University of Glasgow | School of Computing Science

Unpacking MovieLens 1M: A Study of Collaborative and Content-Based Filtering for Movie Recommendations

Ahnaf Ismat Tasin

School of Computing Science

Sir Alwyn Williams Building

University of Glasgow

G12 8RZ

A dissertation presented in partial fulfillment of the requirements of the Degree of Master of Science at the University of Glasgow

31st August 2023

Abstract

The rapid growth of digital content has made it increasingly challenging for users to find relevant items, such as movies, that align with their preferences. Recommender systems have emerged as a pivotal solution to this information overload, guiding users towards choices that best match their tastes. This thesis unpacks the intricacies of the MovieLens 1M dataset, a rich collection of movie ratings, to explore the effectiveness of two predominant recommendation techniques: Collaborative and Content-Based Filtering. Through a meticulous data analysis, this study delves into the strengths and limitations of each approach, assessing their performance metrics and offering insights into their applicability. Collaborative filtering, which harnesses the power of collective user preferences, and content-based filtering, which focuses on item attributes, are examined. The results underscore the nuances of recommendation accuracy and provide a roadmap for future systems aiming to enhance user experience. This study not only offers a comprehensive understanding of the MovieLens 1M dataset but also contributes to the broader discourse on optimizing recommendation systems in the digital age.

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic form.

Name: Ahnaf Ismat Tasin Signature: *Ahnaf Ismat Tasin*

Acknowledgments

I would take this opportunity to express my gratitude and thank my supervisor, Dr. McCaig, for his precious mentorship, continuous support, and time throughout the dissertation. I am also thankful to the School of Computing Science for giving me the opportunity to be a part of it. Finally, I'd like to thank my parents and my wife for supporting me throughout my Master's program.

Table of Contents

Chapter 1 Introduction

1.1 Motivation

In today's digital era, an exponential growth in available content has been observed, leading to an extensive array of choices for users. The challenge that emerges is discerning how users might navigate these vast options to find items that align with their unique preferences. This challenge is addressed by recommender systems, which are designed to suggest items based on intricate algorithms and diverse data inputs.

1.2 Aim

At the heart of this study lies the MovieLens 1M dataset, a rich compilation of movie ratings that provides insights into the preferences of a multitude of users. While many studies have been conducted using this dataset with the aim of constructing and refining recommender systems, the approach adopted in this research is distinct. Instead of focusing on the creation of a robust recommender system, the MovieLens 1M dataset is meticulously unpacked. The emphasis is placed on a foundational analysis, aiming to understand the nuances of user preferences and movie attributes contained within the data. The primary objective of this research is to delve deeply into the MovieLens 1M dataset, extracting patterns and insights that might inform the broader discourse on recommendation systems. Through this exploration, a contribution to the foundational understanding of this domain is sought, highlighting the significance of thorough data comprehension.

1.3 Report Structure

To facilitate a structured exploration, the report is organized as follows:

Chapter 2: Literature Review - An examination of pertinent literature, providing context for the subsequent analysis.

Chapter 3: Exploratory Data Analysis - An initial assessment of the dataset, identifying its inherent characteristics.

Chapter 4: Data Preparation - The processes undertaken to refine and ready the data for deeper analysis are detailed.

Chapter 5: Analysis of User Profiles and Movie Preferences - An in-depth exploration of user profiles and their associated preferences.

Chapter 6: Methodology - The methodologies and techniques employed within this research are outlined.

Chapter 7: Collaborative Filtering - A study of a primary recommendation technique is presented.

Chapter 8: Content-Based Filtering - Another pivotal recommendation approach, focusing on content attributes, is examined.

Chapter 9: Conclusion - The findings are synthesized, and insights for future research in the field are proposed.

Chapter 2 Literature Review

1. Introduction to Recommender Systems

Recommendation systems have emerged as a crucial tool in the digital age, addressing the challenge of information overload by providing personalized content and service suggestions to users. These systems analyze patterns of user behavior and preferences, leveraging various algorithms and techniques to predict items that a user might find interesting. The goal is to present users with recommendations that are both relevant and engaging, enhancing their online experience. Whether it's suggesting movies on a streaming platform, books on an e-commerce site, or articles on a news website, recommendation systems play a pivotal role in driving user engagement and satisfaction. Their widespread application across various domains underscores their significance in today's digital landscape (Ricci, F., Rokach, L., & Shapira, B., 2011) [1].

2. Types of Recommender Systems

At the heart of these systems lie various methodologies tailored to provide personalized suggestions. Collaborative filtering is one such method, relying on past user behaviors to predict future preferences. It operates under the assumption that users who have agreed in the past will continue to do so in the future (Goldberg et al., 1992) [2]. In contrast, content-based filtering focuses on item attributes, recommending items by comparing the content of the items and a user profile, with content described in terms of several descriptors that are inherent to the item (Pazzani and Billsus, 2007) [3]. Hybrid methods combine both collaborative and content-based filtering to provide recommendations, aiming to leverage the strengths and mitigate the weaknesses of both methods (Burke, 2002) [4][5]. As the digital landscape continues to evolve, so do recommendation systems, with each method offering unique advantages tailored to specific applications and user needs.

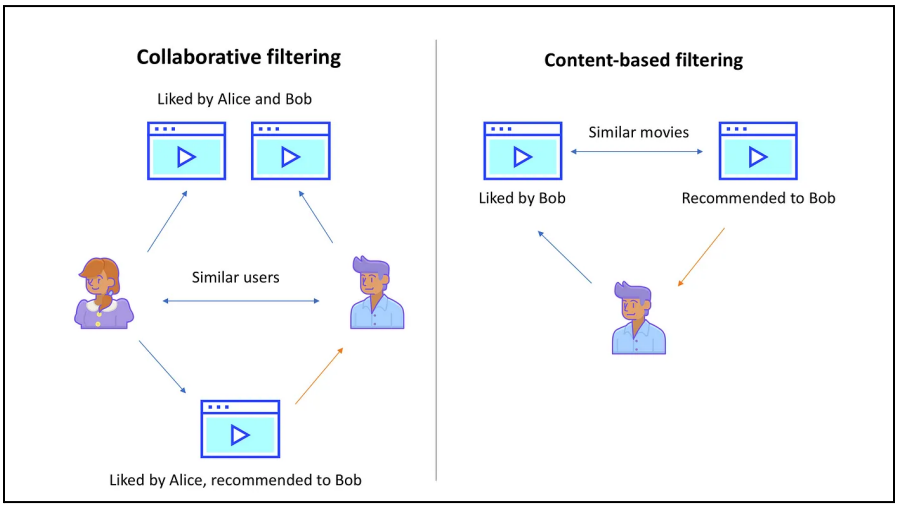

Figure 2.0 (a): Collaborative Filtering vs Content-Based Filtering (Source [5])

2. Collaborative Filtering

Collaborative filtering is a technique rooted in the idea of user similarity. By analyzing past user behaviors, the system can predict future preferences based on the behaviors of similar users. Spoorthy G. et al. (2021) [6] delved into this concept, proposing a hybrid algorithm that combined the strengths of K-nearest neighbors and restricted Boltzmann machines. Fong et al. (2008) [7] explored

the adaptability of collaborative filtering, introducing a genetic algorithm-based approach. Additionally, M. Muhammad (2021) [8] presented an innovative approach to user-based collaborative filtering using agglomerative clustering, emphasizing the technique's potential to enhance recommendation accuracy.

