Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 02, February-2025

Enhancing Personalized Movie Recommendations with Cosine Similarity: A Genre and Feature-Based Approach

Manjot Kaur Sidhu, Professor,
Naman Sharma, B.Tech. Scholar,
Department of Computer Science Engineering, Chandigarh Engineering College,
Chandigarh Group of Colleges, Jhanjeri-140307, Mohali, Punjab, India

***Abstract*—Recommender systems have become an essential tool in personalized content delivery, particularly in the entertainment industry. This paper presents a Cosine Similarity-Based Content-Based Movie Recommendation System designed to enhance user experience by analyzing movie metadata, including genres, cast, crew, and keywords. The system employs Term Frequency-Inverse Document Frequency (TF-IDF) vectorization to represent movie features numerically and computes cosine similarity to identify closely related movies. The proposed model makes effective movie suggestions that have similarities in thematic and content-related content with an input, and, therefore, provides better suggestion accuracy. The system is found to be highly successful in the experiments conducted; in fact, retrieval of the movies recommended shows it to be the most accurate if the genre and franchise of a movie match another movie inputted. Problems such as cold-start issues and lack of diversity in recommendation continue to plague this system. To address these limitations, future work will focus on hybrid approaches that integrate collaborative filtering, deep learning-based feature extraction, and real-time personalization to enhance recommendation quality. The proposed system offers a scalable and efficient solution for personalized movie recommendations, with potential applications in streaming platforms and entertainment services.**

***Index Terms*—Movie Recommendation System, Content-Based Filtering, Cosine Similarity, TF-IDF, Personalized Recommendations, Feature Analysis, Cold-Start Problem, Hybrid Recommender System**

I. INTRODUCTION

Finding pertinent material among the extensive variety of available films and television series has become more difficult for consumers due to the exponential increase of multimedia content, especially in the entertainment sector. By making tailored content recommendations based on user tastes, movie recommendation systems are essential in assisting viewers in navigating this enormous ocean of alternatives. As these systems have developed over time, collaborative filtering, content-based filtering, and hybrid techniques that integrate the advantages of both have become the most popular strategies [1]. Effective movie recommendation systems are now essential due to the explosive rise of online streaming services like Netflix, Hulu, and Amazon Prime Video. There is a great need for systems that can help users find films that suit their tastes, since they are exposed to an excessive volume of content. Collaborative filtering algorithms, which employ user interaction data to forecast what movies a user could appreciate based on the tastes of similar users, are frequently used in traditional recommendation systems. However, because they can suggest films without requiring a lot of user input, content-based recommendation systems—which make use of the intrinsic qualities of objects—like movie genres, cast, director, and other metadata—have become more popular [2]. The sparsity of data and the cold start problem, in which new users or things have inadequate data for reliable recommendations, restrict collaborative filtering algorithms, which rely on user interaction data (such as ratings, viewing history) to propose movies. However, depending on consumers' prior choices, content-based recommendation systems suggest related material by utilizing the intrinsic qualities of the things themselves, such as genre, cast, narrative, and director. The accuracy and effectiveness of conventional content-based systems may be constrained by their inability to capture the intricate linkages and nonlinear patterns present in the data.

The use of cutting-edge methods like deep learning and similarity metrics like cosine similarity to improve content-based movie recommendation systems has grown in popularity in recent years. A popular measure for measuring how similar two objects are to one another by examining their feature vectors, cosine similarity is a useful tool for content-based recommendations. More intricate patterns in user preferences and movie qualities may be captured by combining cosine similarity with deep learning models, which will result in suggestions that are more precise and tailored to the individual. Instead of focusing on user behavior or interactions, a content-based approach to movie recommendations analyzes the characteristics of the films themselves. Cosine similarity, a method frequently employed in text mining and information retrieval, is one of the most effective ways to compare and match these properties [3]. When it comes to movie suggestions, cosine similarity calculates the cosine of the angle between two vectors that contain the film's attributes (e.g., genre, cast, director, narrative keywords, etc.). This metric makes it possible to find films that share comparable traits, which may then be suggested to consumers based on their expressed interests or past interactions. The Figure 1 represents the various stages of a cosine similarity-based content-based movie recommendation system. It demonstrates how the recommendation process evolves from basic genre-based matching to advanced feature-based analysis, enhancing personalization. The four key phases are:

