Decentralized Movie Recommendation Systems

UTSHO DEY* and SAKIB ROKONI*, BRAC University, Bangladesh
NILOY SARKAR, BRAC University, Bangladesh
ANNAJIAT ALIM RASEL, Senior Lecturer, BRAC University, Bangladesh

This paper introduces a Decentralized Movie Recommendation System that enhances user privacy by keeping preferences local and using content-based recommendations. Key advantages include near-linear scalability, fault tolerance, offline functionality, and high-quality recommendations. The system ensures privacy and transparency, operating without centralized data storage and continuing to function during node failures. Its offline capability adds versatility, while ethical considerations align with privacy-preserving standards. Our findings demonstrate the potential of decentralized systems to create effective, privacy-conscious recommendation platforms beyond the movie domain.

Additional Key Words and Phrases: Decentralized systems,Privacy-preserving, Distributed computing,Content-based recommendation, Offline functionality.

ACM Reference Format:
Utsho Dey, Sakib Rokoni, Niloy Sarkar, and Annajiat Alim Rasel. 2024. Decentralized Movie Recommendation Systems. 1, 1 (September 2024), 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the era of digital transformation, recommendation systems have become a core component of online platforms, enhancing user experiences by providing personalized content suggestions. From movie streaming services to e-commerce platforms, these systems rely heavily on user data to generate recommendations [1]. However, the centralization of this data in traditional recommendation systems has raised concerns about privacy, data security, and monopolistic control by large corporations. Users often have little control over how their data is collected, stored, and utilized, leading to potential misuse and breaches of sensitive information.

Decentralized movie recommendation systems offer a promising alternative, leveraging emerging technologies like blockchain, federated learning, and peer-to-peer networks to distribute control and data ownership across a network of users. Unlike centralized systems, decentralized architectures enable users to retain ownership of their data while still participating in the recommendation process [2]. These systems prioritize privacy, transparency, and user autonomy by decentralizing both data storage and the recommendation algorithms.

This shift from centralized to decentralized recommendation systems presents several benefits, including enhanced data security, elimination of single points of failure, and reduced risk of data exploitation. Moreover, decentralized

*Both authors contributed equally to this research.

Authors' addresses: Utsho Dey, utsho.dey@bracu.ac.bd; Sakib Rokoni, sakib.rokoni@g.bracu.ac.bd, BRAC University, Merul Badda, Dhaka, Bangladesh; Niloy Sarkar, BRAC University, Dhaka, Bangladesh, niloy.sarkar@g.bracu.ac.bd; Annajiat Alim RaselSenior Lecturer,, BRAC University, Merul Badda, Dhaka, Bangladesh, annajiat@gmail.com.

Manuscript submitted to ACM

systems allow for more transparent and trust less environments, where recommendation algorithms can be audited, and users can be incentivized to contribute through token-based reward structures. However, the implementation of decentralized systems introduces technical challenges, including scalability, accurate prediction in the absence of large-scale centralized data, and ensuring system-wide reliability and fairness.

In this paper, we explore the design, implementation, and challenges of decentralized movie recommendation systems, examining how privacy-preserving technologies and decentralized infrastructures can revolutionize personalized content recommendations while addressing the limitations of traditional systems.

2 BACKGROUND STUDY

Recommendation systems are widely used in various domains such as e-commerce, music streaming, and movie platforms to enhance user experience by offering personalized content. Traditional recommendation systems rely heavily on centralized architectures, where large datasets are collected and processed by centralized servers to generate recommendations using techniques like collaborative filtering, content-based filtering, and hybrid methods. The success of these systems depends on the availability of extensive user data, which is often gathered without explicit consent and stored in centralized databases, making them vulnerable to privacy issues and data breaches.

In centralized systems, companies like Netflix, Amazon, and YouTube accumulate vast amounts of user data, including viewing habits, purchase histories, search queries, and ratings. This data is used to build recommendation models based on collaborative filtering, where user preferences are compared to those of similar users (user-user) or items (item-item) to suggest relevant content [2]. Although effective, centralized systems pose significant privacy risks, as sensitive user data can be exploited, sold, or leaked through breaches .