3. Content-Based Filtering

Content-based filtering operates on the principle of matching content attributes with user preferences. By analyzing the content of items and comparing them to a user's profile, the system can make tailored recommendations. Tianyu Li et al. (2018) [9] tackled the challenge of data sparsity in recommender systems, introducing a method that leverages heterogeneous auxiliary information for content-based filtering. Kyung-Yong Chung (2008) [10] expanded the scope of content-based filtering, proposing a system that uses image-based filtering for personalized item recommendations. Furthermore, Salter and Antonopoulos (2006) [11] introduced the CinemaScreen recommender agent, which combines collaborative and content-based filtering to recommend films, emphasizing the potential of hybrid approaches.

4. Hybrid Systems

Hybrid recommender systems have emerged as a powerful approach to combine the strengths of both collaborative and content-based filtering methods, aiming to mitigate the limitations inherent in each. These systems can be implemented in various ways: by making predictions based on the weighted sum of different recommendation techniques, by integrating the outputs of separate recommendation models, or by unifying different recommendation strategies into a single cohesive model (Burke, 2002) [4]. The primary advantage of hybrid systems is their ability to provide accurate recommendations even when data is sparse, addressing the cold-start problem that often plagues collaborative filtering systems. Furthermore, by leveraging diverse data sources and techniques, hybrid systems can offer more comprehensive and nuanced recommendations, catering to a broader spectrum of user preferences (Ricci et al., 2011) [1].

5. Recent Advancements

In recent years, recommender systems have witnessed significant advancements, largely driven by the integration of sophisticated machine learning techniques. Deep learning, especially the use of neural networks, has become a cornerstone in modern recommendation algorithms, enabling them to capture intricate patterns in vast datasets with enhanced accuracy (Zhang et al., 2019) [12]. Furthermore, the rise of hybrid models, which combine traditional collaborative and content-based filtering with newer methods, has addressed longstanding challenges such as the cold-start problem. These innovations not only improve recommendation quality but also ensure scalability in handling ever-growing user and item datasets (Chen et al., 2020) [13].

6. Streaming Platforms and Recommender Systems:

Streaming platforms, such as Netflix, Spotify, and YouTube, have revolutionized the way content is consumed in the digital age. Central to their success is the implementation of sophisticated recommender systems. These systems curate vast libraries of content, tailoring selections to individual users based on their preferences, viewing history, and even the behaviors of similar users. The goal is twofold: to enhance user engagement by delivering relevant content and to introduce users to new content they might not have discovered on their own. As the competition among streaming platforms intensifies, the precision and effectiveness of these recommender systems become critical differentiators. They not only drive user satisfaction but also play a pivotal role in retaining subscribers and reducing churn rates. The dynamic nature of streaming content, coupled with the continuous influx of user data, presents both challenges and opportunities for the evolution of recommendation algorithms in this domain (Covington et al., 2016) [14].

7. The MovieLens Dataset and its History:

The MovieLens dataset stands as a testament to the evolution of recommendation systems. Developed by the GroupLens Research Project at the University of Minnesota, this dataset offers a comprehensive collection of user ratings on movies. González et al. (2022) [15] utilized this dataset to explore potential biases in collaborative filtering-based recommender systems, highlighting the importance of understanding underlying data structures and user behaviors . Fong et al. (2008) [7] further showcased the dataset's versatility by integrating it with IMDB data, exploring the potential of genetic algorithms in collaborative filtering.

In the realm of recommender systems, the MovieLens datasets have emerged as a benchmark for both research and algorithmic advancements. Harper and Konstan (2015) [16] provide an in-depth exploration of the history and context of these datasets in their seminal work published in the ACM Transactions on Interactive Intelligent Systems. The authors delve into the evolution of the MovieLens datasets, highlighting their significance in fostering advancements in the field of recommendation systems. They underscore the datasets' role in addressing challenges related to scalability, algorithmic accuracy, and the nuances of user behavior. The comprehensive nature of the MovieLens datasets, encompassing diverse user interactions and movie attributes, has made them an invaluable resource for researchers aiming to develop cutting-edge recommendation algorithms. This paper not only offers insights into the intricacies of the datasets but also situates them within the broader context of recommendation research, emphasizing their continued relevance in the face of rapidly evolving technological landscapes.

8. Foundational Reference: Acknowledging the MovieLens 1M Deep Dive

A pivotal influence on the direction and methodology of this study has been the article titled "MovieLens 1M Deep Dive: Part I" [17] published on Towards Data Science. This comprehensive piece not only offers a meticulous exploration of the MovieLens 1M dataset but also provides a structured approach to understanding and analyzing it. The article's insights and methodologies have been instrumental in shaping the foundational aspects of this research. By adopting and building upon the techniques and perspectives presented in the article, this study has been able to delve deeper into the intricacies of the dataset. The invaluable guidance from the article has ensured that the research remains grounded in proven methodologies while also allowing for innovative explorations. It is essential to acknowledge the significant contribution of this article in providing a robust framework upon which this research has been constructed.

Chapter 3 Exploratory Data Analysis

The MovieLens 1M dataset, which is a widely recognized benchmark in the domain of recommendation systems, was utilized in this study [16]. This dataset comprises three main tables: “Users”, “Movies**”**, and “Ratings”.

The Users table provides demographic information about the users, including age, gender, occupation, and zip code. The Movies table contains details about each movie, such as its title and associated genres. The Ratings table captures the interactions between Users and Movies, detailing which user rated which movie, along with the rating value and timestamp of the rating.

With 1 million ratings from 6,040 users on 3,952 movies, this dataset offers a comprehensive snapshot of user preferences and movie characteristics, making it an invaluable resource for studying and evaluating recommendation systems.

In this chapter, exploratory data analysis is conducted to understand the dataset's structure, identify key features, and uncover patterns and relationships in the data. Statistical analysis and data visualization techniques are used to gain insights into the distribution of ratings, user behavior, and movie characteristics.

3.1 Exploring the “Movies” Dataset

The "Movies" dataset has 3,952 movies and each entry represents a unique movie and is associated with specific attributes that offer insights into the film's characteristics.