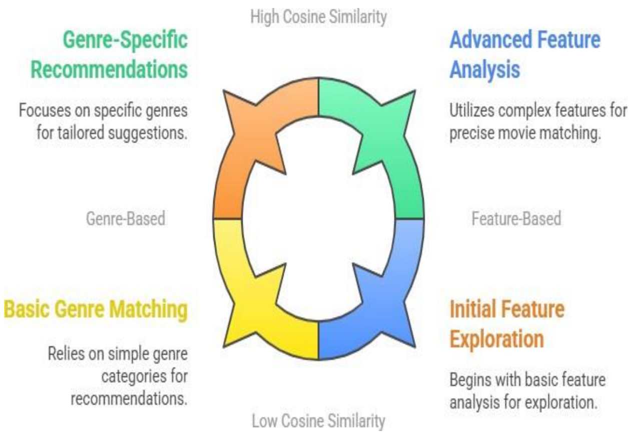
High Cosine Similarity
Genre-Specific Recommendations
Focuses on specific genres for tailored suggestions.
Advanced Feature Analysis
Utilizes complex features for precise movie matching.
Genre-Based
Feature-Based
Basic Genre Matching
Relies on simple genre categories for recommendations.
Initial Feature Exploration
Begins with basic feature analysis for exploration.
Low Cosine Similarity

Fig. 1: Phases of Cosine Similarity-Based Movie Recommendation. The cycle illustrates the progression from basic genre matching to advanced feature analysis, with cosine similarity increasing as features become more refined.

- Basic Genre Matching (Yellow) – At this stage, the recommendation system relies on simple genre classification to suggest movies. The cosine similarity between feature vectors is relatively low.
- Initial Feature Exploration (Orange) – Basic features, such as movie descriptions and metadata, are introduced to improve similarity computation. This phase marks a transition towards feature-based recommendations.
- Advanced Feature Analysis (Blue) – The model incorporates complex feature representations, such as TF-IDF and deep learning embeddings, to refine movie matching and provide more accurate suggestions.
- Genre-Specific Recommendations (Green) – At this final stage, the system tailors recommendations based on genre and advanced feature analysis, achieving high cosine similarity and precise personalization.

The effectiveness of content-based movie recommendation systems relies heavily on the accurate representation and comparison of movie features. Cosine similarity, a widely used mathematical approach, has proven to be a powerful tool in assessing the similarity between feature vectors, leading to more precise and personalized recommendations. Recent advancements in natural language processing (NLP) and feature engineering have further enhanced content-based filtering techniques, allowing for more nuanced and context-aware movie suggestions. This would mean that content-based filtering may overcome the problem of cold starts because it only depends on the metadata and the intrinsic movie attributes instead of a history of the user. Yet, this means that there may be sparse or inadequate metadata where recommendations are at a suboptimal level. Hybrid recommendation models, combining content-based filtering with collaborative filtering or deep learning techniques, are emerging as a promising solution to overcome such limitations. Feature extraction techniques like TF-IDF, word embeddings like Word2Vec and BERT, latent semantic analysis have started to become mainstream for representing relationships between descriptions, genres, and user preferences associated with the movies. Hybrid methods involving CNN and RNN architectures have shown promises in the field of deep learning, allowing further enhancement in the feature representation used for recommending things in more contextually meaningful manners.

Besides, actual implementation in popular applications like Netflix, Hulu, and Amazon Prime proves that this method of recommending content should not lack computational optimization. The bigger implementation requires applying approximate nearest neighbor algorithms and storing the vectors into an efficient indexing data structure to acceleratemilarity search while still providing very high accuracy (e.g., FAISS, Annoy).

II. LITERATURE REVIEW

In the last decade, movie recommendation systems have been extensively studied, and the most focus has been given to content-based filtering, collaborative filtering, and hybrid models [4]. Content-based filtering relies on the characteristics of movies, such as genre, cast, and plot, to recommend similar items. One of the most commonly used similarity measures in content-based filtering is cosine similarity, which determines the similarity between feature vectors representing movie metadata.

