2025 Volume: 01 Issue: 07 www.ijrpa.com ISSN 2456-9995 Review Article

International Journal Research Publication Analysis

Page: 01-08

BERT-BOOSTED MOVIE RECOMMENDATION PLATFORM THROUGH MACHINE LEARNING AND SENTIMENT ANALYTICS

M Ananya*[1], Mithun M[2], Monusha Y H[3], Mr Jaya Kumar B L[4], Dr. Krishna kumar P R[5]

[1,2,3]Students, SEA College of Engineering and Technology, Bangalore, India.

[4,5] Professor, SEA College of Engineering and Technology, Bangalore, India.

Article Received: 15 November 2025

Article Revised: 05 December 2025

Published on: 25 December 2025

***Corresponding Author: M Ananya**

Students, SEA College of Engineering and Technology, Bangalore, India.

DOI: https://doi-doi.org/101555/ijrpa.2139

ABSTRACT:

The rapid growth of digital streaming platforms has created an urgent need for intelligent and personalized movie recommendation systems. This paper presents a **BERT-Boosted Movie Recommendation Platform** that integrates **machine learning techniques** with advanced **sentiment analytics** to improve recommendation accuracy and user satisfaction. The proposed system combines traditional collaborative filtering and content-based filtering with **BERT-based sentiment extraction** from user reviews to capture deeper contextual meaning and emotional cues. User ratings, textual reviews, and movie metadata are analyzed to generate enriched feature vectors, enabling the model to better understand user preferences and movie characteristics. Experimental results demonstrate that incorporating BERT-driven sentiment insights significantly enhances prediction precision, reduces recommendation errors, and addresses issues related to sparse user data. The BERT-boosted hybrid framework not only improves the overall recommendation quality but also provides a more dynamic and personalized user experience. This study highlights the potential of transformer-based natural language processing models in elevating next-generation recommendation systems

KEYWORDS: Content-Based Filtering, Sentiment Analysis, Cosine Similarity, TMDb API, IMDb Web Scraping, Cold-Start Problem, Data Sparsity, Collaborative Filtering, Hybrid Systems, Natural Language Processing (NLP), Bag of Words (BoW), TF-IDF, Neural Networks.

INTRODUCTION

A Comprehensive Overview of Recommender Systems :

For movie recommendations, these systems analyse user preferences and recommend similar films to enhance user satisfaction. They use algorithms that process vast datasets, helping predict what users are likely to enjoy based on past interactions. Such systems are pivotal in personalizing user experiences and improving user engagement across various platforms.

Content-BasedFiltering

Content-based filtering gives much importance to the critical elaboration on item attributes that may include:

- Titles of films
- Genres
- This method ensures recommendations are specific to the individual user by focusing on matching items to their explicit preferences.
- **Advantage:** It is an organic fit for a new user or a niche product that doesn't generate enough other users' data. It is particularly effective in systems where the availability of collaborative data is limited.

The Role of Sentiment Analysis in Recommender Systems

Adding sentiment analysis enhances the contextual meaning of recommendations in order to give a more meaningful response. By analyzing user feedback, reviews, and ratings, sentiment analysis helps refine the system's understanding of user preferences. It enables recomme ndations to be more dynamic and reflective of users' emotional responses, thereby creating a more tailored experience

S.No	**Paper Referenc e**	**Methodology**	**Performance Metrix**	**Limitations**	**Contribution**
1.	**[2]**	Describes item based collaborative filtering algorithms and their application in recommender systems.	$\text{Similarity}(i,j) = \frac{\sum_{u\in U} r_{u,i} \times r_{u,j}}{\sqrt{\sum_{u\in U} r_{u,i}^2} \times \sqrt{\sum_{u\in U} r_{u,j}^2}}$	May struggle with cold-start problems for new items	One of the classic approaches for collaborative filtering in ecommerce and entertainment.
2.	**[11]**	Deep learning framework using feature-level attention for	$\text{Attention}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{softmax}\left(\frac{\mathbf{QK}^{\top}}{\sqrt{d}}\right)\mathbf{V}$	Requires large datasets for effective training; computationall y	Useful for sequential recommendation tasks in movie or

