2025 IEEE International Conference on Signal Processing, Information, Communication and Systems (SPICSCON)
21–22 November 2025, Department of ICE, University of Rajshahi, Rajshahi-6205, Bangladesh

HRECSYS++: A Personalized Hybrid Movie Recommender System with Dynamic Weighting

Nahid Karim Emon
Dept. of Computer Science and Engineering
Jashore University of Science and Technology
Jashore, Bangladesh
nahidkarimemon@gmail.com

Md Nasim Adnan
Dept. of Computer Science and Engineering
Jashore University of Science and Technology
Jashore, Bangladesh
nasim.adnan@just.edu.bd

Md Shojib Hossain
Dept. of Computer Science and Engineering
Jashore University of Science and Technology
Jashore, Bangladesh
mdshojib922@gmail.com

Prakash Chandra Mondal
Information and Communication Technology Dept.
Bangladesh Bank
Dhaka, Bangladesh
prakash.mondal@bb.org.bd

***Abstract*—The rapid growth of digital contents and user interactions in modern technology has made it more challenging to deliver precise and personalized suggestions. The cold-start problem, data sparsity, and scalability problems are common problems with traditional methods including content-based, collaborative, and demographic filtering. This work offers a hybrid movie recommender system that combines the advantages of demographic, collaborative, and content-based filtering techniques in order to overcome these problems. The proposed approach makes use of advanced Machine Learning techniques, such as K-Means, Cosine Similarity, and PageRank algorithms, to enhance recommendation accuracy, variety, and scalability. A dynamic weighted hybrid technique is used to cluster movie attributes, identify user similarities, and rank recommendations based on their relevance. The system creates highly customized recommendations for every user by combining user preferences, demographic data, and the popularity of movies.**

***Keywords*—Movie Recommendation System, Hybrid Filtering, Content-based Filtering, Collaborative Filtering, Demographic Filtering**

I. Introduction

Recommender Systems (RecSys) are now widely used, particularly in the entertainment industry, because of the rapid expansion of digital content and developments in Artificial Intelligence (AI). By offering personalized content recommendations based on each user's watching tastes and habits, RecSys greatly improves the user experience as users are faced with an enormous number of content possibilities. These methods have a direct impact on platform profitability in addition to increasing user engagement. Netflix, for instance, attributes to approximately 80% of its streamed hours to personalized recommendations, highlighting RecSys's critical role in user retention and content consumption [1]. Similarly, Uber Eats implemented an efficient recommendation system that notably increased consumer retention, conversion rates, and gross bookings, forecasting a revenue boost of approximately $1.5 million weekly if deployed globally [2]. Collaborative Filtering (CF), Content-based Filtering (CBF), and Demographic Filtering (DF) are three well-known RecSys approaches. By recognizing similar user behaviors, CF uses past user interactions to recommend appropriate details. Despite its efficacy, CF has drawbacks, including data sparsity, which lowers recommendation accuracy, and the cold-start issue, which affects new users or items without prior data. On the other hand, CBF uses inherent item characteristics such as cast, director, genre etc. to generate recommendations that are very similar to a user's past choices. Despite being extremely customized, this method frequently leads to overspecialization, which thereby narrowing the range of recommendations. Using user demographic information like nationality, language, religion, education etc. to group users and leverage collective trends to make meaningful recommendations even with minimal individual interaction data, DF overcomes cold-start problems. To overcome traditional RecSys limitations, advanced Machine Learning (ML) techniques have been integrated, significantly improving recommendation quality. Cosine Similarity (CS), for instance, measures user-item vector similarities to enhance recommendation precision [3]. K-Means refines recommendation relevance by grouping similar movies, enhancing structured content organization. Additionally, graph-based ranking approaches like PageRank (PR) prioritize influential movies based on their network connections, further refining recommendation accuracy and contextual relevance [4]. Accuracy, diversity, flexibility, and scalability are some of the main RecSys problems that are successfully addressed by combining these innovative techniques within a hybrid recommendation system, which eventually guarantees more engaging and personalized user experiences.

The remainder of this paper is structured as follows: Section II provides a Literature Review, examining existing RecSys and identifying areas for improvement. In Section III details on the Proposed Work is presented, outlining the recommendation framework and the integration of advanced filtering tech-

979-8-3315-7316-4/25/$31.00 ©2025 IEEE

niques. Section IV describes the Implementation process in detail, focusing on system development and recommendation generation. Section V presents the Discussion, comparing the proposed work with existing ones. In section VI, we offer the concluding remarks.

