdsourced nature of the data ensures that the system reflects real user experiences and preferences.

After collecting the data, the next stage is data preprocessing. Raw datasets usually contain noise, irrelevant information, and inconsistent data formats that can negatively affect model performance. Therefore, preprocessing techniques are applied to clean and organize

the data. For textual information such as app descriptions and reviews, Natural Language Processing techniques are used. These include text cleaning, removal of special characters, tokenization, stop-word removal, and normalization. These steps help in transforming unstructured textual information into a format that can be processed by machine learning algorithms.

The system performs feature extraction and representation. Feature extraction is an important step because it converts the processed textual data into numerical values that can be used for model training. Techniques such as Term Frequency–Inverse Document Frequency (TF-IDF) or vectorization are used to represent the importance of words in application descriptions. In addition to textual features, other attributes such as application category, ratings, and usage frequency are also included as input features. These combined features provide a comprehensive representation of both application characteristics and user preferences.

The next stage is model selection and training. In the proposed system, the Random Forest Classifier is selected as the primary machine learning model. Random Forest is an ensemble learning algorithm that builds multiple decision trees during training and combines their predictions to produce a final result. Each tree in the forest analyzes different subsets of the dataset, which helps reduce overfitting and improves prediction accuracy. The model learns patterns from the training data and identifies relationships between application features and student categories. Hyperparameters such as the number of trees and tree depth can be adjusted to optimize the model's performance.

Once the model is trained, the system proceeds to the prediction and recommendation stage. When new user data is provided, the trained model analyzes the features extracted from the input data and predicts the most suitable educational applications for that particular student. The system then generates a list of recommended apps that align with the student's academic level and learning requirements. This personalized approach helps students quickly identify useful applications without spending excessive time searching through thousands of options.

The performance evaluation stage is conducted to measure the effectiveness of the recommendation system. Several evaluation metrics are used, including accuracy, precision, recall, and F1-score. These metrics help determine how well the model predicts suitable applications and how reliable the recommendations are. The results are also compared with other baseline models such as decision trees or logistic regression to verify the advantages of the Random Forest approach.

The proposed methodology integrates crowdsourced educational data, Natural Language Processing techniques, and machine learning algorithms to build a robust recommendation system. By combining behavioral data and textual information, the system can provide accurate and personalized mobile app recommendations for students across different academic levels. This approach enhances learning efficiency, improves user experience, and contributes to the development of intelligent educational technologies.

5. RESULTS AND DISCUSSION

Mobile app recommendation system was implemented and evaluated using a dataset consisting of educational mobile application information and user interaction data collected from crowdsourced sources. The dataset included various features such as application name, category, textual description, ratings, download statistics, and user engagement patterns. These features helped the system understand both the characteristics of the applications and the preferences of students from different academic levels. Before training the model, the dataset underwent a comprehensive preprocessing phase in which Natural Language Processing (NLP) techniques were applied to the textual data. This preprocessing included text cleaning, tokenization, removal of stop words, and vectorization to convert the unstructured textual data into structured numerical features that could be used by machine learning algorithms.

After preprocessing, the dataset was divided into two subsets: a training dataset and a testing dataset. The training dataset was used to train the machine learning model, while the testing dataset was used to evaluate the performance of the system on unseen data. The Random Forest Classifier was selected as the core algorithm for this recommendation system because of its strong ability to handle high-dimensional data and its robustness in classification tasks. Random Forest works by constructing multiple decision trees during the training process and combining their outputs to produce a final prediction. This ensemble learning approach helps improve prediction accuracy and reduces the risk of overfitting.

During the training phase, the Random Forest model learned the relationship between application features and the academic level of students, including Undergraduate (UG), Postgraduate (PG), and Graduate levels. Each decision tree in the forest analyzed different subsets of the data and features, allowing the model to capture diverse patterns in the dataset. Once the training process was completed, the model was tested using the testing dataset to determine its effectiveness in recommending suitable educational applications.

To evaluate the performance of the recommendation system, several standard evaluation metrics were used. These included accuracy, precision, recall, and F1-score. Accuracy measures the overall correctness of the model's predictions, while precision indicates how many of the recommended applications were actually relevant to the user. Recall measures the system's ability to identify all relevant applications, and the F1-score provides a balanced measure of both precision and recall. The experimental results demonstrated that the Random Forest Classifier achieved a high level of accuracy in predicting suitable educational apps compared with traditional

classification methods such as Decision Trees and Logistic Regression.

The results also showed that incorporating NLP-based feature extraction significantly improved the model's ability to understand application descriptions and categorize them effectively. By analyzing textual information along with other features such as ratings and usage patterns, the system was able to capture deeper insights into the relevance of different applications for specific academic levels. This combination of behavioral data and content-based features enabled the recommendation system to generate more personalized and context-aware suggestions.

Another important observation from the experimental analysis was the ability of the Random Forest model to handle complex and high-dimensional datasets efficiently. Since the dataset contained both textual and numerical attributes, the ensemble nature of the algorithm helped balance the influence of different features during prediction. And the feature importance analysis provided by the Random Forest model helped identify which attributes had the greatest impact on the recommendation process. Features such as app category, textual description keywords, and user engagement patterns were found to be among the most influential factors in determining the suitability of an application.