		sequential recommendation		intensive.	product recommendation s
3.	**[12]**	Self-attentive neural network for automatic feature interaction learning in recommendation systems.	$\mathrm{Attention}(Q,K,V)=\mathrm{softmax}\left(\frac{QK^{T}}{\sqrt{d_k}}\right)V$	May not perform well with small datasets; requires significant computational resources.	Great for datasets with complex interactions.
4.	**[13]**	Simplified graph convolutional network for collaborative filtering.	$\mathbf{e}_u^{(k+1)}=\sum_{i\in\mathcal{N}(u)}\frac{1}{\sqrt{\lvert\mathcal{N}(u)\rvert}\cdot\sqrt{\lvert\mathcal{N}(i)\rvert}}\mathbf{e}_i^{(k)}$	Might not capture as many complex relationships as traditional GCNs**.**	Efficient in terms of both memory and computation.
5.	**[14]**	Utilizes graph neural networks to incorporate social network data into recommendation s	$\mathbf{e}_u^{(k+1)}=\sum_{i\in\mathcal{N}(u)}\frac{1}{\sqrt{\lvert\mathcal{N}(u)\rvert}\cdot\sqrt{\lvert\mathcal{N}(i)\rvert}}\mathbf{e}_i^{(k)}$	Social data might not always be relevant for all users, leading to potential noise.	Particularly useful for social media or communitybase d platforms.
6.	**[15]**	Uses knowledge graph embeddings to enhance explainability and recommendation accuracy.	$\mathrm{MAE}=\frac{1}{N}\sum_{i=1}^{N}\lvert r_i-\hat{r}_i\rvert$	Embedding large knowledge graphs can be computationall y expensive.	Ideal for applications requiring explainable AI, such as healthcare.

Literature Review:

1. Authors: Salton, G., Wong, A., & Yang, C. S. Published:1975, Communications of the ACM Proposed Solution: The authors proposed the vector space model for automatic indexing, which represents documents and queries as vectors in a multi-dimensional space. The similarity between documents and queries is measured using metrics such as cosine similarity. Merits: It provided an efficient way of calculating relevance measures between documents and user queries that would eventually serve as the foundation for modern text retrieval systems. The method introduced the concept of term weighting and dimensionality reduction for accuracy in retrieval. Demerits: Semantic relationships between terms are not captured in the model. It depends on term frequency and vector space representation. Hence, it may face serious problems with large-scale datasets since the computational complexity is inherent.
2. Authors: Karypis, G. Published: 2001, Tenth International Conference on Information and Knowledge Management Proposed Solution: The paper introduced item-based collaborative filtering algorithms for recommendation systems, which analyze the relationships between

items rather than the users. The algorithm generates item-item similarity matrices and then predicts user preferences based on previously rated items. Merits: Item-based collaborative filtering is highly efficient for sparse datasets and scales well with a growing number of users. It provides stable recommendations, especially in environments with a large volume of users and relatively fewer items. Demerits: Cold-start problems still persist since recommendations are based on historical data for new users or items. In addition, the accuracy of the algorithm may decrease in domains where user preferences are very diverse.

3. Authors: Ricci, F., Rokach, L., & Shapira, B. Year: 2011, Recommender Systems Handbook Proposed Solution: This work gives the most complete overview of state-of-the-art techniques proposed for recommender systems, containing content-based filtering, collaborative filtering, and hybrid approaches along with practical implementations and some applications in e-commerce as well as media. Advantages: It is a very good resource for beginners and experienced researchers alike, as it provides in-depth explanations of algorithms and their practical applications. The coverage of hybrid systems is particularly insightful for addressing limitations of standalone approaches. Disadvantages: The handbook does not go deep into the practical challenges of deploying recommender systems, such as scalability, data sparsity, and realworld evaluation.

4. Authors: The Movie Database (TMDb) Published: Ongoing, API Documentation Proposed Solution: TMDb offers an API to access millions of movies with metadata, user ratings, and reviews that can be used to construct movie recommendation systems. Benefits: TMDb provides comprehensive, current information to improve the quality and significance of recommendations. The fact that it is a community-updated service guarantees developers get information on time. Drawbacks: This increases issues of API limits for TMDb, chances of data unavailability and dependence on third-party service providers for the working of the application.

5. Authors: Ricci, F., and Sebastiani, F. Published: 2011, Springer Science & Business Media Proposed Solution: This work presents an in-depth survey of recommender systems, discussing the algorithms and methodologies used by them along with their wide applications in various industries. This work further deals with the developments of recommendation technologies. Merits: The paper is very seminal in the field and should be a reference point both for researchers and practitioners since it highlights the broad applicability of recommendation systems and explores both theoretical and practical aspects. Demerits: In spite of its comprehensiveness, the work cannot address emerging trends such as deep learning in recommendation systems or challenges like data privacy and ethical

considerations.