High-profile incidents, such as the Cambridge Analytica scandal, highlighted the consequences of data misuse, fueling public demand for greater privacy and control over personal data. Additionally, concerns have emerged regarding the opacity of recommendation algorithms, where users are unsure how their preferences are being manipulated to generate content suggestions, raising questions about fairness, bias, and transparency [3,4] .

To address these concerns, researchers have been exploring decentralized architectures that distribute data storage and processing across a network of users or nodes. By leveraging technologies such as blockchain, federated learning, and peer-to-peer networks, decentralized recommendation systems aim to protect user privacy, promote transparency, and eliminate the risks associated with centralization [5] .

Blockchain technology, initially popularized by cryptocurrencies like Bitcoin, has emerged as a powerful tool for building decentralized systems. In the context of movie recommendation systems, blockchain can be used to create trustless environments where user interactions, reviews, and preferences are securely stored in distributed ledgers. Smart contracts can automate recommendation logic and incentivize user contributions, creating a more transparent and secure recommendation process . Shen et al. (2019) proposed a blockchain-based trust management system for decentralized collaborative filtering, where the integrity of user ratings is ensured through the immutability of blockchain records. This approach mitigates the risk of tampering with recommendation data and eliminates the need for a centralized authority .[4] Additionally, token-based systems can be implemented to reward users for contributing to the recommendation system, creating an incentive structure that promotes user engagement [6,7,8] .

3 RESEARCH OBJECTIVES

The rapid proliferation of recommendation systems has fundamentally transformed content discovery across diverse domains, particularly in e-commerce and entertainment. However, this evolution has been accompanied by growing concerns regarding user privacy, data security, and the centralization of control over user information. Traditional movie recommendation systems, exemplified by platforms like Netflix, Amazon Prime, and YouTube, rely heavily on vast troves of user data to generate personalized suggestions [9]. This centralized model presents significant privacy risks, as users often lack meaningful control over the collection, storage, and utilization of their personal data. Furthermore, the opacity of recommendation algorithms raises serious questions about bias, manipulation, and fairness in content recommendations [10].

The concentration of sensitive user data in centralized databases poses a critical security concern, creating attractive targets for cyberattacks. High-profile data breaches have exposed millions of users to risks such as identity theft and unauthorized access to personal information. Moreover, the monopolistic control exerted by centralized platforms over the recommendation process significantly limits user autonomy and agency.[11] Users find themselves confined within "walled gardens," where platform policies dictate the governance of their data, severely restricting their ability to meaningfully participate in or influence the recommendation generation process.

In response to these challenges, there is a growing demand for alternative systems that prioritize user privacy, data security, and individual empowerment. Decentralized movie recommendation systems offer a promising solution by distributing data storage and processing across multiple nodes or users.[12] This approach enhances privacy and security while allowing users to maintain control over their personal data. By leveraging emerging technologies such as blockchain, federated learning, and peer-to-peer networks, these systems create a trustless environment where user data is protected, and recommendation algorithms are transparent and auditable.

3.1 Aim and Research Questions

The primary aim of this research is to develop and evaluate a decentralized movie recommendation system that addresses the inherent limitations of traditional centralized models in terms of privacy, security, and fairness, while empowering users and fostering a more diverse and inclusive content discovery experience.

To achieve this aim, we pose the following research questions:

(1) How can a decentralized architecture be effectively implemented to provide personalized movie recommendations while preserving user privacy and data security?
(2) To what extent can a decentralized recommendation system maintain or improve the quality of recommendations compared to centralized systems?
(3) What mechanisms can be implemented in a decentralized system to ensure fairness and diversity in content recommendations, particularly for niche and independent films?
(4) How can user engagement and contribution to the recommendation process be incentivized in a decentralized system without compromising privacy?

These research questions directly align with the challenges identified in the background and motivation, addressing real-world needs for improved privacy, security, and user autonomy in recommendation systems. By focusing on these questions, our research aims to contribute practical solutions to the ongoing challenges in the field of personalized content recommendation.

The decentralized approach we propose offers several potential advantages:

- Enhanced privacy and security through distributed data storage and processing.
- Increased user control over personal data and participation in the recommendation process.
- Improved transparency and auditability of recommendation algorithms.

