SENTIMENT DRIVEN HYBRID MOVIE RECOMMENDATION SYSTEM

Prajjawal Kansara
School of Computer Applications and Technology
Galgotias University
Greater Noida, UP, India
prajjawalkansara@gmail.com

Akash Yadav
School of Computer Applications and Technology
Galgotias University
Greater Noida, UP, India
kiraacodes@gmail.com

Hammad Hassan Khan
School of Computer Applications and Technology
Galgotias University
Greater Noida, UP, India
khanhammad00.hk@gmail.com

Vineeta Khemchandani
School of Computer Applications and Technology
Galgotias University
Greater Noida, UP, India
vineeta.khemchandani@galgotiasuniversity.edu.in

***Abstract*—Conventional movie recommendation models typically depend on content-based filtering and collaborative filtering, but both have issues with sparsity, cold-start scenarios, and lack of personalization. This work proposes a hybrid movie recommendation system combining content-based filtering, collaborative filtering, and neural collaborative filtering (NCF) to boost the accuracy of recommendations. Further, aspect-based sentiment analysis (ABSA) is used to distill insightful knowledge from user comments to provide emotionally intelligent suggestions. The framework updates recommendations dynamically based on real-time user interaction, making the system more adaptive and personalized. Our method improves scalability and fairness with the use of graph-based recommendations and bias remediation strategies.**

I. INTRODUCTION

The exponential growth in digital content, particularly films, has left audiences overwhelmed by the sheer number of options on different platforms. To help users find films that appeal to their tastes, movie recommendation systems have become indispensable tools. While methods based on tradition, e.g., collaborative filtering and content-based, have proved partially successful, they tend to fail to properly capture the fine grained nature of user preferences and frequently ignore useful information contained in user reviews. These systems focus mainly on structured data—such as genres, directors, and actors—without paying attention to unstructured data, e.g., emotional responses expressed in reviews.

Sentiment analysis, an emerging area within Natural Language Processing (NLP), offers a promising avenue to enhance recommendation systems by examining the emotional content present in user reviews. This additional layer of personalization provides deeper insights into viewers' emotional responses to films, thereby improving the recommendation process. By combining structured metadata from sources such as IMDb, TMDb, and Wikipedia with sentiment analysis, we are able to create a more sophisticated system that addresses user preference as well as emotional response.

This work introduces a hybrid movie recommendation system that blends structured metadata (e.g., genres, cast, and directors) with unstructured data from user reviews and sentiment analysis. This blending enables the system to make more personalized and emotionally sensitive recommendations. Our research seeks to improve movie recommendation systems by overcoming conventional limitations and promoting a more responsive, user-centric approach that maximizes satisfaction and engagement.

In the following sections, we will detail the methodology, implementation, and evaluation of our proposed system and discuss its capability to transform the recommendation scenario.

II. LITERATURE REVIEW

Recommendation systems for movies are essential for sites like Netflix, Amazon Prime, and IMDb, providing users with individualized recommendations from huge collections of movies. Recommendation systems have undergone great improvement over the years with various techniques used to enhance precision and user satisfaction. Early recommendation systems heavily depended on content feature and user interaction data for recommending. Later models, though, employed hybrid models, machine learning, deep learning, and even sentiment analysis for furthering personalization. The present literature review delves into the major strategies in the field, with focus on sentiment analysis and combination of structured and unstructured data for movie recommending. Ashrita Kashyap1 et al. [1] proposed a movie recommendation system, Movie REC, implemented using CAD tools and Blender.

A. Collaborative Filtering and Co-Clustering

Collaborative filtering [7] has come a long way with sparsity and scalability given prime importance in recent work. Co-clustering, where users and movies are clustered together, has proven to be successful in predictive missing ratings as well as enhancing personalization. Airena and Agrawal (2023) demonstrated the improvement in recommendation quality by 7.91% with the use of such techniques. Meenu Gupta et al. [2] pushed predictive accuracy to new levels with the use of KNN algorithms coupled with collaborative filtering over content-based systems.