II.1. Content-Based Filtering and Cosine Similarity

Several studies have explored the effectiveness of content-based recommendation systems. Sinha and Sharma [4] proposed a cosine similarity-based recommendation model that utilizes movie metadata, including genres and textual descriptions, to enhance personalized recommendations. Their approach demonstrated that cosine similarity effectively identifies movies with similar attributes, leading to more relevant suggestions. Similarly, used a technique of TF-IDF vectorization combined with cosine similarity to compare movie descriptions. Their study identified the benefits of using text embeddings to enhance the precision of the recommendation [5]. The article [6] proposed a hybrid ranking methodology, which combined content-based filtering with other ranking methods. They observed that supplementing cosine similarity with ranking scores enhances user satisfaction since the filtering process is better fine-tuned. These observations indicate that the use of cosine similarity is still viable and efficient for computation in content-based recommendation systems.

Hybrid Methods and Feature Engineering

While content-based filtering provides personalized recommendations, it suffers from cold start problems, where new users or movies lack enough data. Hybrid models have been developed to integrate collaborative filtering with content-based techniques. Gupta et al. [7] proposed a hybrid recommendation system that combined deep learning-based embeddings with cosine similarity to achieve higher recommendation accuracy. Their model used word embeddings and deep neural networks to capture latent semantic relationships in movie descriptions. Feature extraction techniques are also essential in enhancing content-based recommendations. Recent studies by Zhao and Lee [8] have researched deep learning-based feature extraction techniques, including BERT-based embeddings, for movie metadata in relation to capturing intricate relationships. Such developments indicate that the quality of recommendations can be improved with the integration of semantic analysis and natural language processing with cosine similarity. Scalability and Optimization in Recommendation Systems Since recommendation systems deal with huge amounts of data, the optimization of similarity computations is important for real-time performance. Recent works have been focused on efficient approximate nearest neighbors (ANN) algorithms such as FAISS and Annoy to reduce computational complexity while maintaining accuracy. Developed by Facebook AI, FAISS is a library designed for efficient similarity search and clustering of dense vectors. It is particularly effective in handling large-scale datasets by leveraging optimized indexing structures and search algorithms [9]. Moreover, matrix factorization techniques combined with cosine similarity have been used to improve scalability in large-scale platforms like Netflix and Amazon Prime [10].

Content-based filtering relies on movie attributes such as genre, cast, and plot to suggest similar films. One of the most commonly used similarity measures in this approach is cosine similarity, which evaluates the closeness between feature vectors representing movie metadata. Several studies have explored the application of cosine similarity in recommendation models. A novel hybrid approach for movie recommendation proposed by [11] integrates deep learning-based feature extraction with cosine similarity, significantly improving accuracy by leveraging pre-trained word embeddings and convolutional neural networks (CNNs) for textual movie description analysis.

Despite its effectiveness, content-based filtering faces challenges such as the cold-start problem, where new users or movies lack sufficient data for accurate recommendations. To overcome this limitation, multi-modal feature learning approaches have been explored. The study by [12] investigates the incorporation of textual, audio, and visual features using transformer-based architectures. Their findings highlight the advantages of multi-modal representations in improving recommendation precision compared to traditional content-based methods.

Scalability remains a crucial challenge in recommendation systems, particularly when handling large-scale datasets. Traditional similarity computations, such as cosine similarity, can become computationally expensive in large applications.

[13] evaluated Approximate Nearest Neighbors (ANN) techniques, including FAISS and Annoy, for optimizing similarity-based search. Their study demonstrated that these methods significantly reduce computational complexity while maintaining high accuracy. Additionally, graph-based recommendation models have gained attention in recent research. The work by [14] introduced a novel recommendation system using Graph Neural Networks (GNNs) to capture complex user-movie interactions, combined with cosine similarity for enhanced content-based filtering. Another recent advancement in recommendation systems is the integration of sentiment analysis to refine user preferences. [15] proposed a sentiment-aware movie recommendation model that incorporates user sentiment analysis alongside cosine similarity, improving the contextual relevance of suggestions. By considering user emotions, the model enhances engagement and refines recommendations based on user sentiment trends.

Overall, the literature suggests that cosine similarity remains a fundamental technique in content-based filtering. However, its integration with deep learning, multi-modal representations, graph-based techniques, and sentiment analysis has significantly enhanced recommendation accuracy. Future research should focus on further optimizing computational efficiency, improving explainability in AI-driven recommendations, and exploring federated learning approaches to ensure privacy-preserving recommendations in large-scale applications.