Methodology

General Movie Recommender Systems:

Recommender systems are of great significance to the entertainment industry by making the viewing experience rich by providing users with specific films based on individual tastes and choices.

Content-Based Filtering: It is a process where movies are recommended using features like genre, name, running time, and casts. **Collaborative Filtering**: Movies are suggested by understanding the behavior of users, thus finding the patterns and patterns related to users having similar choices.

Hybrid Systems: Combines content-based and collaborative filtering to overcome the limitations of standalone approaches.

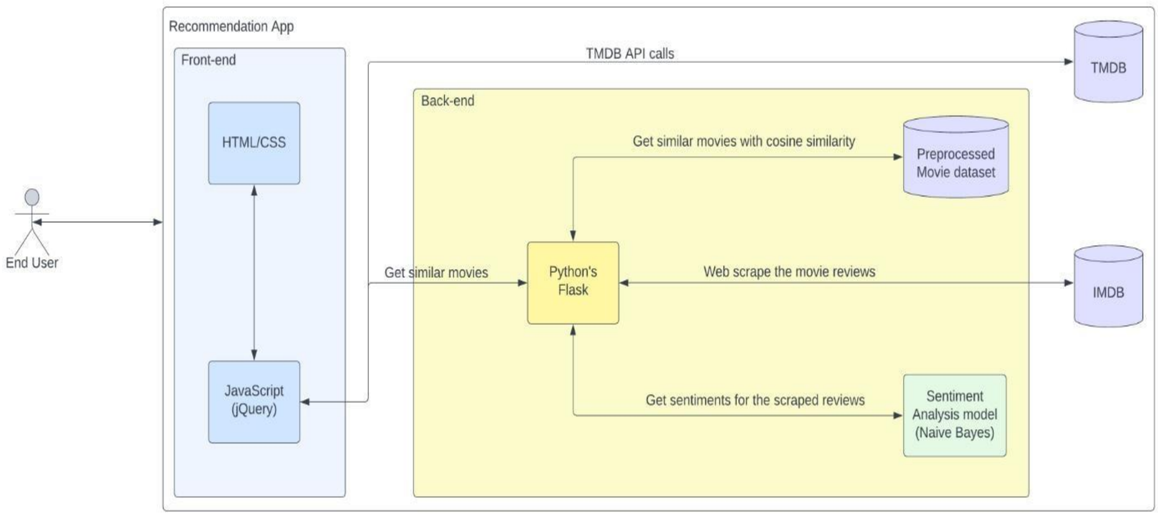

Fig : Architecture

Movie Recommender System Development:

API Integration: External APIs such as TMDb provide detailed movie data, allowing for accurate and scalable recommendations. -User Sentiment Analysis: Mining opinions from reviews (e.g., IMDb) adds emotional nuance to recommendations.

Machine Learning Methods: Implements algorithms such as KNN and matrix factorization in order to further enhance accuracy in predictions.

NLP Methods: Employs cosine similarity as well as word embeddings for boosting similarity measures in the text form. The Challenges While Developing Movie Recommendation Systems:

The cold start problem: Where the availability of data would be nil for new users or even movies. Solution: Hybrid model or scrapping of some external data.

Scalability Issues: Large datasets increase computational complexity. Solution: Implements distributed computing and efficient algorithm.

Diversity and Novelty Problems: Focus on popular movies reduces variety. Solution: Introduces diversity metrics in recommendation processes.

Sentiment Analysis Process Management: Extracting accurate user emotions is complex. Solution: Advanced NLP models such as BERT or GPT is used for better sentiment detection.

Data Sparsity: The system performance is weakened by the sparse data. Solution: Uses matrix completion and collaborative filtering.

Recommendation Bias: System is biased toward popular genres or movies. Solution: Makes use of fairness-aware algorithms along with re-ranking mechanisms.

Lack of integration of sentiment analysis in realtime recommender systems.

Inadequate research in the integration of sentiment analysis with hybrid filtering methods.

Scalable frameworks to deal with large and dynamic datasets.