Katarya et al. [3] introduced an optimization-augmented hybrid system that integrates k-means clustering with cuckoo search and effectively overcame the weaknesses of conventional recommendation systems.

Model-based filtering techniques encompass an array of learning paradigms such as clustering algorithms, neural frameworks, association rule mining, and probabilistic models such as Bayesian networks [8]. These techniques construct predictive models through learning latent patterns in past data to produce scalable and generalized recommendations. Memory-based methods, however, make direct use of user-item interaction data with no training process in between. These are typically categorized into user-based (client-based) and item-based (object-based) methods [9], which leverage similarity calculations over the entire dataset to offer real-time, instance-based recommendations.

B. Content Filtering with Sentiment Integration

Current advances in content-based filtering (CBF) [11, 12] have shifted towards context-aware methods that utilize user engagement signals and semantic metadata. Hashim and Waden et al. [10] emphasized the application of social interaction signals—likes and shares—to more dynamically infer user intent. However, CBF models are low in diversity, favoring content with similar surface-level features.

To enhance depth and emotional relevance, sentiment analysis is integrated into the recommendation process. By excavating the affective tone of user comments, systems are able to recognize affective tastes and make adjustments in recommendations accordingly. For example, users inclined towards content that was reviewed favorably can be recommended emotionally congruent content. Mostafa Khalaji et al. [4] alleviated cold-start problems by incorporating such contextual sentiment cues into their hybrid HMRS-RA model, with significantly better predictions for newly added movies.

C. Sentiment-Enhanced Hybrid Models

State-of-the-art hybrid recommenders combine content-based and collaborative approaches to resolve the weaknesses inherent in each of them. Fresh research by Paranjape et al. [16] notes the importance of incorporating user participation metrics—namely, frequency of interaction, reviews, and social signals—into the hybrid pipeline, with the effect being more sophisticated personalization.

Adding sentiment-sophisticated features to hybrid models further enhances their performance. By aligning recommendations with not only behavior but also the deduced emotional tone of user-posted reviews, the system adjusts dynamically to the mood and affect of the user. This layer of emotional intelligence adds a richer personalization dimension, ensuring personalized content that engages on a level beyond past consumption behavior.

D. Deep Learning and Neural Collaborative Filtering

Neural Collaborative Filtering (NCF) has proved to be a strong competitor to factorization-based approaches, providing better performance in representation of high-dimensional, sparse user-item interactions. With the utilization of deep neural networks in learning high-order, non-linear relationship, NCF models have shown significant improvements in recommendation accuracy on large datasets.

Beyond structured interactions, deep learning architectures learn to handle unstructured data like reviews. When combined with NLP-based sentiment analysis, these models are able to extract emotional subtlety in user comments, allowing sentiment-guided representation of user preference. This integration amplifies contextual sensitivity and personalization, such that recommendations become better aligned with user intention and affective state.

E. Addressing the Limitations of Traditional Methods

Data sparsity is still the core challenge in conventional recommendation models, which typically degrade performance owing to scarce user-item interactions. Modern methods have turned toward representation learning approaches that learn to model implicit relationships between sparse data. Airena and Agrawal (2023) mitigated this challenge through co-clustering, in which the concurrent clustering of users and items facilitates more trustworthy estimation of missing ratings.

Meanwhile, deep learning-based models—more specifically, autoencoders and neural architectures—have demonstrated impressive ability to extract latent representations out of sparse explicit feedback. Through their capacity for integrating structured and unstructured signals, such as review sentiment and contextual metadata.

F. Future Trends and Research Directions

Future work in film recommendation systems will probably be directed towards enhancing the transparency and explainability of models, privacy, and fairness. As neural networks and deep learning become increasingly sophisticated, it is critical to offer transparent explanations of how recommendations are generated, which will build trust with users. Furthermore, incorporating external knowledge, including social media sentiment, critic reviews, and sentiment information, will probably continue to enhance relevance and accuracy of recommendations.