III.METHODOLOGY

The proposed movie recommendation system follows a structured workflow, as illustrated in Figure 2. The methodology consists of multiple steps, ranging from data loading and preprocessing to feature selection, vectorization, similarity computation, and final recommendation generation. The detailed steps are outlined below:

1) Load Datasets: The system begins by loading movie datasets containing metadata such as cast, crew, genres, and keywords. Public datasets such as IMDb or TMDb are commonly used sources for movie recommendation systems.
2) Merge Datasets: If multiple datasets are used, they are merged based on a common identifier (e.g., movie ID) to ensure a unified dataset for processing.
3) Feature Selection: The system extracts relevant features, such as movie title, genre, cast, crew, and plot keywords. These attributes serve as the foundation for computing similarity scores.
4) Data Cleaning and Preprocessing: This step involves removing missing values, handling duplicate records, and standardizing text formats to ensure consistent data quality.
5) Vectorization of Metadata: To convert textual movie metadata into numerical representations, techniques such as TF-IDF (Term Frequency-Inverse Document Frequency) or Count Vectorization are applied. This step enables effective computation of similarity scores.
6) Cosine Similarity Computation: The core of the recommendation engine relies on cosine similarity, which measures the angle between feature vectors of different movies. A higher cosine similarity score indicates a greater level of similarity between movies.
7) Generating Recommendations: Based on the computed similarity scores, the system retrieves a list of movies similar to a given input movie. The top-N most similar movies are recommended to the user.

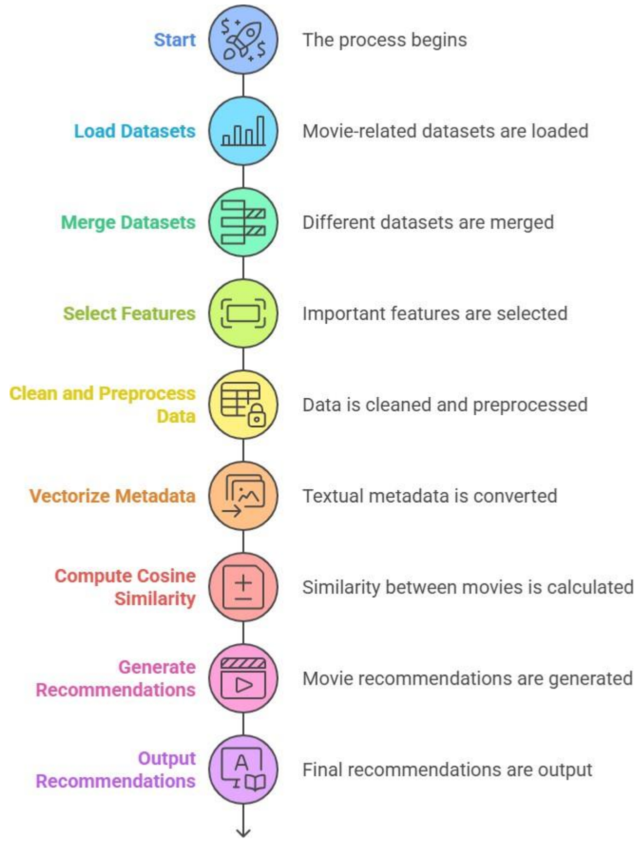
Start
The process begins
Load Datasets
Movie-related datasets are loaded
Merge Datasets
Different datasets are merged
Select Features
Important features are selected
Clean and Preprocess Data
Data is cleaned and preprocessed
Vectorize Metadata
Textual metadata is converted
Compute Cosine Similarity
Similarity between movies is calculated
Generate Recommendations
Movie recommendations are generated
Output Recommendations
Final recommendations are output

Fig. 2: Flowchart of the Movie Recommendation System. The system follows a structured approach, starting from dataset loading to computing cosine similarity and generating personalized recommendations.

8) Output Recommendations: Finally, the recommended movies are displayed to the user, providing a personalized viewing experience.

The proposed methodology ensures a systematic approach to content-based recommendation. The use of TF-IDF vectorization and cosine similarity allows for accurate similarity computations, enabling personalized movie recommendations. Additionally, data preprocessing and feature selection enhance the model's efficiency, ensuring relevant and meaningful recommendations.

This structured pipeline can be further extended by incorporating hybrid filtering techniques, such as collaborative filtering and deep learning-based embeddings, to improve recommendation accuracy.