II. Literature Review

With the growth of ML and AI techniques, Movie Recommender Systems (MRecSys) experience significant changes and are now crucial for the delivery of contents on streaming platforms. User preservation, satisfaction, and engagement are all directly impacted by effective recommender systems. Hybrid Recommender Systems (HRecSys) are unique among different approaches because they successfully overcome the drawbacks of independent ones like CF, CBF and DF. HRecSys utilizes CF, CBF and sophisticated ML techniques including PR, K-Means, and CS to improve accuracy and scalability in order to address these issues. CS enhances similarity metrics, resulting in more accurate suggestions. K-Means improves recommendation efficiency by systematically grouping users and items. PR ensures highly relevant content prioritization by optimizing recommendation rankings based on items' relationship strength. Numerous hybrid techniques have been explored in recent years, each with significant advantages and disadvantages. In [5], the authors emphasized the lack of adaptive weighting methods for changing user preferences while demonstrating the accuracy and user satisfaction benefits of hybrid models. Using Markov Chains, in [6] the authors suggested a hybrid sequential strategy that enhanced user behavior tracking but lacked integration of real-time interactions. In [7], the authors created an optimized hybrid model by merging CF and CBF, however they neglected to include important ranking processes like PR. In [4], the authors used matrix factorization to address data sparsity, however their static weighting had limited adaptability.

Although they improved scalability and mitigated cold-start issues, other studies like [8], [9] were unable to dynamically balance recommendations. In [3], the authors successfully employed LSTM-based hybrid filtering at the cost of computational inefficiencies. In [10], the authors combined CF and ML techniques to increase accuracy, but personalization was constrained by the lack of DF. In [11], the authors used word embedding to comprehend semantics, however they did not use multidimensional filtering. To increase accuracy, in [12] the authors used CS to convert text to numbers, however they failed to consider clustering algorithms for user similarity modeling. In [13], the authors presented clustering-based techniques that increase satisfaction but limit diversity by over-segmenting. RNN and Naive Bayes techniques were successfully used in studies like [14], [15], respectively, although they encountered issues with scalability and feature representation. In [16], the authors employed CNN models, which resulted in accuracy but performance issues. In [17], the authors used CS to successfully handle cold-start concerns, but they lacked adaptive ranking.

Existing HRecSys face persistent challenges, including the lack of adaptive weighting, insufficient real-time adaptability, limited multidimensional filtering, and inadequate ranking optimizations. Fixed weight assignments limit adaptability to evolving user preferences, while the absence of DF worsens the cold-start issue. Additionally, limited use of advanced ranking techniques, such as PR, hampers prioritization and recommendation diversity. Over-specialization, especially in CBF, further limits user exploration. Addressing these gaps can significantly improve recommendation accuracy, personalization, scalability, and user satisfaction.

III. Proposed Work

HRecSys++ is the proposed personalized HRecSys that integrates CF, CBF and DF to improve recommendation accuracy, scalability, and adaptability. Lack of user behavior adaptation is a common problem with traditional RecSys. HRecSys++ gets beyond these limitations by using real-time ranking algorithms, adaptive weighting, and a structured knowledgebase to make sure that suggestions are constantly improved in response to user interactions. The system uses a suggestion process with multiple stages. Experienced users are given recommendations based on CF and CBF, while new users with insufficient interaction data are initially given DF. HRecSys++ uses current user feedback and interaction patterns to dynamically modify the impact of each filtering technique. Gathering user data, including viewing history, ratings, likes, and general preferences, is the first step in the suggestion process. The system dynamically chooses a suitable filtering method based on the available data: DF for users with no prior data, CF for similar user identification, or CBF for movie attributes analysis. Using adaptive weighting and PR score, HRecSys++ integrates both approaches when content-based and collaborative data are available. The final recommendations are calculated as per Equation 1.

$$S_{hybrid}(i) = w_{cbf} \cdot S_{cbf}(i) + w_{cf} \cdot S_{cf}(i) + w_{pr} \cdot PR(i) \tag{1}$$

where: $S_{hybrid}(i)$ is the final hybrid score for movie i, $S_{cf}(i)$ is the collaborative similarity score for movie i, $S_{cbf}(i)$ is the content-based similarity score for movie i, $PR(i)$ is the PageRank score for movie i, and w_{cf}, w_{cbf}, w_{pr} are adaptive weights for collaborative, content-based, and PageRank scores.