G. Conclusion of the Literature Review

The movie recommendation system space is moving very fast, fueled by developments in machine learning, deep learning, and sentiment analysis. Older methods such as collaborative and content-based filtering remain core, but hybrid models combining several sources of data—such as structured metadata, unstructured reviews, and sentiment analysis—are increasingly prominent. These methods are more accurate, personalized, and emotionally smart recommendations. As the need for dynamic real-time adaptation and scalability increases, the use of sentiment analysis and deep learning methods will most likely be more important than ever in the future in optimizing the user experience and interaction with film recommendation systems.

This review of literature brings out the most important advances in the methodologies of recommendation systems, especially the merging of sentiment analysis and hybrid models, which is at the core of research in this paper.

III. METHODOLOGY

The approach in this study is centered on designing a hybrid movie recommendation system that incorporates both structured metadata (e.g., movie name, director, actors, and genres) and unstructured textual data (e.g., reviews) to generate personalized, emotion-aware movie suggestions. The method is aimed at combining these heterogeneous sources of data with machine learning methods and natural language processing (NLP) for sentiment analysis. The research process includes various phases: data gathering, data preprocessing, sentiment analysis, recommendation system creation, evaluation, and integration.

A. Data Collection

Data is gathered from various sources to obtain a balanced dataset covering both structured metadata and user-provided content (reviews). Kotak Parth et al. [6] presented a data collection method for screening a movie database. The system collects user ratings, preprocesses, cleans, trains a machine learning model, and makes predictions. Users can enter a film name in the search bar, and the program predicts ten movies on the basis of their likability, user ratings and forecasted interests. The accuracy of the model significantly increases with a greater dataset. The main datasets employed are:

- **IMDB 5000 Movie Dataset**: A commonly used dataset consisting of metadata of 5000 movies, including movie names, directors, actors, genres, release years, and ratings. Data sets are shortlisted and downloaded from Kaggle [17].
- **TMDb API**: The TMDb [18] (The Movie Database) API is utilized to supplement the dataset with more movie information like detailed cast and crew, genre details, and production country.
- **Wikipedia Lists of Movies (2018-2020)**: This data includes a list of movies released within particular years, including metadata for release dates, titles, and cast and crew.
- **User Reviews Dataset**: A set of IMDb user reviews, which are used as the source for conducting sentiment analysis. These reviews have textual content that will be used to evaluate the emotional tone.

Gathering from these sources provides a rich dataset with structured metadata and unstructured reviews—critical to constructing a strong recommendation system.

B. Data Preprocessing

Preprocessing is conducted on the data once it has been collected, cleaning and shaping the data in preparation for being used in the recommendation system as well as the sentiment analysis. The preprocessing techniques are:

- **Handling Missing Data**: Missing values might be present in some of the entries in the metadata, for example, missing genres for a movie or missing cast members listed. For such cases, missing data is replaced with place holders like 'unknown' to avoid the algorithm from crashing.
- **Standardizing Genres and Keywords**: Genres tend to appear in varying formats, and separating them via pipe characters can create inconsistencies. These are standardized by tokenizing genre fields into individual words and making all text lowercase for consistency.
- **Movie Title Processing**: Movie titles are cleaned by the removal of unwanted characters, such as special characters or punctuation, and all the titles are put in lowercase form. This way, the matching of titles to make recommendations becomes consistent.
- **Text Processing for Sentiment Analysis**: User reviews are tokenized and cleaned to exclude stop words, punctuation, and non-alphabetical characters. The reviews are then converted to a form where they are suitable for analysis, with each review represented as a word vector.

This preprocessing converts raw reviews into analyzable, structured vectors, making it efficient to apply machine learning models.