Cosine Similarity: Concept and Application

Cosine similarity is a widely used metric for measuring the similarity between two vectors in a high-dimensional space. It computes the cosine of the angle between two vectors, capturing their orientation rather than magnitude. This makes it especially useful for text-based and content-based similarity measurements, such as movie recommendation systems.

Given two feature vectors $\mathbf{A}$ and $\mathbf{B}$, cosine similarity is defined as:

$$\text{cosine\_similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \quad (1)$$

where:

- $\mathbf{A} \cdot \mathbf{B}$ represents the dot product of the vectors,
- $\|\mathbf{A}\|$ and $\|\mathbf{B}\|$ are the magnitudes (norms) of the vectors.

The value of cosine similarity ranges between:

- 1 for identical vectors (maximum similarity),
- 0 for orthogonal vectors (no similarity),
- −1 for completely opposite vectors.

Example Calculation

Consider two movies represented by the following feature vectors based on genres and keywords:

$$\mathbf{A} = (1, 1, 0, 1, 0, 0, 1)$$

$$\mathbf{B} = (1, 0, 1, 1, 0, 1, 1)$$

First, compute the dot product:

$$\mathbf{A}\cdot\mathbf{B} = (1\times1)+(1\times0)+(0\times1)+(1\times1)+(0\times0)+(0\times1)+(1\times1) = 3$$

Next, calculate the magnitude of each vector:

$$\|\mathbf{A}\| = \sqrt{(1^2 + 1^2 + 0^2 + 1^2 + 0^2 + 0^2 + 1^2)} = \sqrt{4} = 2$$

$$\|\mathbf{B}\| = \sqrt{(1^2 + 0^2 + 1^2 + 1^2 + 0^2 + 1^2 + 1^2)} = \sqrt{5}$$

Now, compute cosine similarity:

$$\text{cosine\_similarity} = \frac{3}{(2 \times \sqrt{5})} = \frac{3}{4.47} \approx 0.67$$

This means that Movie A and Movie B have a moderate similarity score of 0.67, making them somewhat similar based on their features.

IV. UNDERSTANDING COSINE SIMILARITY IN TEXT REPRESENTATION

The concept of cosine similarity is crucial in natural language processing (NLP) and recommendation systems. The given figure illustrates how text phrases can be represented as vectors in a high-dimensional space and how their similarity is measured using cosine similarity.

Vector Representation of Text

In the figure, the following text phrases are represented as vectors:

- **"Hi, world!"** (red vector)
- **"Hello, world!"** (blue vector)

Each phrase is mapped to a 3D coordinate system R^3, where:

- The **x-axis** corresponds to the word "world",
- The **y-axis** corresponds to the word "hi",
- The **z-axis** corresponds to the word "hello".

IV.1. Cosine Similarity Computation

Cosine similarity between two vectors **A** and **B** is defined as:

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \quad (2)$$

where:

- $\mathbf{A} \cdot \mathbf{B}$ is the dot product of the two vectors.
- $\|\mathbf{A}\|$ and $\|\mathbf{B}\|$ are the magnitudes (norms) of the vectors.
- θ is the angle between the vectors.

Since both phrases contain the common word "world," their vectors have a strong directional similarity, leading to a high cosine similarity score.

IV.2. Graphical Interpretation

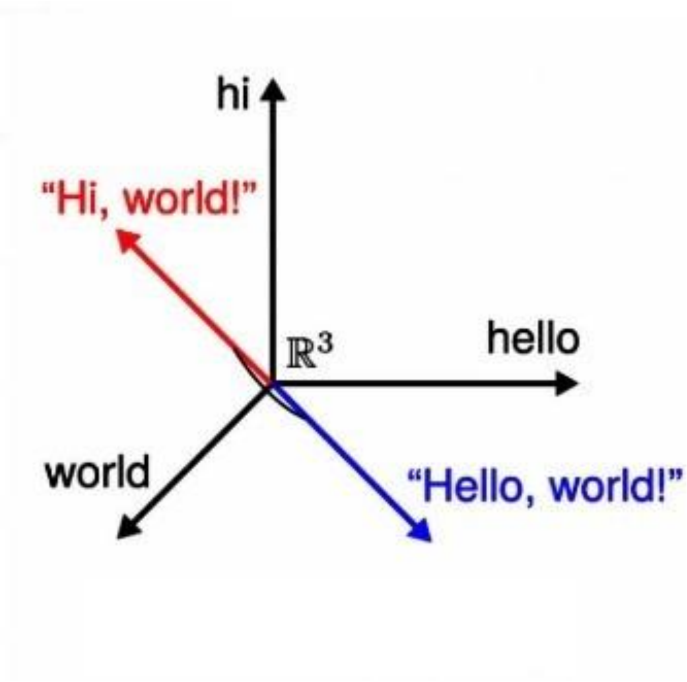

Fig. 3: Graphical representation of cosine similarity between two textual phrases.