1. **Movie ID**: A unique identifier assigned to each movie
2. **Title**: Name and year of release of the movie. The year is useful when dealing with movies that might share similar titles but differ in their release date.
3. **Genres**: Movies are categorized into one or multiple genres, ranging from Action, Adventure, and Animation to War, Western, and others. Each movie may be classified into several genres, and in this study, any movie in question was counted multiple times for each genre.

Movie_ID	Movie Name	Movie Genre
1	Toy Story (1995)	Animation|Children's|Comedy
2	Jumanji (1995)	Adventure|Children's|Fantasy
3	Grumpier Old Men (1995)	Comedy|Romance
4	Waiting to Exhale (1995)	Comedy|Drama
5	Father of the Bride Part II (1995)	Comedy
6	Heat (1995)	Action|Crime|Thriller
7	Sabrina (1995)	Comedy|Romance
8	Tom and Huck (1995)	Adventure|Children's
9	Sudden Death (1995)	Action
10	GoldenEye (1995)	Action|Adventure|Thriller
11	American President, The (1995)	Comedy|Drama|Romance
12	Dracula: Dead and Loving It (1995)	Comedy|Horror
13	Balto (1995)	Animation|Children's
14	Nixon (1995)	Drama
15	Cutthroat Island (1995)	Action|Adventure|Romance

Figure 3.1 (a): First 15 Entries of the Movies Dataset

3.1.1 Distribution of Movies Per Genre

This analysis aimed to provide insights into the volume and variety of movies available in each genre category. Utilizing a visual representation in the form of a bar chart, the distribution was clearly delineated, showcasing the prevalence of certain genres over others in the dataset. Some genres boasted a vast array of movies, i.e. Drama, indicating their widespread production and potential popularity in the cinematic world. In contrast, other genres, i.e. Film-Noir, had a more modest representation, suggesting niche interests or evolving cinematic trends.

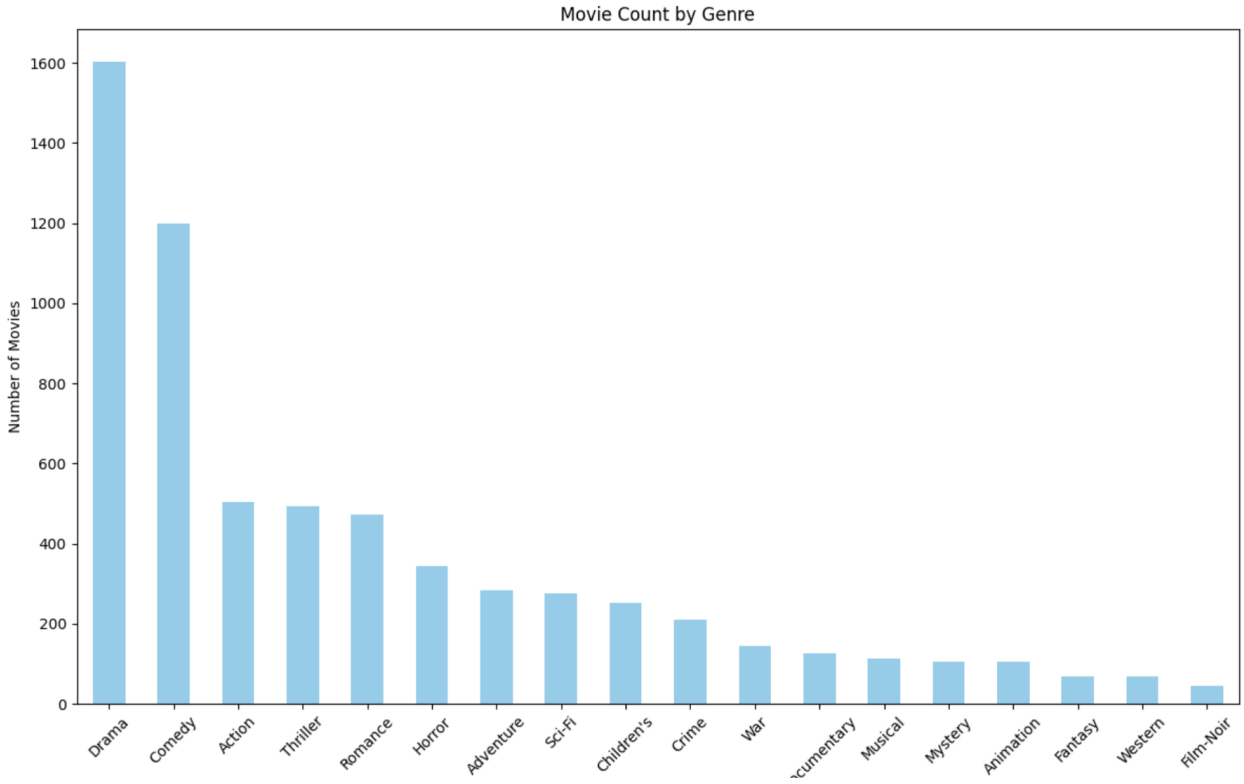

Figure 3.1 (b): Distribution of Movies Per Genre

Understanding this distribution is crucial, as it not only reflects the industry's production tendencies but also offers a glimpse into the diverse tastes and preferences of movie audiences. Such insights can be instrumental in guiding future movie production decisions, marketing strategies, and recommendation system refinements.

3.2 Exploring the "Ratings" Dataset

Each of the one million rows in this dataset signifies a user's evaluation of a particular film, capturing three primary attributes:

1. **User ID**: A unique identifier denoting an individual user
2. **Movie ID**: Unique identifier from the "Movies" dataset, linking the rating to a specific film.
3. **Rating**: Represented on a scale of 1 to 5, this numeric value captures a user's assessment of a film, with higher values indicating greater appreciation.

user_id	movie_id	rating	timestamp
1	1193	5	978300760
1	661	3	978302109
1	914	3	978301968
1	3408	4	978300275
1	2355	5	978824291
1	1197	3	978302268
1	1287	5	978302039
1	2804	5	978300719
1	594	4	978302268
1	919	4	978301368
1	595	5	978824268

Figure 3.2 (a): First 10 Entries of the Ratings Dataset

3.2.1 Frequency of Ratings by Genre

The frequency of ratings provided insights into the popularity of genres, revealing which genres were more commonly watched and, therefore, rated by users in general. A notable challenge encountered was the presence of movies associated with multiple genres (e.g., a movie categorized as both "Action" and "Drama"). To ensure a comprehensive understanding of each genre's ratings, each genre was treated individually, even if associated with the same movie. This decision provided a clearer insight into the ratings of each genre, eliminating potential distortions from multi-genre movies. The following chart showcases the volume of ratings each genre received, highlighting their relative popularity.