HRecSys++ process is depicted in Figure 1, which shows how filtering strategies, similarity metrics, and ranking methods interact. HRecSys++ keeps an organized knowledgebase that changes as a result of system input and human interactions. It supports real-time adaptability by storing user profiles, recommendation history, similarity measurement criteria, and movie information. The knowledgebase was first created using expert-defined similarity criteria, but it continuously learns to improve relationships and criteria in response to changing user preferences and marketplace developments as depicted in Figure 2.

HRecSys++ is supported by three ML algorithms. Here K-Means is used to improve CBF by classifying films according

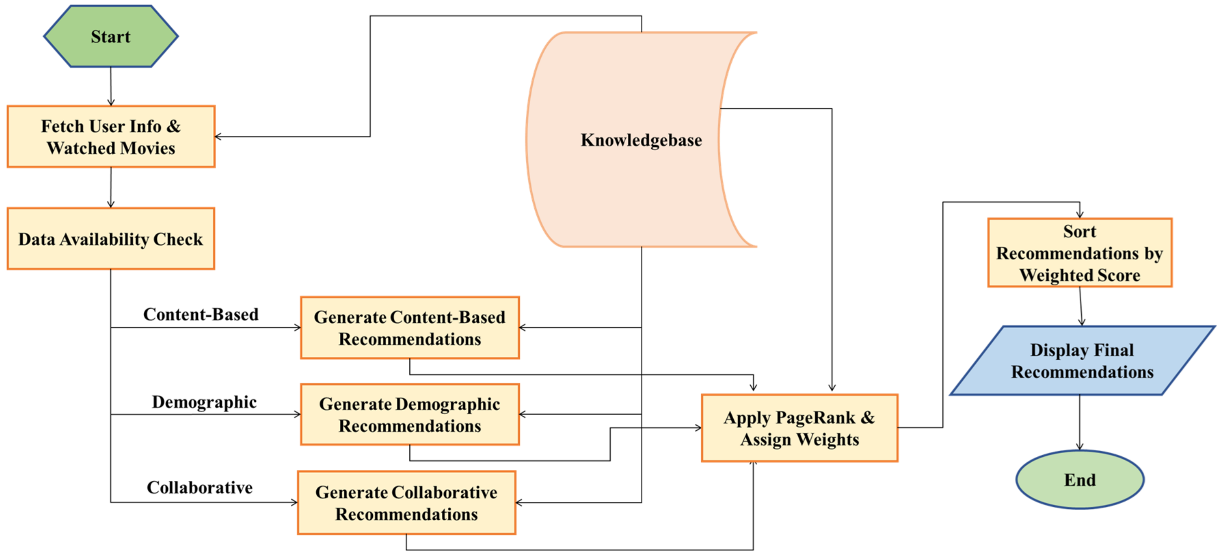

Fig. 1. Flowchart of the proposed HRecSys++

to director, cast, and genre. It optimizes groupings to provide structured and contextually relevant recommendations by minimizing variance within clusters. CS is used for finding user preference patterns within CF since it calculates the similarity of individuals or items based on vector orientations. PR prioritizes suggestions that are regularly referenced by users according to Equation 2.

$$PR(i) = (1 - d) + d \cdot \sum_{j \in M(i)} \frac{PR(j)}{L(j)} \tag{2}$$

where: $PR(i)$ is the PageRank score of movie i, representing its relative importance among movies, $M(i)$ is the set of movies linked to i, contributing to its importance, $L(j)$ is the number of outgoing links from movie j to other movies, and d is the damping factor, set to 0.85 in HRecSys++, representing the probability of continuing to follow links.

HRecSys++ begins by clustering new users through DF using the K-Means algorithm. Following clustering, the model generates initial movie recommendations tailored to the user's demographic profile. Users are empowered to influence recommendations by manually assigning or updating weights of DF, CBF and CF methods, aligning the suggestions more closely with their preferences. Besides, users can choose between randomly generated weights if they want to avoid manually assigning the weights and are dissatisfied with the recommendation. Finally, PR is integrated to prepare the list movies recommended by HRecSys++.