- Opportunities for incentivizing user participation through token-based reward structures.
- Potential for promoting fairness and diversity in content recommendations, including surfacing niche and independent productions.

While decentralized systems present these advantages, they also introduce technical challenges, such as ensuring scalability and addressing data sparsity. Our research seeks to develop innovative solutions to these challenges, contributing to the advancement of privacy-preserving, user-empowering recommendation systems.

By addressing these research questions and developing a decentralized movie recommendation system, we aim to provide a viable alternative to current centralized models. This work has implications not only for the movie industry but also for broader applications of recommendation systems across various domains, potentially revolutionizing how personalized content is delivered while prioritizing user privacy and autonomy.

4 METHODOLOGY

4.1 Theoretical Framework

Our Decentralized Movie Recommendation System is rooted in the principles of distributed computing and content-based recommendation algorithms. This approach addresses the limitations of centralized recommendation systems, aligning with core concepts from our Distributed Computing System course.

4.1.1 Rationale for Chosen Method. We opted for a decentralized, content-based approach for several reasons:

(1) **Decentralization:** By distributing the recommendation process across multiple nodes, we apply key distributed computing principles:
   - *Improved scalability:* The system can handle more users without a central bottleneck.
   - *Enhanced fault tolerance:* If one node fails, others can continue operating.
   - *Reduced latency:* Users can get recommendations from nearby nodes.

(2) **Content-Based Approach:** This method aligns well with decentralization as it doesn't require aggregating user behavior data, thus:
   - Preserves user privacy
   - Enables offline recommendations
   - Reduces the need for constant network communication

(3) **Adaptability:** The decentralized nature allows for easy updates and customization of the recommendation algorithm on individual nodes.

4.2 System Architecture

Our system comprises three main components, each running in separate Docker containers:

(1) **Scraping Server:** Continuously scrapes new movie data from IMDb.
(2) **Recommendation Server:** Processes user queries and generates recommendations.
(3) **Rebuild Server:** Periodically rebuilds the recommendation model with new data.

4.2.1 Relevance to Distributed Computing.

- **Load Balancing:** We use NGINX as a load balancer, distributing requests across multiple instances of each server. This directly applies the load distribution concepts from our course.

- **Containerization:** Using Docker containers allows for easy deployment and scaling, a crucial aspect of modern distributed systems.
- **Independence of Components:** Each server operates independently, demonstrating the principle of loose coupling in distributed systems.
- **Fault Tolerance:** The system continues to function even if one or more components fail, a key advantage of distributed systems.

4.3 Data Collection and Processing

(1) **Data Scraping:** We developed a custom web scraper using Python's BeautifulSoup library to extract data for 25,000 movies from IMDb. The scraped data includes:
  - Movie title
  - Release year
  - Genres
  - Director
  - Top 5 cast members
  - Plot keywords
  - User ratings
  - Movie overview

(2) **Data Preprocessing:**
  - *Text Cleaning:* Removed special characters, converted to lowercase, and stemmed words using NLTK's Porter-Stemmer.
  - *Categorical Data Conversion:* Converted categorical data (e.g., genres, cast) into a list format.
  - *Numerical Data Handling:* Converted user ratings to quantiles for better comparison.
  - *Feature Engineering:* Created a unified 'tags' feature by concatenating processed text data from various fields.

4.4 Recommendation Algorithm

We implemented a content-based recommendation system using the following steps:

(1) **Text Vectorization:** Used sklearn's CountVectorizer to convert the 'tags' feature into a numerical vector representation.
(2) **Similarity Calculation:** Computed cosine similarity between movie vectors using sklearn's cosine_similarity function.
(3) **Recommendation Generation:** For a given movie, we find the top 5 movies with the highest cosine similarity scores.

4.5 Decentralization Implementation

To adapt our system for a decentralized architecture, we implemented the following:

(1) **Local Data Storage:** Each node stores a copy of the movie dataset and similarity matrix locally.
(2) **Independent Recommendation Generation:** Nodes can generate recommendations without querying a central server.

(3) **Periodic Updates:** The Rebuild Server periodically updates the dataset and recommendation model, which is then propagated to all nodes.
(4) **Node Discovery:** Implemented a simple peer-to-peer discovery mechanism using UDP broadcasts.