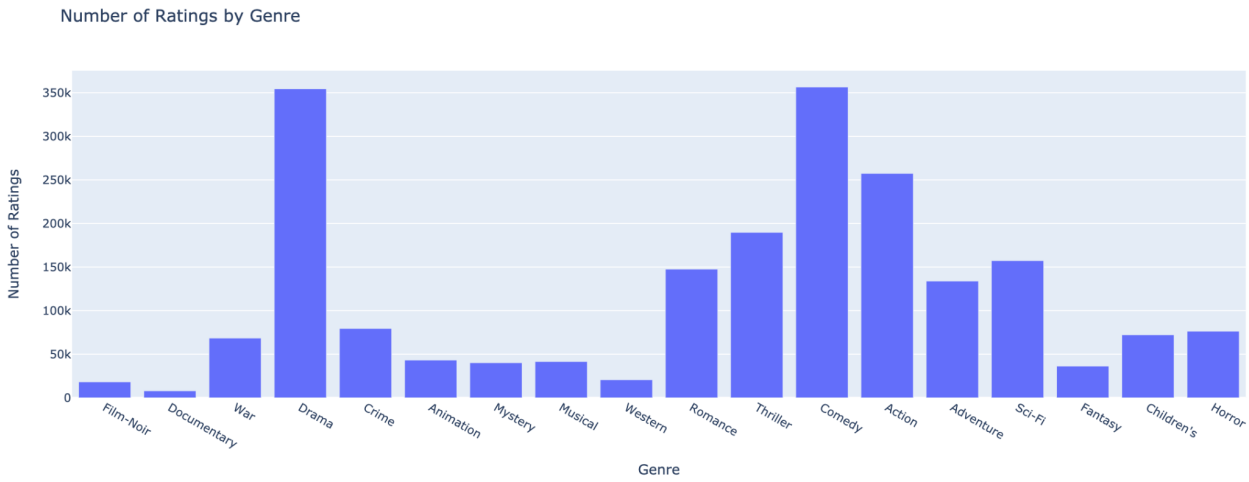

Figure 3.1 (b): Frequency of Ratings by Genre

3.2.2 Average Rating of Genres Grouped by Gender

This overview of the average rating of each genre, segmented by gender proved insightful. Males exhibited a tendency to rate Sci-Fi movies more favorably than females, while females demonstrated a predilection for giving higher ratings to romance movies compared to males. These observations, in my analysis, align with conventional gender-based cinematic preferences, serving as an indication of the dataset's reliability and consistency. The following visualization reveals gender-based preferences in movie ratings.

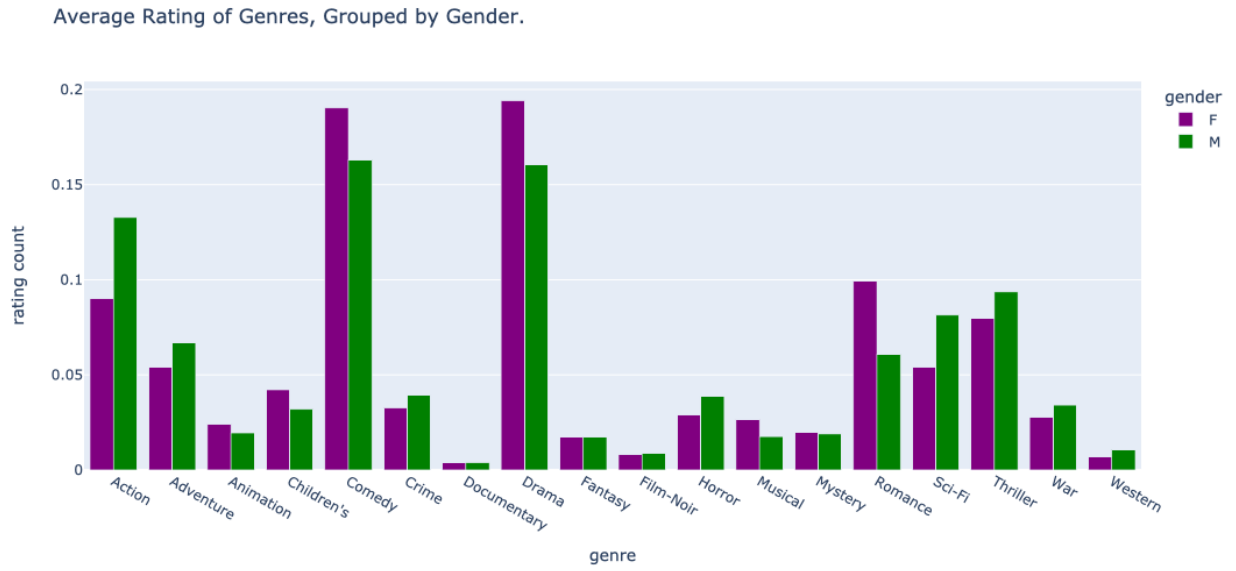

Figure 3.2 (c): Average Rating of Genres Grouped by Gender

These findings not only shed light on user preferences but also underscored the importance of genre as a determinant in movie selection and viewership satisfaction.

3.3 Exploring the “Users” Dataset

The "Users" dataset consists of information on 6,040 users and offers demographic details such as age, gender, occupation, and zip code. Such data plays a pivotal role in assessing movie predictions which are done in the later stages of this thesis report.

Figure 3.3 shows the “occupation” column in the users.dat file of the MovieLens 1M dataset is initially encoded with numerical values, each representing a specific occupation type. To enhance the interpretability and clarity of the data, these numerical encodings were replaced with their corresponding textual descriptions. This transformation was achieved by extracting the occupation mapping from the accompanying README file, which provided a clear correspondence between each numerical code and its respective occupation description. By doing so, the dataset was enriched with more meaningful information, allowing for a more intuitive understanding of user occupations and facilitating subsequent data analysis tasks.

user_id	gender	age	zip_code	occupation_id
1	F	1	48067	10
2	M	56	70072	16
3	M	25	55117	15
4	M	45	2460	7
5	M	25	55455	20
6	F	50	55117	9
7	M	35	6810	1
8	M	25	11413	12
9	M	25	61614	17
10	F	35	95370	1

occupation_type
"K-12 student"
"self-employed"
"scientist"
"executive/managerial"
"writer"
"homemaker"
"academic/educator"
"programmer"
"technician/engineer"
"academic/educator"

Figure 3.3: First 10 Entries of the Modified Users Dataset

In Chapter 5, a more detailed analysis is conducted on selected users from this dataset, as their top-rated movies were assessed to gain deeper insights into individual preferences and patterns.

Chapter 4 Data Preparation

This is the data pre-processing and examining step of this study and includes retrieving movie content information. This process ensures the data is clean and in the right format for the subsequent steps. It involves handling missing values, encoding categorical variables, and extracting relevant features for content-based filtering.