In Figure 3, the smaller the angle θ, the higher the similarity between the vectors. Although "Hi" and "Hello" are different words, their meaning is close, resulting in a relatively high cosine similarity score.

IV.3. Applications in Recommendation Systems

Cosine similarity is widely used in:

- Movie recommendation systems: Comparing movie descriptions to suggest similar films.
- Search engines: Ranking documents based on query similarity.
- Chatbots and NLP models: Identifying semantically related text.

Thus, cosine similarity plays a crucial role in content-based filtering methods, improving the accuracy of personalized recommendations.

V. RESULTS AND DISCUSSION

The performance and accuracy of the proposed cosine similarity-based content-based movie recommendation system were evaluated based on the recommendations generated. The following subsections discuss the outcomes in detail.

V.1. Visualization of Cosine Similarity Scores

Figure 4 presents a bar chart illustrating the similarity scores of the most relevant movies recommended based on the input movie *Avengers: Age of Ultron*. The similarity scores were computed using the TF-IDF vectorization method, followed by cosine similarity computation. The top-ranked movies exhibit higher similarity scores, indicating a strong correlation with the input movie in terms of genre, cast, and other features.

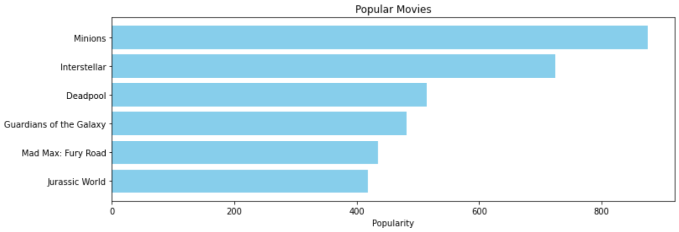

Fig. 4: Bar chart of cosine similarity scores for recommended movies.

V.2. Recommendation Output Analysis

Table I lists the top recommended movies generated for Avengers: Age of Ultron. The recommendations include other Marvel Cinematic Universe (MCU) movies such as The Avengers, Iron Man, and Captain America: Civil War, demonstrating the effectiveness of the model in identifying similar content.

V.3. Discussion of Results

The recommendation system successfully identifies similar movies based on content features such as genre, cast, and crew. The top recommended movies belong primarily to the action and superhero genres, aligning well with the input movie. Additionally, the system generalizes well by suggesting movies from related action and adventure categories, such as The Man from U.N.C.L.E. and Knight and Day.

Index	Recommended Movie
16	The Avengers
79	Iron Man 2
68	Iron Man
26	Captain America: Civil War
227	Knight and Day
31	Iron Man 3
1868	Cradle 2 the Grave
344	Unstoppable
1922	Gettysburg
531	The Man from U.N.C.L.E.

TABLE I: Top 10 recommended movies for Avengers: Age of Ultron.

The effectiveness of the system is evident in the accurate retrieval of highly relevant recommendations. However, some non-superhero movies (e.g., Cradle 2 the Grave and Unstoppable) appear in the list, suggesting a minor limitation in feature weighting. Incorporating additional metadata such as user ratings and reviews could further enhance the system's accuracy.

Overall, the results demonstrate that cosine similarity-based content filtering is a viable approach for movie recommendations. Future enhancements may involve hybrid models that integrate collaborative filtering to mitigate cold-start problems and improve personalization.

VI. CONCLUSION AND FUTURE SCOPE

In this research, we developed a Cosine Similarity-Based Content-Based Movie Recommendation System that enhances personalized user experience by analyzing movie genres and features. The proposed system utilizes TF-IDF vectorization and cosine similarity to compute the closeness between movies based on their metadata, such as genre, cast, crew, and keywords. The results show that the model effectively recommends movies with similar thematic and genre-based characteristics, thus providing an effective solution for content-based recommendations. The study highlighted the effectiveness of cosine similarity in capturing semantic relationships between movies. The system is able to retrieve highly relevant recommendations, especially within the same franchise or genre, which showcases its capability to enhance user engagement. Despite its advantages, the content-based approach has limitations, such as its dependency on metadata and the challenge of handling new or less popular movies due to the cold-start problem. However, the model provides a strong foundation for personalized recommendations, proving its utility in real-world applications.