Fig 1: Mobile App Recommendation Sign up page

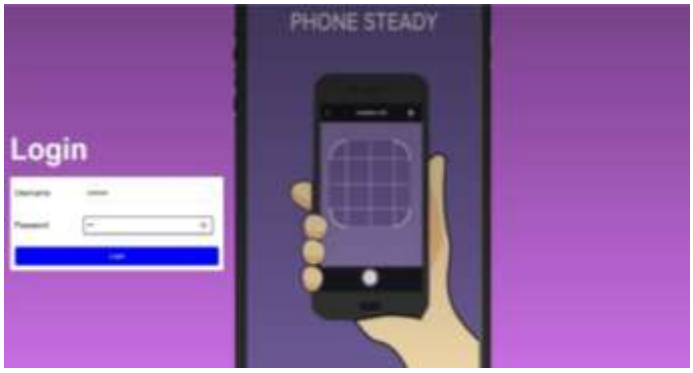

Fig 2: Login page

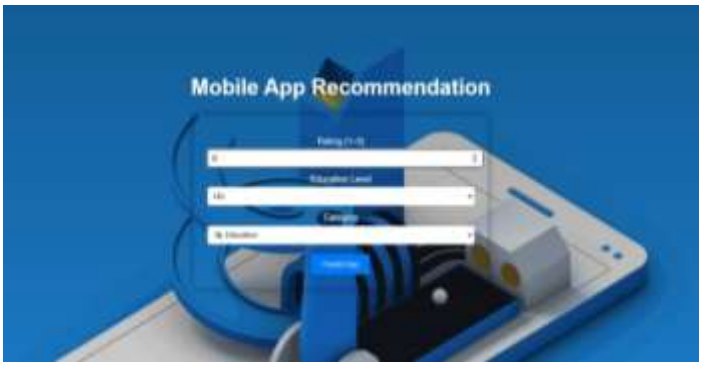

Fig 3: filling app recommendation details

Fig 4: Prediction result

The discussion of the results highlights that the proposed recommendation system successfully addresses the challenge of identifying relevant educational applications from a large pool of available apps. Students often face difficulty in selecting appropriate learning tools due to the overwhelming number of options in mobile app stores. By providing personalized recommendations based on data-driven analysis, the system helps users discover applications that match their academic needs and learning preferences.

The scalability of the system suggests that it can be applied to larger datasets and integrated into real-world educational platforms. With additional data and continuous learning, the system could further improve its prediction accuracy and recommendation quality. The results also suggest that integrating additional contextual features such as user feedback, learning outcomes, and time-based interaction data could further enhance the performance of the system.

The experimental results confirm that the proposed approach effectively combines crowdsourced educational data, Natural Language Processing techniques, and machine learning algorithms to develop an intelligent mobile app recommendation system. The use of the Random Forest Classifier enables the system to generate accurate, reliable, and interpretable recommendations for students across different academic levels. This study demonstrates the potential of artificial intelligence in improving educational technology and supporting personalized digital learning environments.

SYSTEM ARCHITECTURE:

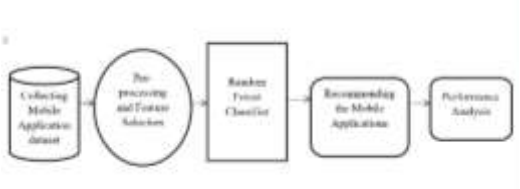

Fig 5: System Architecture

The system architecture of the proposed mobile app recommendation system consists of several modules that work together to provide personalized educational app suggestions. The Data Collection Module gathers information about educational applications and user interaction data such as app descriptions, categories, ratings, and usage patterns.
Then the Data Preprocessing Module cleans and processes the collected data using Natural Language Processing techniques such as text cleaning, tokenization, and stop-word removal. After preprocessing, the **Feature** Extraction Module converts the textual data into numerical features using techniques like TF-IDF or vectorization.
These features are then used to train a Random Forest Classifier, which acts as the core machine learning model. The model analyses patterns in the dataset and predicts suitable applications for different student levels such as Undergraduate, Postgraduate, and Graduate. Then Recommendation Module generates personalized educational app suggestions for users based on the trained model's predictions.

6.CONCLUSION

This research presents a mobile application recommendation system designed to assist students in identifying suitable educational apps based on their learning needs. By utilizing crowdsourced educational data, the system analyzes application usage patterns and textual app descriptions to understand user preferences. Natural Language Processing techniques are applied to preprocess and extract meaningful features from the data.

The Random Forest Classifier is used as the core machine learning model due to its robustness, accuracy, and ability to handle high-dimensional data. The model successfully classifies and recommends educational applications for students at different academic levels, including Undergraduate, Postgraduate, and Graduate students. Experimental evaluation demonstrates that the proposed system provides accurate and reliable recommendations.

The system improves the process of discovering educational applications by offering personalized suggestions that support effective learning. The proposed approach highlights the potential of combining machine learning and NLP techniques to enhance digital learning environments and assist students in selecting the most relevant educational tools.