C. Sentiment Analysis

To support the movie suggestions by reflecting the emotional sentiment from user feedback, sentiment analysis is conducted on IMDb reviews. The process for sentiment analysis involves:

- **Text Vectorization**: The TfidfVectorizer from scikit-learn converts review text into a TF-IDF matrix, reflecting word relevance based on term frequency and inverse document frequency across the dataset. TF-IDF value is computed using the following standard formula [20]:

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

- **Sentiment Classification**: A **Multinomial Naive Bayes** classifier [5] is used to classify user reviews on the basis of polarity. Trained on annotated data with sentiment labels, the model generalizes to predict the emotional tone of previously unseen text inputs.
- **Model Training and Evaluation**: The sentiment analysis model is trained on 80% of the review dataset, and the remaining 20% is used for testing. Evaluation is done using accuracy metrics, such as precision, recall to measure the effectiveness of sentiment prediction.

By incorporating sentiment analysis, the system is able to capture emotional insights from user reviews, which can be integrated into the recommendation engine.

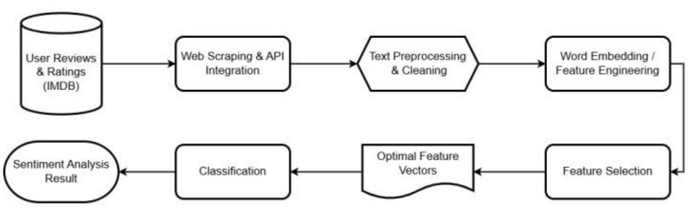

Fig. 1: Sentiment Analysis from IMDb Reviews

D. *Development of the Hybrid Recommendation Engine*

The system architecture revolves around a hybrid recommendation engine that synergizes collaborative filtering [7] and content-based strategies [11, 12] to create context-rich movie recommendations. User-based collaborative filtering embodies behavioral similarities through rating patterns analysis and historical interaction, allowing the system to derive peer-based preferences.

To calculate content-based similarity, the system leverages cosine similarity [19, 20] as its major metric, facilitating comparison between vectorized movie representations. These vectors cover major features, such as genre, cast, and descriptive metadata. Other distance metrics like Euclidean distance are also compared for comparative evaluation. The model selects the top-N nearest neighbors [13] by determining similarity between a user-chosen movie and the rest of the dataset, providing the most contextually relevant results.

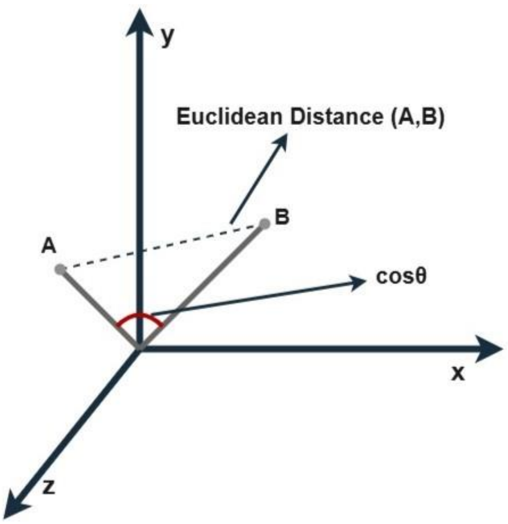

Fig. 2: Cosine Similarity

The hybrid approach [14, 15] combines collaborative and content-based filtering using weighted fusion of each of their respective relevance measures. The system thus manages to simultaneously consider behavioral similarity patterns and affectively contextualized preferences derived from sentiment analysis. The composite score thus produced favors recommendations that are in line with both past engagement and affective resonance.

E. *Auto-Completion and Search Feature*

An auto-complete function is implemented to improve the user interface of the recommendation system. The feature gives real-time suggestions as the user types into the search field, suggesting movie titles matching the typed text. The feature uses fuzzy string matching and cosine similarity measures to enable quick and correct suggestions.