4.1 Data Enrichment with Wikipedia Summaries

Movie plot summaries are fetched from Wikipedia to obtain a concise description for each movie, which could be beneficial for various analyses, including content-based filtering and natural language processing tasks. It involved the following steps:

- **Movie List Compilation**: A list of movie titles was extracted from the `movies_df` dataframe, which contained information about approximately 4,000 movies from the MovieLens 1M dataset.
- **Wikipedia API Integration**: The `wikipedia` Python package was utilized to programmatically query Wikipedia. For each movie title in the list, an attempt was made to retrieve the first ten sentences of its Wikipedia summary.
- **Handling Ambiguities**: In cases where a movie title matched multiple Wikipedia entries (a disambiguation error), the entry was skipped to ensure data accuracy. Similarly, if no Wikipedia page was found for a particular movie, that movie was also skipped.

- **Data Persistence**: To avoid the time-consuming process of fetching data from Wikipedia in future sessions, data was saved locally. This ensured that the enriched data was readily available for subsequent analyses without the need for internet access or additional API calls beyond the initial setup.

movie_name	movie_plot
Toy Story (1995)	Toy Story is an American media franchise owned by The Walt Disney Company. It centers on toys that, unknown to humans, are secretly living, sentient creatures. It began in 1995 with the release of animated feature film of the same name, which focuses on a diverse group of toys that feature a classic cowboy doll named Sheriff Woody and a modern spaceman action figure named Buzz Lightyear. The Toy Story franchise consists mainly of five CGI-animated feature films: Toy Story (1995), Toy Story 2 (1999), Toy Story 3 (2010), Toy Story 4 (2019), and the spin-off prequel film within a film Lightyear (2022). A fifth film was recently announced. It also includes the 2D-animated direct-to-video spin-off film within a film Buzz Lightyear of Star Command: The Adventure Begins (2000) and the animated television series Buzz Lightyear of Star Command (2000–01) which followed the film. The first Toy Story was the first feature-length film to be made entirely using computer-generated imagery. The first two films were directed by John Lasseter, the third film by Lee Unkrich (who acted as co-director of the second film alongside Ash Brannon), the fourth film by Josh Cooley, and Lightyear by Angus MacLane. Produced on a total budget of $720 million, the Toy Story films have grossed more than $3.3 billion worldwide, becoming the 20th highest-grossing film franchise worldwide and the third highest-grossing animated franchise. Each film of the main series set box office records, with the third and fourth included in the top 50 all-time worldwide films.
Jumanji (1995)	Jumanji is a 1995 American urban fantasy adventure film directed by Joe Johnston from a screenplay by Jonathan Hensleigh, Greg Taylor, and Jim Strain, based on the 1981 children's picture book of the same name by Chris Van Allsburg. The film is the first installment in the Jumanji film series. It stars Robin Williams, Kirsten Dunst, David Alan Grier, Bonnie Hunt, Jonathan Hyde, and Bebe Neuwirth. The story centers on a supernatural board game that releases jungle-based hazards upon its players with every turn they take. Jumanji was released on December 15, 1995, by Sony Pictures Releasing. The film received mixed reviews from critics, but was a box-office success, grossing $263 million worldwide on a budget of approximately $65 million. It was the tenth-highest-grossing film of 1995. The film spawned an animated television series, which aired from 1996 to 1999, and was followed by a spin-off film, Zathura: A Space Adventure (2005), and two indirect sequels, Jumanji: Welcome to the Jungle (2017) and Jumanji: The Next Level (2019). == Plot == In 1969, Alan Parrish lives with his parents, Sam and Carol, in Brantford, New Hampshire. One day, he escapes a group of bullies and retreats to Sam's shoe factory.
Grumpier Old Men (1995)	Grumpier Old Men is a 1995 American romantic comedy film, and a sequel to the film Grumpy Old Men. It stars Jack Lemmon, Walter Matthau, Ann-Margret, Sophia Loren, Burgess Meredith (in his final film role), Daryl Hannah, Kevin Pollak, Katie Sagona and Ann Morgan Guilbert. Grumpier Old Men was directed by Howard Deutch, with the screenplay written by Mark Steven Johnson and the original music score composed by Alan Silvestri. Meredith developed Alzheimer's disease and had to be coached through his role in the film. He died in 1997. == Plot == The feud between Max and John has cooled and they have become good friends. Their children, Melanie and Jacob have become engaged. Meanwhile, John is enjoying his marriage to new wife Ariel. John and Max still call each other "moron" and "putz" respectively, but with friendly intentions. The spring and summer fishing season is in full swing with the annual quest to catch "Catfish Hunter," an unusually large catfish that seems to enjoy eluding anyone that tries to catch it.
Waiting to Exhale (1995)	Waiting to Exhale is a 1995 American romance film directed by Forest Whitaker (in his feature film directorial debut) and starring Whitney Houston and Angela Bassett. The film was adapted from the 1992 novel of the same name by Terry McMillan. Lela Rochon, Loretta Devine, Dennis Haysbert, Michael Beach, Gregory Hines, Donald Faison, and Mykelti Williamson rounded out the rest of the cast. The original music score was composed by Kenneth "Babyface" Edmonds. The story centers on four women living in the Phoenix metropolitan area and their relationships with men and one another. All of them are "holding their breath" until the day they can feel comfortable in a committed relationship with themselves. == Plot == Four friends (Savannah, Robin, Bernadine, and Gloria) get together frequently to support one another and listen to each other vent about life and love. They each want to be in a romantic relationship, but they each have difficulties finding a good man. Successful television producer Savannah "Vannah" Jackson believes that one day her married lover will leave his wife for her. She later realizes that he won't, and that she must find her own man who will love her for who she really is.
Father of the Bride Part II (1995)	Father of the Bride Part II is a 1995 American comedy film starring Steve Martin, Diane Keaton, and Martin Short. It is a sequel to Father of the Bride, remake of the 1951 film Father's Little Dividend which was the sequel to the original 1950 titular film, and fourth installment overall in the Father of the Bride franchise. == Plot == The film begins four years after the events of the first one, with George Banks telling the audience he is ready for the empty nest he will soon receive with all of his children grown up. Shortly thereafter, Annie tells the family that she is pregnant. George begins to mildly panic, insisting he is too young to be a grandfather. He has his assistant make a list of people who are older than him, dyes his hair brown, and after the roof leaks, decides he and Nina should sell the home their children have grown up in if one more thing goes wrong with it. Termites strike the house two weeks later. George sells the house to the Habibs without telling Nina. At dinner, after a discussion on whether the baby's last name will be hyphenated or not, George reveals the house has been sold. Nina is livid, as she and George have to be out in ten days and have no place to go.

Figure 4.1: Movie Plot Summaries Collected from Wikipedia

4.2 Merging Movie Metadata with Plots

This step involved integrating the 'movie_plots' dataframe, which contained the movie plots sourced from Wikipedia, with the primary 'movies_df' dataframe that stored the movie metadata. To accomplish this integration, the `pd.merge()` function from the pandas library was utilized. The merging operation was conducted based on the 'movie_name' column, which acted as a shared key between the two dataframes. By specifying the `how='inner'` argument, it was ensured that only the rows with matching movie names in both dataframes were retained.