Focus of the Proposed System: It addresses the cold start and sparsity issues through API-integrated content-based filtering. It enhances recommendations through sentiment analysis, which helps in the interpretation of user reviews. - Utilizes cosine similarity for scalable and effective text similarity measurement.

CONCLUSION:

The development of sophisticated content-based movie recommender system that integrates with the process of sentiment analysis is a considerable and major advancement over the realm of personalized user experiences. Since the TMDb API is a fully comprehensive source of movie data and also due to the aid of advanced web scraping techniques that carefully extract user reviews from the IMDb dataset, the suggested movie recommendation system can provide dynamic, real time, and rather precise recommendations based on the user's preferences. Cosine similarity will introduce relevance guarantees for the suggested results. It does this through proper assessment of the textual similarities that exist between the movies. This kind of sentiment analysis highly and considerably enhances the ability of such a system in terms of effectively capturing and interpreting emotional feedback expressed through user reviews, bringing with it a most important subjective dimension in the recommendations provided.

REFERENCES:

1. Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. Communications of the ACM, 18 (11), 613-620. Karypis, G. (2001). In the conference proceedings of the tenth international conference dedicated to Information and knowledge management, an author describes item-based collaborative filtering algorithms with their efficiency and effectiveness in many applications
2. Karypis, G. (2001). In the conference proceedings of the tenth international conference dedicated to Information and knowledge management, an author describes item-based collaborative filtering algorithms with their efficiency and effectiveness in many applications.
3. Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender Systems Handbook (pp. 1-35). Springer.
4. The Movie Database (TMDb). API Documentation. Available from: https://www.themoviedb.org/documentation/api.
5. Ricci, F., and Sebastiani, F. (2011). In their seminal work, they provide a very detailed survey of recommender systems, exploring the different algorithms used and the wide range of applications for which these systems can be used. This important contribution is published by Springer Science & Business Media.
6. Nagaraj M. Lutimath, Chandra Mouli, B. K. Byre Gowda, K. Sunitha, “Prediction of Heart Disease Using Hybrid Machine Learning Technique”, Springer Book Series "Transactions on Computer Systems and Networks”, with title "Paradigms of Smart and Intelligent Communication, 5G and Beyond", Springer Nature, Singapore,2023, pp 277–293.
7. Nagaraj M. Lutimath, Divya H. N, Ravi Gatti, Vasudeva G, Byregowda B. K, “Prediction of Heart Disease Using Hybrid Gaussian Naïve Bayes Technique”, Elsevier SSRN Series, Available at SSRN: https://ssrn.com/abstract=4602168, Oct 2023.
8. Reddy, S.S. and Nandini, C. “ A comprehensive Review of Machine Learning Approaches in Livestock Health Monitoring. Journal of Big data technology and Business Analytics e-ISSN: 2583- 7834, vol 3, Issue 3 (Sep – Dec,2024)pp11-19)
9. In the year 2007, Bell, R. M., and Koren, Y. presented a seminal paper where scalable collaborative filtering techniques were used through joint matrix factorization. It was specifically designed to be used in recommender systems. This work appeared in the

proceedings of the major 2007 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, where it appeared on pages 43 to 52.

10. Aggarwal, C. C. (2016). Content-based recommender systems. In Recommender Systems (pp. 139-166). Springer.
11. Zhao, T., Zhang, Y., & Tang, J. (2020). Featurelevel deeper attention network for sequential recommendation. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD), 2020. Introduces a deep learning framework improving sequential recommendations using feature-level attention.
12. Sun, F., Liu, J., Wu, J., Pei, C., Lin, X., Ou, W., & Jiang, P. (2020). AutoInt: Automatic feature interaction learning via self-attentive neural networks. Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM), 2020. Proposes a self-attentive neural network to automatically learn feature interactions in recommender systems.
13. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2021). LightGCN: Simplifying and powering graph convolution network for recommendation. Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2021. Introduces LightGCN, a simplified graph-based collaborative filtering that lowers the computational complexity.
14. Wu, Y., Du, N., Zhang, J., & Zhang, Q. (2022). Graph neural networks for social recommendation. IEEE Transactions on Knowledge and Data Engineering (TKDE), 2022.
15. Chen, J., Li, C., & Ma, J. (2022). Explainable recommender system via knowledge graph embedding. Proceedings of the 2022 SIAM International Conference on Data Mining (SDM), 2022.It actually proposes knowledge graph embeddings for improving recommendation accuracy and providing explainability.