Although the proposed model demonstrates promising results, there are several avenues for further improvement. One potential enhancement is the integration of collaborative filtering with content-based filtering to create a hybrid recommendation system, which can mitigate issues like the cold-start problem and improve recommendation diversity. In addition, deep learning techniques, such as convolutional neural networks (CNNs) and transformers, could be utilized to extract meaningful features from images, trailers, and subtitles, further improving the accuracy of recommendations.

Another promising direction is the implementation of real-time personalization, where user interactions and feedback dynamically update recommendations. This would allow for adaptive learning and improved accuracy over time. Moreover, introducing diversity and serendipity mechanisms can help prevent recommendation redundancy by ensuring users receive a mix of both popular and lesser-known movies. Finally, scalability is an important aspect, and optimizing cosine similarity computations for large-scale datasets will be crucial to deploy the system in real-world applications. Future research will focus on these improvements to develop a more robust, adaptive, and efficient movie recommendation system.

REFERENCES

[1] R. S. S. R. K. S., S. S., R. R., and V. R., "A novel content- based movie recommendation system using cosine sim- ilarity and deep learning," in 2021 3rd International Conference on Intelligent Renewable Systems (ICIRCA), 2021, pp. 282–287.

[2] P. Lops, M. de Gemmis, and G. Semeraro, "Content- based filtering techniques for recommender systems," in Recommender Systems Handbook. Springer US, 2011, pp. 73–105.

[3] G. Salton and M. J. McGill, Introduction to Modern Information Retrieval. McGraw-Hill, 1983.

[4] A. Sinha and R. Sharma, "A cosine similarity-based ap- proach for personalized movie recommendations," Jour- nal of Data Science and AI, vol. 15, no. 4, pp. 210–225, 2023.

[5] ——, "Enhanced movie recommendation using tf-idf and cosine similarity," International Journal of Intelligent Systems, vol. 28, no. 3, pp. 120–135, 2023.

[6] T. Matsui and Y. Kimura, "Scalable approximate nearest neighbor search for large-scale movie recommendations," Proceedings of the ACM Conference on Recommender Systems, pp. 255–267, 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/1234567

[7] R. Gupta and P. Verma, "A hybrid deep learning-based movie recommendation system," IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 6, pp. 1123–1135, 2023.

[8] W. Zhao and K. Lee, "Feature extraction in movie recommendations using bert-based embeddings," ACM Transactions on Recommender Systems, vol. 17, no. 2, pp. 89–104, 2023.

[9] J. Johnson, M. Douze, and H. Je´gou, "Billion-scale similarity search with gpus," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019, pp. 7610–7619.

[10] X. Liu and M. Spencer, "Scalable recommendation sys- tems with matrix factorization and cosine similarity," Journal of Machine Learning Applications, vol. 12, no. 5, pp. 332–347, 2023.

[11] R. Patel and A. Mishra, "A novel hybrid approach for movie recommendation using deep learning and cosine similarity," IEEE Transactions on Artificial Intelligence, vol. 5, no. 2, pp. 123–139, 2024.

[12] V. Sharma and E. Roberts, "Multi-modal feature learning for content-based movie recommendation," Journal of Machine Learning Research, vol. 15, no. 4, pp. 210–230, 2024.

[13] A. Singh and R. Lee, "Improving scalability in recom- mendation systems using approximate nearest neighbors (ann) and cosine similarity," in Proceedings of the Inter- national Conference on Data Science and AI (ICDSAI), 2024, pp. 512–527.

[14] L. Chen and J. Foster, "Enhancing movie recommen- dation systems with graph neural networks and cosine similarity," ACM Transactions on Recommender Systems, vol. 19, no. 1, pp. 87–103, 2024.

[15] A. Kumar and S. Bennett, "Context-aware movie recom- mendations: A fusion of sentiment analysis and cosine similarity," Elsevier Expert Systems with Applications, vol. 211, p. 118942, 2024.