IV. Implementation

HRecSys++, a Personalized HRecSys with dynamic weighting has been implemented using the Laravel (PHP framework), HTML, CSS, and JavaScript. Since ML models are developed directly in PHP, web-based applications can use them without the need for additional Python-based ML libraries. The following link will take you to GitHub where you may view the complete source code and implementation details.

https://github.com/nahid-karim-emon/Movie_Recommendation_System

A. Recommendation Generation Process

HRecSys++ combines several recommendation methods, including DF, CBF, CF, and hybrid approaches, to produce personalized movie recommendations. By dynamically updating recommendations based on real-time user involvement and interactions, the system cleverly adjusts to patterns of user behavior. Individuals with little prior experience are guaranteed to obtain insightful recommendations because of DF. The algorithm makes sure that new users receive appropriate recommendations even in the absence of previous interactions by looking at demographic-based patterns of behavior. As seen in Figure 3, viewers are given movie recommendations that are specific to their profile features based on demographic data.

HRecSys++ is in effect an HRecSys that integrates CBF and CF outputs according to PageRank scores and adaptive weighting. The system uses user feedback and system learning to dynamically allocate weights to each technique when both collaborative and content-based data are available. By evaluating each movie's relative value inside the system, the PR algorithm further improves the final ranking and makes sure that the most significant and contextually relevant movies are at the top of the list. Figure 4 illustrates how the hybrid recommendation model combines CBF and CF, dynamically modifying the weighting in response to user input and system learning.