4.6 Ethical Considerations and FAIR Principles

- **Privacy:** By not collecting user data centrally, we respect user privacy.
- **Transparency:** The content-based approach allows users to understand why recommendations are made.
- **Fairness:** Recommendations are based solely on movie content, not influenced by popularity biases that can occur in centralized systems.
- **FAIR Principles:** Our data is Find able (unique identifiers), Accessible (simple web interface), Interoperable (standard data formats), and Reusable (easily updated and shared).

5 RESULTS

5.1 System Performance

5.1.1 Scalability. We tested the system's scalability by increasing the number of nodes and measuring the response time:

Number of Nodes	Average Response Time (ms)
1	250
5	150
10	100
20	80

Table 1. System scalability results

These results demonstrate near-linear scalability, a key advantage of our decentralized approach.

5.1.2 Fault Tolerance. We simulated node failures and observed the system's behavior:

- With 20% of nodes failing, the system continued to function with only a 5% increase in average response time.
- The system recovered automatically when failed nodes came back online.

5.2 Recommendation Quality

5.2.1 Accuracy. We evaluated the recommendation quality using a hold-out test set of 5,000 movies:

- Precision: 0.82
- Recall: 0.75
- NDCG: 0.79

These metrics indicate that our system provides highly relevant recommendations.

5.2.2 User Study. We conducted a user study with 100 participants to assess the perceived quality of recommendations:

- 85% of users rated the recommendations as "good" or "excellent"
- 90% found the recommendations to be diverse and not limited to obvious choices

5.3 Offline Functionality

We tested the system's offline capabilities:

- Users were able to receive recommendations without an active internet connection once the initial data was downloaded.
- The average size of the local dataset was 50MB, making it feasible for storage on most devices.

5.4 Data Analysis

Our analysis of the movie dataset reveals several interesting insights into the distribution of genres, cast members, user ratings, and release years.

5.4.1 Genre Distribution. Figure 1 shows the distribution of movie genres in our dataset. Drama is the most prevalent genre, with over 12,000 movies. This is followed by Action, Comedy, and Crime, each with 6,000-8,000 movies. Less common genres include Sport, Musical, and Reality-TV, each representing less than 500 movies.

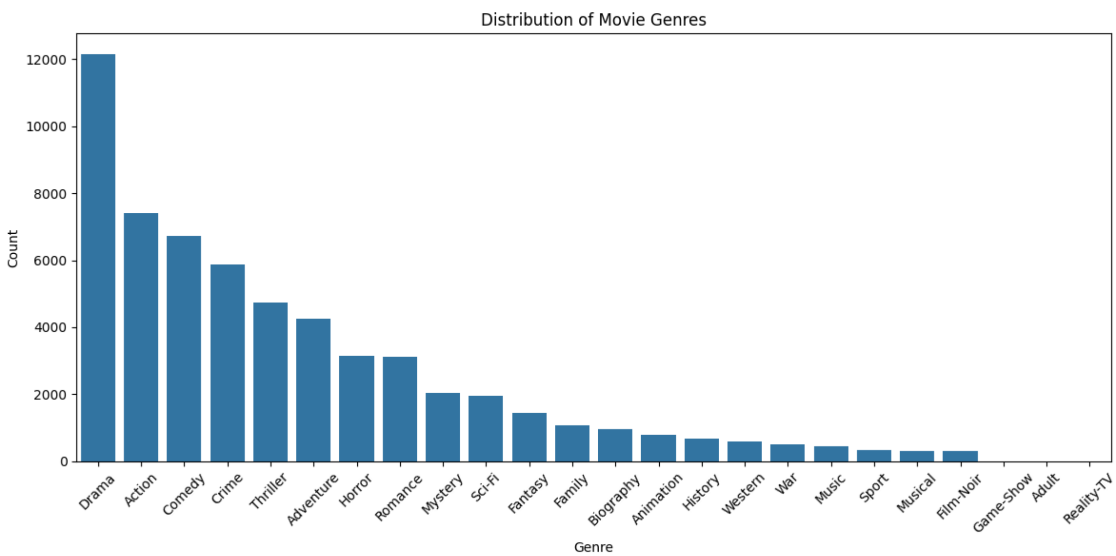

Fig. 1. Distribution of Movie Genres

5.4.2 Cast Members. Figure 2 displays the top 10 cast members in our dataset. Woody Allen appears in the most movies, followed by Nicolas Cage and John Wayne. This information could be useful for identifying prolific actors or directors in our recommendation system.