7. FUTURE SCOPE

Our mobile app recommendation system can be further improved by incorporating advanced machine learning and deep learning techniques such as BERT, LSTM, and transformer-based models to enhance recommendation accuracy. Future work may also include integrating real-time user interaction data, such as app usage time and user engagement patterns, to generate more dynamic and personalized recommendations. Additionally, implementing hybrid recommendation approaches that combine collaborative filtering and content-based methods can help overcome challenges like cold-start and data sparsity. The system can also be expanded to support multilingual applications and integrated with university Learning Management Systems (LMS) to provide course-related app recommendations. These enhancements can make the system more efficient, scalable, and useful for improving personalized digital learning experiences.

REFERENCES

[1] M. Zhai, X. Wang, and X. Zhao, ‘‘The importance of online customer reviews characteristics on remanufactured product sales: Evidence from the mobile phonemarketonAmazon.com,’’J.RetailingConsum.Services, vol. 77, Mar. 2024, Art. no. 103677. [2] C. C. Aggarwal, ‘‘An introduction to recommender systems,’’ in Rec ommender Systems: The Textbook. Cham, Switzerland: Springer, 2015, pp. 1–28.

[3] G. Linden, B. Smith, and J. York, ‘‘Amazon.com recommendations: Item to-item collaborative filtering,’’ IEEE Internet Comput., vol. 7, no. 1, pp. 76–80, Jan. 2003.

[4]N.TintarevandJ.Masthoff,‘‘Beyondexplainingsingleitemrecommenda tions,’’ in Recommender Systems Handbook. Cham,Switzerland: Springer, 2022, pp. 711–756.

[5] C. Musto, M. de Gemmis, P. Lops, F. Narducci, and G. Semeraro, ‘‘Semantics and content-based recommendations,’’ in Recommender Sys tems Handbook. Cham, Switzerland: Springer, 2022, pp. 251–298.

[6] P. Winoto and T. Y. Tang, ‘‘The role of user mood in movie recommenda tions,’’ Expert Syst. Appl., vol. 37, no. 8, pp. 6086–6092, Aug. 2010.

[7] P. Li and A. Tuzhilin, ‘‘A dynamical system framework for explor ing consumer trajectories in recommender system,’’ SSRN, May 2024, pp. 1–55.

[8] X. Su and T. M. Khoshgoftaar, ‘‘A survey of collaborative filtering tech niques,’’ Adv. Artif. Intell., vol. 2009, no. 1, Oct. 2009, Art. no. 421425.

[9] M. Asad, S. Shaukat, E. Javanmardi, J. Nakazato, and M. Tsukada, ‘‘A comprehensive survey on privacy-preserving techniques in feder ated recommendation systems,’’ Appl. Sci., vol. 13, no. 10, p. 6201, May 2023.

[10] M. Singh, ‘‘Scalability and sparsity issues in recommender datasets: Asurvey,’’ Knowl. Inf. Syst., vol. 62, no. 1, pp. 1–43, Jan. 2020.

[11] M. Moser, ‘‘College Tennis University recommendation system (CTURS),’’ M.S. thesis, Dept. Psychol., Univ. Salzburg, Salzburg, Austria, 2022.

[12] I.Hossain,M.A.H.Palash,A.T.Sejuty,N.A.Tanjim,M. A. A. L. Nasim, S. Saif, A. B. Suraj, M. M. A. Haque, and N. Karim, ‘‘A survey of recommender system techniques and the ecommerce domain,’’ 2022, arXiv:2208.07399.

[13] G. B.Martins, J. P. Papa, and H. Adeli, ‘‘Deep learning techniques for rec ommender systems based on collaborative filtering,’’ Expert Syst., vol. 37, no. 6, p. 12647, Dec. 2020.

[14] F. Camilli and M. Mézard, ‘‘The decimation scheme for symmetric matrix factorization,’’ J. Phys. A, Math. Theor., vol. 57, no. 8, Feb. 2024, Art. no. 085002.

[15] H. Nabli, R. B. Djemaa, and I. A. B. Amor, ‘‘Improved clustering-based hybrid recommendation system to offer personalized cloud services,’’ Cluster Comput., vol. 27, no. 3, pp. 2845–2874, Jun. 2024.

[16] A. A. Ka’bi, ‘‘Proposed artificial intelligence algorithm and deep learning techniques for development of higher education,’’ Int. J. Intell. Netw., vol. 4, pp. 68–73, Jun. 2023.

[17] W. Wang, W. Chen, Y. Luo, Y. Long, Z. Lin, L. Zhang, B. Lin, D. Cai, and X. He, ‘‘Model compression and efficient inference for large language models: A survey,’’ 2024, arXiv:2402.09748.

[18] R. Nahta, G. S. Chauhan, Y. K. Meena, and D. Gopalani, ‘‘Deep learning with the generative models for recommender systems: A survey,’’ Comput. Sci. Rev., vol. 53, Aug. 2024, Art. no. 100646.

[19] J. H. Yoon and B. Jang, ‘‘Evolution of deep learning-based sequential recommender systems: From current trends to new perspectives,’’ IEEE Access, vol. 11, pp. 54265–54279, 2023.