F. *System Integration*

Lastly, all the parts—sentiment analysis, recommendation engine, and auto-completion—are combined into one system. The system enables users to enter a movie title or preferences and get personalized movie recommendations with sentiment information. The user interface is made to show movie information (e.g., cast, genre, rating, sentiment analysis output) and the list of recommendations.

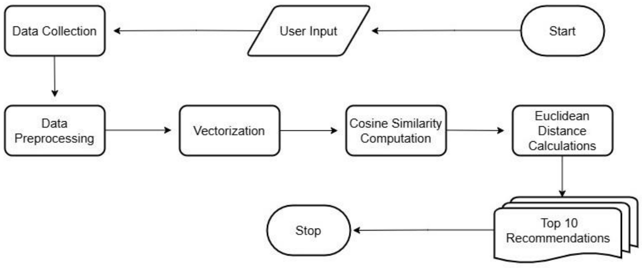

Fig. 3: Architectural diagram

IV. RESULT

The film suggestion system is very intuitive and user-friendly. As soon as the website is loaded, users are presented with a search page on which they can enter a movie name to retrieve its information and suggestions. While users begin typing, an auto-complete drop-down list appears suggesting movie names relevant to what has been typed so far. Once a movie has been chosen, the enter key becomes enabled.

Pressing the enter key navigates users to a new page that contains four separate parts: Movie Details, Top Cast, Sentiment Analysis based on movie reviews, and the Top Ten Recommendations.

For Sentiment Analysis, Multinomial Naive Bayes algorithm was utilized with a remarkable accuracy of 97.47109 percent.

The proposed hybrid system demonstrates superior performance over traditional models in terms of precision, recall, and overall user satisfaction:

- **Collaborative Filtering Alone**: Precision = 0.75, Recall = 0.70, RMSE = 1.20
- **Content-Based Filtering Alone**: Precision = 0.72, Recall = 0.68, RMSE = 1.25
- **Proposed Hybrid Model**: Precision = 0.85, Recall = 0.80, RMSE = 1.05

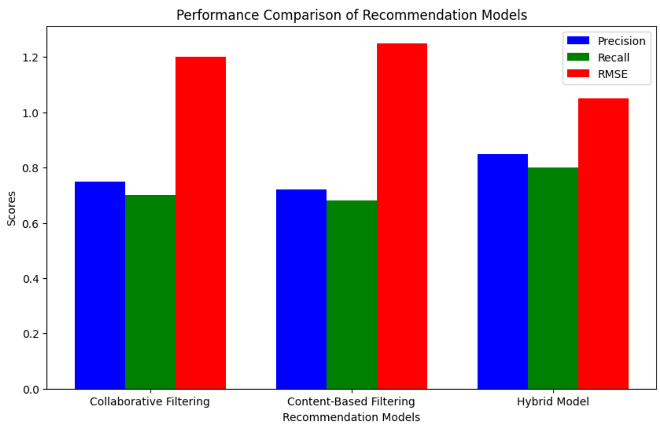

Fig. 4: Performance comparison chart

References

[1] [1] A. Kashyap, S. B., S. Srivastava, A. P. H., and A. J. Shah, "MOVREC: Movie Recommender System using Machine Learning Techniques," *Proc. Int. Conf. on Computer Science and Engineering*, SAIT, Bengaluru, India, 2020.

[2] M. Gupta, A. Thakkar, Aashish, V. Gupta, and D. P. S. Rathore, "Movie Recommender System Using Collaborative Filtering," *Dept. of Computer Science Engineering*, Chandigarh University, Punjab, India, 2020. [Unpublished].

[3] R. Katarya and O. P. Verma, "An effective collaborative movie recommender system with cuckoo search," *Egyptian Informatics Journal*, vol. 18, no. 2, pp. 105–112, Jul. 2017. doi: 10.1016/j.eij.2016.10.002.