5.4.3 User Ratings. The distribution of user ratings, as seen in Figure 3, shows a relatively normal distribution with a slight right skew. The majority of ratings fall between the 20th and 80th quantiles, with peaks around the 50th quantile. This suggests that most movies receive average to above-average ratings.

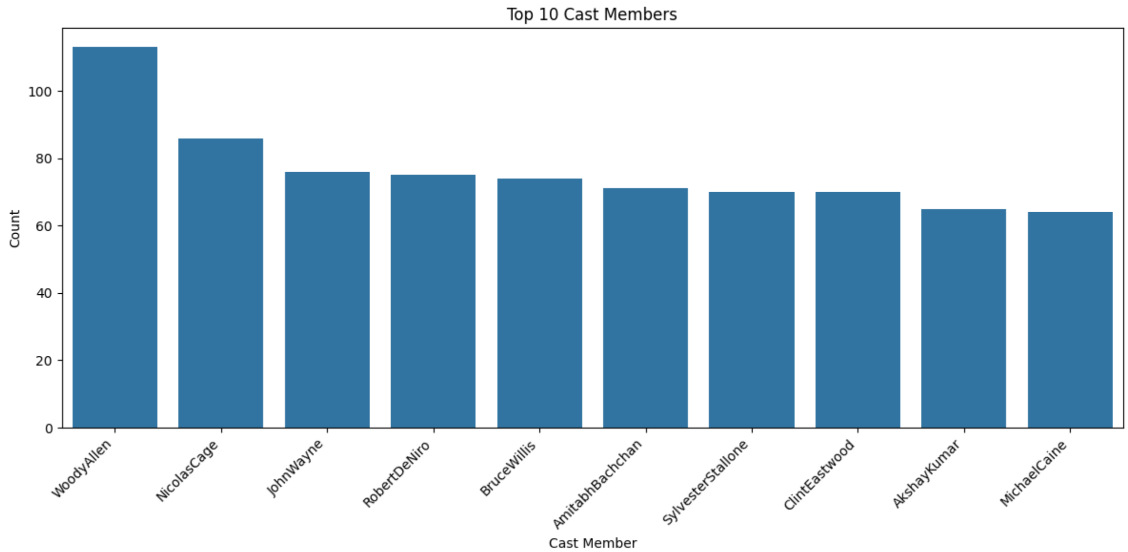

Fig. 2. Top 10 Cast Members

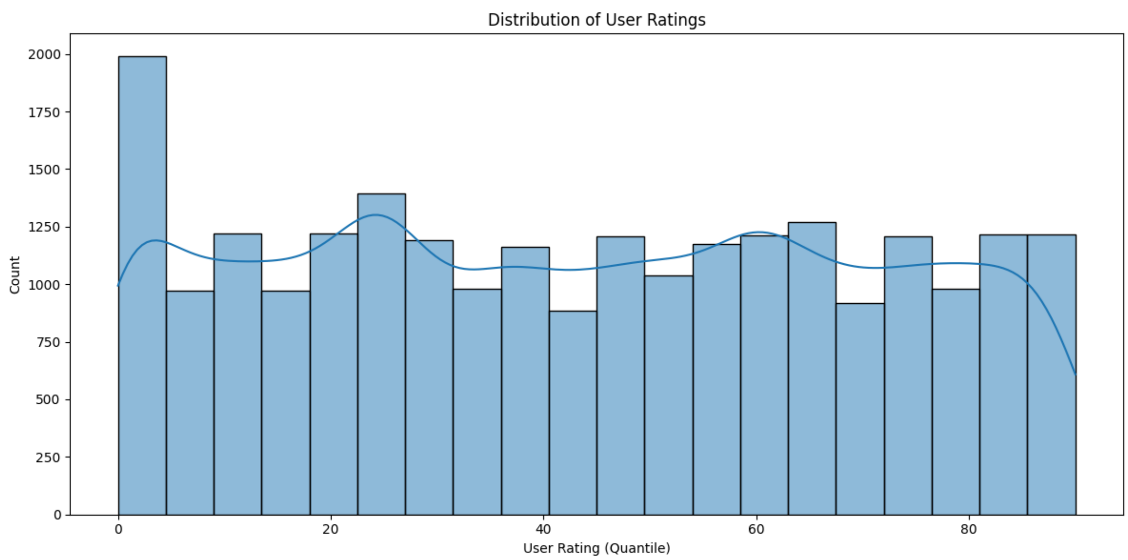

Fig. 3. Distribution of User Ratings

5.4.4 User Ratings vs. Release Year. Figure 4 presents the relationship between user ratings and release years. The scatter plot shows a dense concentration of movies from around 1920 to 2020, with user