The resulting dataframe was then saved as 'movies_df_with_plots.csv' in the designated Google Drive directory for further analysis and model training. This consolidated dataset not only provides movie metadata but also includes a comprehensive plot summary for each movie, which will be instrumental in the content-based filtering approach.

4.3 Handling Missing Wikipedia Summaries

During the data enrichment process, where movie plot summaries were fetched from Wikipedia, it was observed that not all movies from the MovieLens dataset had corresponding summaries on Wikipedia. Specifically, out of the total movies in the dataset, 323 movies did not have a retrieved summary. Several factors could account for this discrepancy:

1. **Non-existent Wikipedia Pages**: Some movie titles may not have had a "Plot" category or even a dedicated Wikipedia page. This is especially possible for less popular or niche films.

2. **Disambiguation Issue**s: Wikipedia might have multiple entries with similar or identical titles to a movie. In such cases, to ensure data accuracy, the retrieval process was designed to skip such titles rather than risk fetching incorrect or irrelevant information.

3. **API Limitations**: Occasional network or API call errors could have interrupted the data fetching process for some movies.

Missing Movie Titles
Beautiful Girls (1996)
Prom Night III: The Last Kiss (1989)
Species (1995)
Horse Whisperer, The (1998)
Bloody Child, The (1996)
Pie in the Sky (1995)
Star Trek: The Motion Picture (1979)
Jupiter's Wife (1994)
Vermin (1998)
Runaway Train (1985)

Figure 4.3: Some of the 323 Movies with Unavailable Plot Summaries

The dataset was refined to include only those movies for which plot information was available. Consequently, any ratings associated with these movies were also removed from the Ratings dataset. This step ensured that the analysis was based solely on movies with comprehensive data, resulting in a refined ratings_df containing 945,621 ratings, down from the original 1,000,209

4.3.1 Potential Drawbacks

While efforts were made to ensure a comprehensive dataset, it's essential to acknowledge that the exclusion of these movies may have several implications.

1. Firstly, the absence of these movies might introduce a selection bias, as the dataset no longer represents the full spectrum of movies, potentially omitting certain genres, time periods, or other categorical distinctions.

2. Secondly, any patterns or insights derived from the analysis might be limited in their generalizability to the broader set of movies.

3. Lastly, users who have a preference for these excluded movies might not receive as accurate or comprehensive recommendations as others.

Having meticulously prepared and cleaned the data, the following Chapter in this study delves deeper into the intricacies of user profiles. By understanding individual preferences and patterns, a clearer picture of movie ratings and their significance can be painted.

Chapter 5 Analysis of User Profiles and Movie Preferences

5.1 Motivation

The motivation behind this chapter is to gain a deeper understanding of individual user preferences. By analyzing specific users and their ratings, it is possible to identify patterns, preferences, and potential outliers in the dataset. It also aids in validating the data, ensuring that the recommendations generated later align with the users' actual preferences.

5.2 Analysis

Three distinct users were selected for detailed analysis:

5.2.1 First User (ID#1)

The first user in the Users dataset is identified as a female K-12 student, though an anomaly in the data suggests her age as 1, which likely indicates a data entry error. An exploration of her top-rated movies provides a blend of classic children's films and critically acclaimed dramas. Titles such as "Dumbo (1941)", "Toy Story (1995)", "Cinderella (1950)", and "Mary Poppins (1964)" align with popular choices for younger audiences, capturing the magic and innocence of childhood. However, interspersed among these are profound and mature films like "One Flew Over the Cuckoo's Nest (1975)", "Schindler's List (1993)", and "Rain Man (1988)".

First User's Top-Rated Movies	
Movie Name	**Rating**
One Flew Over the Cuckoo's Nest (1975)	5
Dumbo (1941)	5
Toy Story (1995)	5
Awakenings (1990)	5
Rain Man (1988)	5
Schindler's List (1993)	5
Cinderella (1950)	5
Sound of Music, The (1965)	5
Pocahontas (1995)	5
Mary Poppins (1964)	5

Figure 5.2 (a): First User's Top-Rated Movies

This juxtaposition suggests a diverse taste in movies, or perhaps the influence of family viewing choices. It underscores the importance of considering the multifaceted nature of individual preferences when developing recommendation systems.

5.2.2 Last User (ID#6040)

This last user in the Users dataset is identified as a 25-year-old male doctor. Examination of his top-rated movies reveals a predilection for classic and critically acclaimed films. His top-rated movies include cinematic masterpieces such as "8 1/2 (1963)", "Treasure of the Sierra Madre, The (1948)", and "Chinatown (1974)". Additionally, his appreciation for international cinema is evident in selections like "After Life (1998)" and "My Life as a Dog (Mitt liv som hund) (1985)". The inclusion of romantic comedies like "When Harry Met Sally... (1989)" showcases a diverse taste, spanning multiple genres.

Last User's Top-Rated Movies	
Movie Name	**Rating**
After Life (1998)	5
8 1/2 (1963)	5
Mystery Train (1989)	5
Microcosmos: Le peuple de l'herbe (1996)	5
Treasure of the Sierra Madre, The (1948)	5
Streetcar Named Desire, A (1951)	5
My Life as a Dog (Mitt liv som hund) (1985)	5
Grand Hotel (1932)	5
When Harry Met Sally... (1989)	5
Chinatown (1974)	5

Figure 5.2 (b): Last User's Top-Rated Movies

The feasibility of these ratings is underscored by the user's age and occupation; a young doctor might have a penchant for thought-provoking and timeless movies. This analysis underscores the importance of personalized recommendations, as individual preferences can be intricate and multifaceted.

5.2.3 Random User (User ID#5676)

The randomly selected user from the Users dataset is a 25-year-old male who identifies as a writer. A glance at his top-rated movies reveals a diverse taste in films, ranging from psychological thrillers to classic horror and indie dramas. For instance, he has given a high rating to "Hard 8 (a.k.a. Sydney, a.k.a. Hard Eight) (1996)", a crime thriller, and "The Birds (1963)", a classic horror by Alfred Hitchcock. Additionally, his preference for "Breaking the Waves (1996)", an intense drama, and "Repo Man (1984)", a cult classic, suggests an inclination towards films with deep narratives or unconventional storylines.

Random User's Top-Rated Movies	
Movie Name	**Rating**
Hard 8 (a.k.a. Sydney, a.k.a. Hard Eight) (1996)	5
Amityville Horror, The (1979)	5
New Jersey Drive (1995)	5
Birds, The (1963)	5
Repo Man (1984)	5
Life and Times of Hank Greenberg, The (1998)	5
Breaking the Waves (1996)	5
Assignment, The (1997)	5
Kika (1993)	5
April Fool's Day (1986)	5

Figure 5.2 (c): Random User's Top-Rated Movies

This user's movie preferences, combined with his profession as a writer, might indicate a penchant for intricate plots, character development, and unique storytelling techniques. Such insights can be valuable when tailoring movie recommendations for similar users in the dataset.