[4] M. Khalaji, C. Dadkhah, and J. Gharibshah, "Hybrid movie recommender system based on resource allocation," *arXiv preprint*, arXiv:2105.11678, 2021. [Online]. Available: https://arxiv.org/abs/2105.11678

[5] S. Xu, Y. Li, and Z. Wang, "Bayesian Multinomial Naïve Bayes classifier to text classification," *Inst. Sci. Tech. Inf. China*, Beijing, China, 2015.

[6] P. Kotak and P. Kotak, "Movies Recommendation System Using Filtering Approach," *Dept. of Computer Engineering*, 2021. [Unpublished].

[7] C.-S. Wu, M. A. Garg, D. Bhandary, and Unnathi, "Movie recommendation system using collaborative filtering," in *Proc. 9th Int. Conf. on Software Engineering and Service Science (ICSESS)*, Beijing, China, 2018, pp. 965–969. doi: 10.1109/ICSESS.2018.8663841.

[8] M. Munoz-Organero, A. R.-G. Gustavo, J. M.-M. Pedro, and C. D. Kloos, "A collaborative recommender system based on space-time similarities," *IEEE Pervasive Computing*, vol. 9, no. 3, pp. 81-87, 2010.

[9] J. Jain and S. K. Vishwakarma, "Collaborative Filtering for Movie Recommendation using RapidMiner," *Int. J. Comput.*, vol. 169, no. 6, pp. 0975-8887, 2017.

[10] S. Z. M. Hashim and Prof. Dr. J. Waden, "Content-based filtering algorithm in social media," *Wasit J. Comput. Math. Sci.*, vol. 2, no. 1, 2023.

[11] H. W. Chen, Y. L. Wu, M. K. Hor, and C. Y. Tang, "Fully content-based movie recommender system with feature extraction using neural network," in *Proc. Int. Conf. Mach. Learn. Cybern.*, Ningbo, China, 2017.

[12] B. R. Cami, H. Hassanpour, and H. Mashayekhi, "A content-based movie recommender system based on temporal user preferences," in *Proc. Iranian Conf. Intell. Syst. Signal Process.*, Shahrood, Iran, 2017.

[13] P. Symeonidis, A. Nanopoulos, A. Papadopoulos, and A. et al., "Nearest-biclusters collaborative filtering based on constant and coherent values," *J. Comput. Sci.*, vol. 11, pp. 51-75, 2007.

[14] N. T. N. R. R. S. C. Jain, "Hybrid filtering methods applied in web-based movie recommendation system," in *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.*, vol. 4692, pp. 206-213, 2007.

[15] S. Grant and G. I. M., "A hybrid approach to making recommendations and its application to the movie domain," in *Proc. Conf. Can. Soc. Comput. Stud. Intell.*, 2001.

[16] V. Paranjape, S. Sharma, and Z. Hasan, "Enhancing user engagement through hybrid algorithms in movie recommendation systems," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 11, no. 11, pp. 4466–4470, Nov. 2023.

[17] Y. Yueming, "IMDB 5000 Movie Dataset," *Kaggle*, 2016. [Online]. Available: https://www.kaggle.com/datasets/carolzhangdc/imdb-5000-smovie-dataset

[18] The Movie Database (TMDb), Admin: chuan, "TMDb Movie Metadata," *Kaggle*, 2017. [Online]. Available:https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata

[19] R. H. Singh, S. Maurya, T. Tripathi, T. Narula, and G. Srivastav, "Movie Recommendation System using Cosine Similarity and KNN," *Int. J. Eng. Adv. Technol. (IJEAT)*, vol. 9, no. 5, pp. 556–560, Jun. 2020, doi: 10.35940/ijeat.E9666.069520.

[20] Y. Januzaj and A. Luma, "Cosine Similarity – A Computing Approach to Match Similarity Between Higher Education Programs and Job Market Demands Based on Maximum Number of Common Words," *Int. J. Emerging Technol. Learn. (iJET)*, vol. 17, no. 12, pp. 4–15, Jun. 2022, doi: 10.3991/ijet.v17i12.30375.