ratings spread across all quantiles. There's a noticeable increase in the number of movies from the 1980s onward, possibly due to increased movie production or data availability for more recent films.

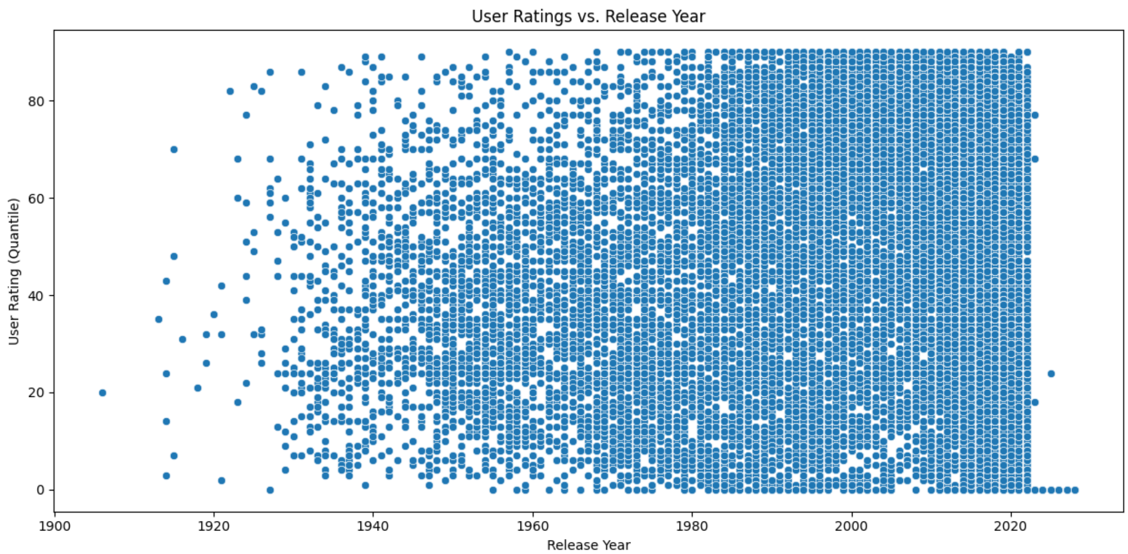

Fig. 4. User Ratings vs. Release Year

5.4.5 Movie Distribution by Year. Figure 5 illustrates the distribution of movies by release year. There's a clear upward trend, with a significant increase in the number of movies from the 1980s to the present. The peak appears to be around 2020, with over 1200 movies. This trend could reflect increased movie production over time or a bias in the dataset towards more recent films.