Upon gaining insights from the user profiles and their distinct movie preferences, it becomes imperative to establish a systematic approach for the subsequent stages of this study. The next chapter outlines a rigorous experiment setup, detailing how the dataset was partitioned and the metrics chosen for evaluation. This structured approach ensures the reliability and validity of prediction results from different algorithms.

Chapter 6 Methodology

6.1 Experiment Setup

6.1.1 Target Movie and User Selection

During Collaborative Filtering, a consistent target movie was selected to evaluate the performance of various recommendation algorithms. The chosen movie is the first movie in the Movies dataset and it is "Toy Story (1995)", a widely recognized and popular film that has been rated by a significant number of users. This selection provides a robust benchmark, given its diverse range of ratings and its familiarity among a broad audience.

During Collaborative Filtering, in addition to the designated movie for evaluation, we opted to use ‘user_1’, the female K-12 student previously examined in the preceding chapter, as the primary user for generating top predictions. The choice of this user was made due to their varied history of movie ratings, which was intended to ensure a thorough assessment of the recommendation algorithms.

6.1.2 Number of Top Predictions

During Collaborative Filteriing, in order to maintain consistency in the evaluation and ensure a manageable number of recommendations, the top 5 predictions were generated for 'user_1' using each algorithm. This number was chosen as it represents a realistic scenario where users are provided with a limited set of top recommendations, which they are more likely to explore.

6.2 Dataset Splits

6.2.1 Train/Test Split

In machine learning and statistics, data is often split into a training set and a testing set. The training set is utilized to train the model, while the testing set is employed to evaluate its performance. This method assists in gauging how well the model might perform on unseen data [18].

In this study, the dataset was divided using a standard train/test split. 75% of the data was allocated for training and the remaining 25% for testing. This approach facilitated a straightforward evaluation of the model's performance on data that hadn't been seen before

6.2.2 Leave-One-Out Cross Validation (LOOCV)

LOOCV is a special case of k-fold cross-validation where k is equal to the number of data points in the dataset. In this method, a single data point is designated as the validation set, and the remaining data points constitute the training set. This process is reiterated for each data point in the dataset, ensuring that the recommendation model's performance is assessed on unseen data for each user [19].

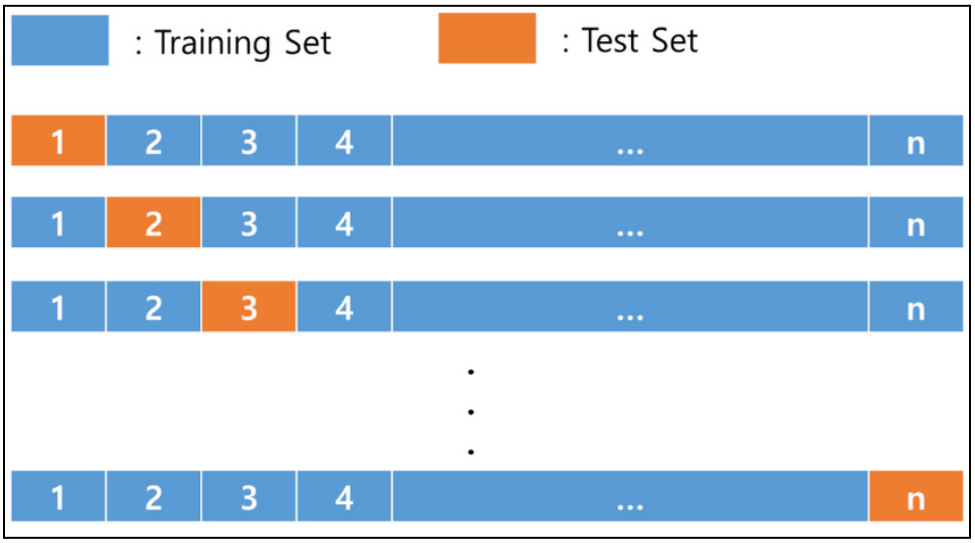

Figure 6.2: Schematic Representation of LOOCV (Source: [20])

This approach is particularly suitable for recommendation systems, especially when evaluating the quality of top-N recommendations.

In the context of this study, LOOCV was employed to evaluate the Hit Rate of the recommendation model. The Hit Rate is an evaluation metric used in the scope of this study and is discussed in the next section.

6.3 Metrics for Evaluation

The evaluation of any recommender system is pivotal to understanding its effectiveness and accuracy. In this chapter, the focus is placed on two critical metrics: Root Mean Square Error (RMSE) and Hit Rate. Both metrics offer unique insights into the system's performance, An understanding of these metrics will provide a clear picture of the system's strengths and areas for improvement, ensuring that the recommendations generated are both precise and pertinent to a particular user.

6.3.1 Root Mean Square Error (RMSE)

In the context of recommendation systems, RMSE is a widely used metric to evaluate the accuracy of prediction models [21]. It quantifies the difference between the predicted values and the actual values by taking the square root of the average of the squared differences. Mathematically, it is represented as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y})^2}$$

y_i is the actual value,

$\widehat{y_i}$ is the predicted value

n is the number of observations.

Figure 6.3: Equation to calculate RMSE (Source [22])

In the study, RMSE was used to measure measures the differences between the predicted ratings and the actual ratings given by users. A lower RMSE indicates a better performance, as it means the algorithm's predictions are closer to the actual ratings [23].

6.3.2 Hit Rate (LOOCV set)

The Hit Rate metric measures the proportion of actual positive interactions (e.g., movie ratings) that are correctly predicted by the model [24]. A "hit" occurs when a recommended item appears in the user's actual list of liked items. The Hit Rate is then the fraction of hits over the total number of recommendations made. It represents the percentage of times the left-out movie was in the top-K ratings for that user.

$$\text{Hit Rate} = \frac{number\ of\ hits}{total\ number\ of\ left{-}out\ predictions}$$

Derived from the Leave-One-Out Cross Validation (LOOCV) method, the hit rate measures the percentage of times the left-out movie (in this case, "Toy Story") appears in the top-K recommendations for 'user_1'. A higher hit rate indicates that the recommendation system is more likely to suggest movies that the user would genuinely enjoy. Calculating the Hit rate metric is very computationally intensive and it took multiple hours to generate Hit Rates for each case.

In Summary, both the RMSE and Hit Rate metrics provide a comprehensive view of the algorithm's performance, with RMSE focusing on the accuracy of the predicted ratings and hit rate emphasizing the relevance of the top recommendations.

6.4 Baseline Metrics

Establishing a baseline is crucial for assessing the effectiveness of more sophisticated recommender systems. Baseline metrics provide a fundamental measure of performance, often derived from simple algorithms or even random predictions [25]. These metrics serve as a starting point against which the results of more complex recommendation algorithms can be compared. By understanding the performance of a basic or random model, a researcher can gauge how much value is added by incorporating advanced features or techniques into their recommendation systems.