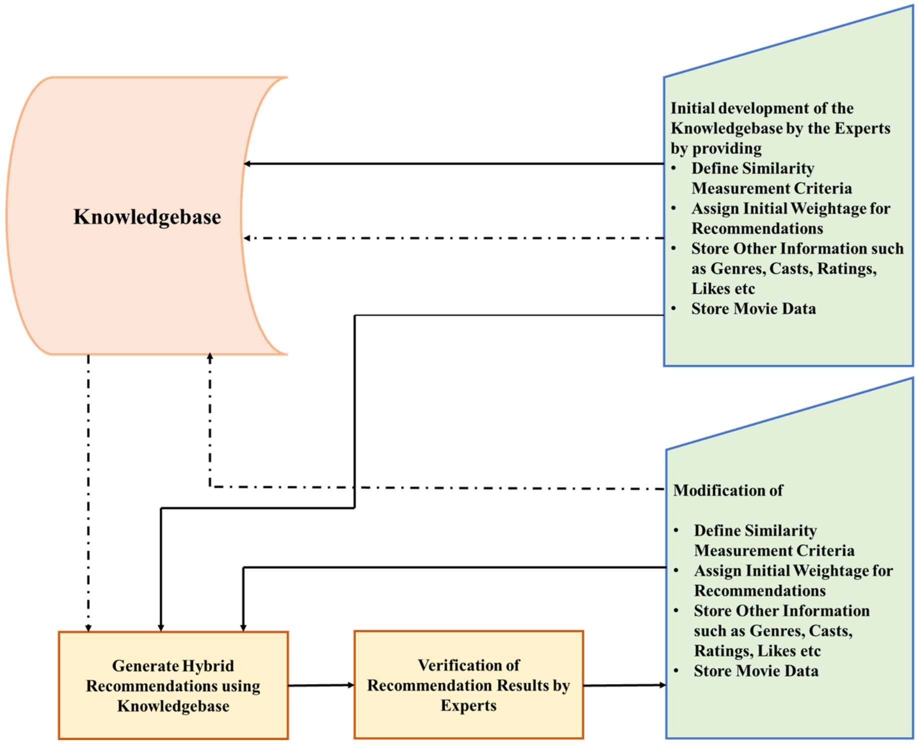

Fig. 2. Knowledgebase development for the proposed HRecSys++

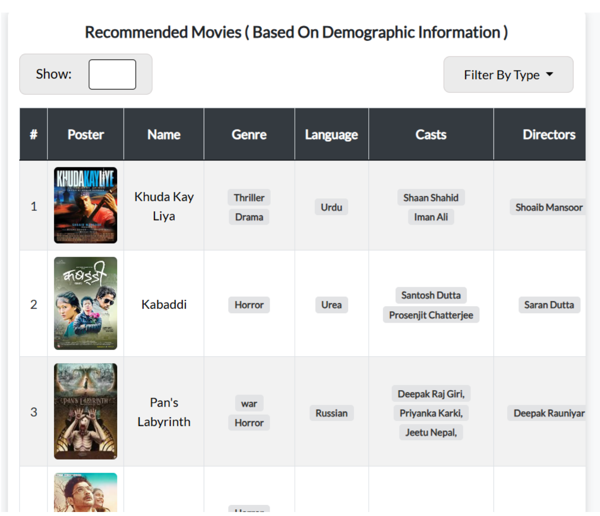

Fig. 3. Recommended movies based on DF

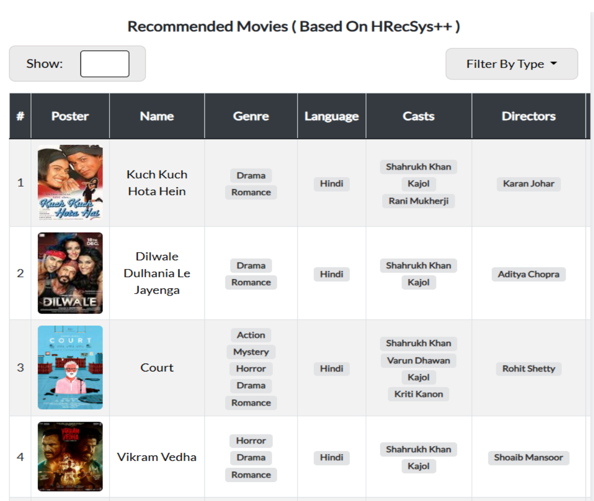

Fig. 4. Recommended movies based on HRecSys++

V. Discussion

In this section, the HRecSys++ is addressed, emphasizing its advantages over current recommender systems by combining sophisticated ranking algorithms like PR and clustering techniques like K-means with CF, CBF, and DF. In a novel way, HRecSys++ uses adaptive weighting to dynamically modify filtering techniques in response to real-time user inputs, greatly enhancing accuracy and customization. By employing demographic-based clustering, the system efficiently addresses cold-start problems and guarantees pertinent suggestions for new users. Improved similarity metrics, particularly cosine similarity, enable better handling of data sparsity and generate correct suggestions even when interaction data is scarce. By integrating PageRank, the ranking system is further optimized and highly influential films are given priority. Table I summarizes the key improvements HRecSys++ offers compared to conventional hybrid recommendation systems. By continuously improving suggestions based on user interactions,

TABLE I
COMPARISON OF HRECSYS++ WITH EXISTING HYBRID RECOMMENDER MODELS

Feature	Conventional hybrid models	HRecSys++
Cold-start problem	Moderate	Low (demographic filtering mitigates cold-start issues)
Data sparsity handling	Improved but limited	Highly effective (cosine similarity enhances performance)
Ranking mechanism	Weighted hybrid ranking	Combined weighted ranking and PageRank-based ranking with dynamic prioritization
Personalization	Improved but still data-limited	Highly adaptive to user behavior and preferences
Scalability	Moderate	Highly scalable with k-means clustering and dynamic filtering

HRecSys++ improves the user experience in real-time. Users can explore a variety of content across several dimensions, such as genre, director, and user likeness, according to this technique, which ensures suggestion diversity and relevancy.

VI. Conclusion and Future Scope

A. Conclusion

HRecSys++, an HRecSys that combines CF, CBF and DF to provide personalized movie recommendations, is developed and put into use in this study. The method effectively overcomes important issues like data sparsity and ranking inefficiencies, which frequently impact conventional recommendation algorithms, by integrating these strategies. HRecSys++'s dynamic weighting technique, which modifies the contributions of CF, CBF and DF in real-time based on user feedback, is one of its main advantages. This guarantees that when user behavior changes, the system will always improve its suggestion method. The system's capacity to incorporate various filtering strategies and dynamic ranking processes guarantees that suggestions stay precise and relevant to current circumstances.

B. Future Scope

HRecSys++ has made great strides, but further research is needed in a few areas to improve scalability, real-time adaptation, and overall suggestion quality. Scalability optimization is a crucial area that needs to be improved. System responsiveness may be impacted by rising computational overhead, particularly for K-Means as the number of users and videos keeps growing. Future research should concentrate on refining these algorithms through the use of Approximate Nearest Neighbor (ANN) search techniques, which provide faster calculations without sacrificing recommendation accuracy, or Principal Component Analysis (PCA) for dimensionality reduction. An important chance to raise the quality of recommendations is provided by Deep Learning integration. By improving high-order feature interactions, Neural Collaborative Filtering (NCF) can produce predictions that are more accurate. Furthermore, different ensemble algorithms can be incorporated in order to analyze usage patterns more effectively [18], [19].