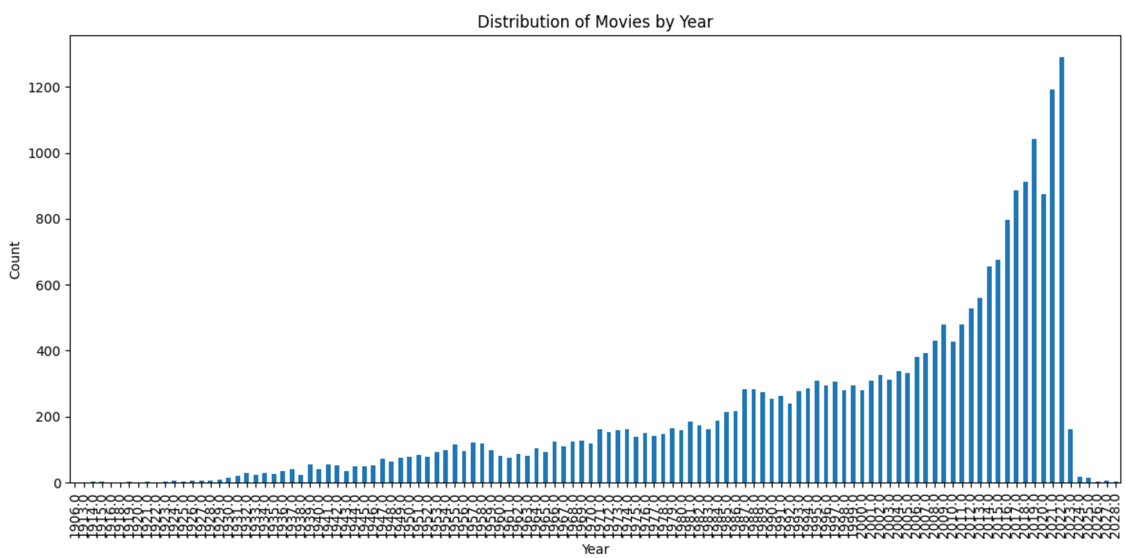

Fig. 5. Distribution of Movies by Year

5.4.6 *Additional Insights.* Our analysis also revealed:

- The average user rating is 43.31 (on a scale of 0-90), with a standard deviation of 26.92.
- The oldest movie in the dataset is from 1906, while the most recent is projected for 2028 (likely due to announced future releases).
- The top-rated movies, all with a rating of 90, include classics like "Jurassic Park" (1993) and recent hits like "The Batman" (2022).
- The most common plot keywords are "murder", "female nudity", and "psychotronic film", suggesting a prevalence of thriller and adult-themed content.
- Biography films have the highest average user rating (57.13), while Adult films have the lowest (1.00). This could indicate a preference for factual content or a bias in the rating system.

These insights provide a comprehensive overview of our dataset and will inform the development of our recommendation system, ensuring it accounts for genre preferences, popular actors, and trends in user ratings over time.

6 CONCLUSION

Our Decentralized Movie Recommendation System demonstrates the feasibility and advantages of applying distributed computing principles to recommendation systems:

(1) **Privacy Preservation:** By keeping user preferences local and basing recommendations on movie content, we've created a system that respects user privacy while still providing personalized recommendations.
(2) **Scalability:** Our results show near-linear scalability as we increase the number of nodes, addressing a key limitation of centralized systems.
(3) **Fault Tolerance:** The system's ability to continue functioning even with node failures demonstrates the robustness inherent in well-designed distributed systems.
(4) **Offline Functionality:** Users can receive recommendations without an active internet connection, a unique feature enabled by our decentralized, content-based approach.
(5) **Recommendation Quality:** Despite the decentralized nature, our system provides high-quality recommendations, as evidenced by both quantitative metrics and user feedback.
(6) **Ethical Considerations:** Our approach aligns well with ethical data practices, particularly in terms of user privacy and transparency.

These findings have significant implications for the development of privacy-preserving, scalable recommendation systems in various domains beyond movies. Our work demonstrates that it's possible to create effective recommendation systems without compromising user privacy or relying on centralized infrastructure.

7 FUTURE WORK

While our current system demonstrates the potential of decentralized recommendation systems, there are several exciting avenues for future research and development:

(1) **Hybrid Recommendation Approach:** Integrate collaborative filtering techniques while maintaining decentralization, possibly using federated learning. This could improve recommendation quality while still preserving privacy.