References

[1] A. Lah, "Algorithms in Streaming Services", AMT Lab @ CMU, August 2021.
[2] S. G. of Business, "A Better Way to Make Recommendations: The Power of Popular Platforms", Stanford Business Insights, August 2021.
[3] M. N. Ilyas and E. B. Setiawan, "Weight-Based Hybrid Filtering in a Movie Recommendation System Based on Twitter with LSTM Classification", JURNAL MEDIA INFORMATIKA BUDIDARMA, Vol. 7, No. 4, pp. 1838–1849, 2023.
[4] Z. Liu and F. Ren, "Algorithm Improvement of Movie Recommendation System Based on Hybrid Recommendation Algorithm", Frontiers in Computing and Intelligent Systems, Vol. 3, No. 3, pp. 113–117, 2023.
[5] N. Ifada, T. F. Rahman, and M. K. Sophan, "Comparing Collaborative Filtering and Hybrid Based Approaches for Movie Recommendation", Proceeding of 2020 6th Information Technology International Seminar (ITIS), Surabaya, pp. 219–223, October 2020.
[6] Y. Yang, H.-J. Jang, and B. Kim, "A Hybrid Recommender System for Sequential Recommendation: Combining Similarity Models with Markov Chains", IEEE Access, Vol. 8, pp. 190136–190146, 2020.
[7] L. Li, Z. Zhang, and S. Zhang, "Hybrid Algorithm Based on Content and Collaborative Filtering in Recommendation System Optimization and Simulation", Scientific Programming, Vol. 2021, No. 1, pp. 7427409, 2021.
[8] I. A. Amaefule, C. C. Ibebuogu, and U. L. Chilaka, "A Movie Data Recommendation System Using a Hybrid Model", 2024.
[9] A. M. Nasser, J. Bhagat, A. Agrawal, and T. Joshva, "Mean-Reversion Based Hybrid Movie Recommender System Using Collaborative and Content-Based Filtering Methods", 2023.
[10] S. Bohra, A. Gaikwad, and G. Singh, "Hybrid Machine Learning Based Recommendation Algorithm for Multiple Movie Dataset", Indian Journal of Science and Technology, Vol. 16, No. 37, pp. 3121–3128, 2023.
[11] V. Nguyen, T.-H. Nguyen, J. J. Jung, and D. Camacho, "Extending Collaborative Filtering Recommendation Using Word Embedding: A Hybrid Approach", Concurrency and Computation: Practice and Experience, Vol. 35, No. 16, pp. e6232, 2023.
[12] M. R. Hasan and J. Ferdous, "Dominance of AI and Machine Learning Techniques in Hybrid Movie Recommendation System Applying Text-to-number Conversion and Cosine Similarity Approaches", Journal of Computer Science and Technology Studies, Vol. 6, No. 1, pp. 94–102, 2024.
[13] P. Bahrani, B. Minaei-Bidgoli, H. Parvin, M. Mirzarezaee, and A. Keshavarz, "A Hybrid Semantic Recommender System Based on an Improved Clustering", The Journal of Supercomputing, pp. 1–45, 2024.
[14] B. M. G. Al Awienoor and E. B. Setiawan, "Movie Recommendation System Based on Tweets Using Switching Hybrid Filtering with Recurrent Neural Network", International Journal of Intelligent Engineering & Systems, Vol. 17, No. 2, 2024.
[15] B. Chandini and B. Rupadevi, "Movies Recommendation System Using Hybrid Filtering and Naive Bayes Algorithm", 2024.
[16] M. Alrashidi, R. Ibrahim, and A. Selamat, "Hybrid CNN-based Recommendation System", Baghdad Science Journal, Vol. 21, No. 2 (SI), pp. 592–592, 2024.
[17] I. Adeoye, "Unlocking Film Fandom: AI and Machine Learning's Reign in Hybrid Movie Recommendations Through Text-to-Number Conversion and Cosine Similarity", Available at SSRN, 2024.
[18] M. N. Adnan and M. Z. Islam, "Forest CERN: A New Decision Forest Building Technique", Proceeding of Lecture Notes in Computer Science, pp. 304–315, 2016.
[19] M. N. Adnan and M. Z. Islam, "Forest PA: Constructing a Decision Forest by Penalizing Attributes Used in Previous Trees", Expert Systems with Applications, Vol. 89, pp. 389–403, 2017.