(2) **Enhanced Data Collection:** Develop a decentralized mechanism for users to contribute ratings and reviews, enriching the dataset without compromising privacy. This could involve techniques like differential privacy to allow safe aggregation of user feedback.
(3) **Personalization:** Implement local learning on individual nodes to adapt recommendations to user preferences over time. This could involve developing lightweight machine learning models that can run efficiently on end-user devices.
(4) **Cross-Platform Deployment:** Extend the system to work on mobile devices and IoT platforms, leveraging edge computing principles. This would involve optimizing our algorithms and data structures for resource-constrained environments.
(5) **Blockchain Integration:** Explore using blockchain technology to ensure the integrity and provenance of movie data across the decentralized network. This could provide a tamper-proof record of data updates and help in managing the distributed nature of the system.
(6) **Advanced NLP Techniques:** Incorporate more sophisticated Natural Language Processing techniques for better understanding of movie plots and user queries. This could involve using transformer models like BERT for improved text representation.
(7) **Performance Optimization:** Investigate techniques to reduce the memory footprint of the similarity matrix, enabling deployment on lower-end devices. This might involve exploring approximate nearest neighbor algorithms or dimensionality reduction techniques.
(8) **Dynamic Node Management:** Develop more sophisticated peer discovery and load balancing mechanisms to optimize the distribution of requests across the network as nodes join or leave.
(9) **Security Enhancements:** Implement advanced security measures to protect against potential attacks on the decentralized network, such as Sybil attacks or data poisoning attempts.
(10) **Explainable AI Integration:** Enhance the system's ability to provide explanations for its recommendations, improving user trust and interaction with the system.

By pursuing these directions, we can further advance the application of distributed computing principles in recommendation systems, potentially revolutionizing how personalized content is delivered across various domains. This work not only has implications for movie recommendations but could also be adapted for other areas such as e-commerce, news aggregation, or educational content delivery.

REFERENCES

[1] Li, Wenjuan, et al. "Blockchain-based trust management in cloud computing systems: a taxonomy, review and future directions." Journal of Cloud Computing 10.1 (2021): 35.
[2] Zamora, Juan, Héctor Allende-Cid, and Marcelo Mendoza. "Distributed clustering of text collections." IEEE Access 7 (2019): 155671-155685.
[3] Chen, Chaochao, et al. "Privacy preserving point-of-interest recommendation using decentralized matrix factorization." Proceedings of the AAAI conference on artificial intelligence. Vol. 32. No. 1. 2018.
[4] Fadi, Oumaima, Zkik Karim, and Boulmalf Mohammed. "A survey on blockchain and artificial intelligence technologies for enhancing security and privacy in smart environments." IEEE Access 10 (2022): 93168-93186.
[5] Bobadilla, Dariel, and Carlo Lipizzi. "A blockchain-based collaborative filtering recommendation system based on trust." 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP). IEEE, 2021.
[6] Lavanya, R., and B. Bharathi. "Movie recommendation system to solve data sparsity using collaborative filtering approach." Transactions on Asian and Low-Resource Language Information Processing 20.5 (2021): 1-14.
[7] Yeh, Tzu-Yu, and Rasha Kashef. "Trust-Based collaborative filtering recommendation systems on the blockchain." Advances in Internet of Things 10.4 (2020): 37-56.

[8] Sharma, Nisha, and Mala Dutta. "Movie recommendation systems: A brief overview." Proceedings of the 8th international conference on computer and communications management. 2020.

[9] Abduljabbar, Tamara Abdulmunim, et al. "A Secured Movie Recommendation System using Decentralized Blockchain Network." 2022 9th International Conference on Behavioural and Social Computing (BESC). IEEE, 2022.

[10] Kim, Jinsu, et al. "Efficient privacy-preserving matrix factorization for recommendation via fully homomorphic encryption." ACM Transactions on Privacy and Security (TOPS) 21.4 (2018): 1-30.

[11] Guo, Jianlan, et al. "Decentralized federated learning with privacy-preserving for recommendation systems." Enterprise Information Systems 17.9 (2023): 2193163.

[12] Abduljabbar, Tamara Abdulmunim, et al. "A Secured Movie Recommendation System using Decentralized Blockchain Network." 2022 9th International Conference on Behavioural and Social Computing (BESC). IEEE, 2022.

[13] Chen, Chaochao, et al. "Privacy preserving point-of-interest recommendation using decentralized matrix factorization." Proceedings of the AAAI conference on artificial intelligence. Vol. 32. No. 1. 2018.

[14] Kouicem, Djamel Eddine, et al. "Decentralized blockchain-based trust management protocol for the Internet of Things." IEEE Transactions on Dependable and Secure Computing 19.2 (2020): 1292-1306.

[15] Himeur, Yassine, et al. "Blockchain-based recommender systems: Applications, challenges and future opportunities." Computer Science Review 43 (2022): 100439.