6.4.1 Normal Predictor

The Normal Predictor is a recommendation algorithm that predicts ratings based on the distribution of ratings present in the training dataset. Instead of providing recommendations based on user-item interactions or similarities, the Normal Predictor generates ratings by drawing from a normal distribution characterized by the mean and standard deviation of the ratings in the training set [26]. This approach essentially provides a baseline model against which more sophisticated recommendation algorithms can be compared.

In this study, the Normal Predictor was employed to generate baseline predictions. The RMSE and Hit Rate obtained from this predictor provided initial benchmarks for evaluating the effectiveness of subsequent recommendation models.

Normal Predictor Metrics
RMSE: 1.5047
HitRate: 0.013245033112582781

Figure 6.4: RMSE & Hit Rate of the Normal Predictor

6.4.2 Evaluation

As previously discussed, the RMSE measures the average magnitude of the errors between predicted and observed ratings. An RMSE of 1.5047 means that, on average, the predictions made by the NormalPredictor are about 1.5047 rating points away from the actual ratings. Given that the rating scale is from 1 to 5, this error is relatively high, but it's expected since the NormalPredictor is essentially making random predictions.

Hit Rate measures the proportion of times the movie left out in the Leave-One-Out Cross Validation (LOOCV) was in the top-K ratings for that user. A hit rate of 0.0132 (or 1.32%) means that in only about 1.32% of the cases, the movie that was left out was among the top recommended movies for the user. This is a low Hit Rate, but again, it's expected for a model that's making random predictions.

As this study transitions into exploring collaborative and content-based filtering, it's essential to keep the baseline results in perspective. These foundational metrics will guide the evaluation of more advanced techniques, ensuring that the pursuit remains focused on models that offer genuine value and improved performance.

Chapter 7 Collaborative Filtering

Collaborative filtering operates under the assumption that users who have agreed in the past tend to agree again in the future about their preference for certain items. The method identifies patterns based on historical interactions between users and items, such as movie ratings, to make predictions about what users might like in the future [27]. There are two primary types of collaborative filtering techniques:

User-based Collaborative Filtering (UBCF)	**Item-based Collaborative Filtering (IBCF)**
UBCF finds users who are similar to the target user and recommends items that similar users have liked. For instance, if user A and user B have both liked certain movies in the past and user A likes a new movie that user B hasn't seen yet, then that movie will likely be recommended to user B.	Instead of finding user similarity, IBCF focuses on item similarity. If two items tend to be liked by the same set of users, then they are considered to be similar. So, if a user likes one of those items, the other will likely be recommended.

Table 7.0: Types of Collaborative Filtering Techniques [28]

7.1 Algorithms Implemented

In this study, two collaborative filtering algorithms were used and both algorithms have their roots in linear algebra and statistics. To implement these algorithms, the Surprise library from Python Scikit was utilized [29]. This library offers a comprehensive suite of tools, such as matrix factorization, and provides a consistent interface that makes comparing algorithm performance easier.

1. **Singular Value Decomposition (SVD)**

SVD is a matrix factorization technique that decomposes a matrix into multiple other matrices [30]. In the context of recommender systems, SVD doesn't fall strictly into the category of user-based or item-based collaborative filtering. Instead, it decomposes the user-item interaction matrix into multiple matrices representing latent factors. However, once factorized, the latent factors can

represent patterns from both users and items. This decomposition allows the system to predict missing interactions, essentially predicting how a user might rate an item they haven't interacted with yet. Therefore, in essence, SVD captures information from both user and item perspectives but isn't categorized as purely user-based or item-based.

2. **K-Nearest Neighbors (KNN)**

KNN is a memory-based collaborative filtering approach that computes the similarity between items or users [31]. When used in recommender systems, it predicts a user's interest by collecting preferences from many other users (neighbors). It computes the similarity between items or users and recommends things based on how similar users rated them. The intuition is that users who agreed in the past will agree again in the future about the rating of certain items.

KNN can be both user-based and item-based, depending on how it's configured. For this study, user-based KNN with Pearson similarity metric was utilized, considering the 50 nearest neighbors [32].

7.2 Evaluation

7.2.1 Metrics Comparison

Two collaborative filtering techniques were implemented and evaluated on the dataset: User-Based Collaborative Filtering (UBCF) using KNN and Singular Value Decomposition (SVD). Based on the results, a clear hierarchy in algorithm performance emerges when considering both RMSE and Hit Rate metrics.

Algorithm	RMSE	Hit Rate (%)
Normal Predictor (Baseline)	1.5047	1.32
SVD	0.8771	3.26
KNN (User-Based)	0.9614	0.17

Table 7.2: RMSE & Hit Rate of the Implemented Algorithms (Part1)

Normal Predictor:

While one might dismiss the Normal Predictor due to its simplistic nature, its performance metrics warrant attention. With an RMSE of 1.5047, it's evident that its predictions, on average, deviate more from actual user ratings compared to SVD. Despite its rudimentary approach, its Hit Rate of 1.32%, although lower than that of SVD, is higher than the KNN-based UBCF. This underscores the unpredictable nature of user preferences and highlights that even a random guess when based on the distribution of existing ratings, can occasionally align with user inclinations. The importance of

sophisticated algorithms like SVD in enhancing recommendation quality was underscored by this baseline.

KNN (User-Based Collaborative Filtering):

An RMSE of 0.9614 was observed for the KNN-based UBCF approach, indicating a moderate prediction error. A Hit Rate of 0.17% suggests that the top-N recommendations were matched with the left-out ratings for a small fraction of the users. Despite KNN's theoretical foundation and its ability to leverage user-item interactions directly, its performance here emphasizes the importance of empirical validation. It's a testament to the fact that algorithmic sophistication doesn't always guarantee superior results, especially in the domain of recommendation systems. While the potential was shown by the KNN-based UBCF, its performance might benefit from further optimization and tuning.

SVD:

Superior performance was exhibited by the SVD approach compared to the KNN-based UBCF in terms of both RMSE and Hit Rate. An RMSE of 0.8771 suggests a relatively lower prediction error, while a Hit Rate of 3.25% indicates a better alignment of top-N recommendations with the left-out ratings. The strength of matrix factorization techniques like SVD in capturing latent factors and providing more accurate recommendations was highlighted by these results.

In summary, Among the techniques evaluated, the best performance in terms of both prediction accuracy (RMSE) and the quality of top-N recommendations (Hit Rate) was demonstrated by SVD.

7.2.2 Further Analysis of Top Predictions and Similar Movies

When identifying movies similar to "Toy Story," the algorithm's selections are predominantly animated films, such as "Aladdin" and "A Bug's Life." This indicates the algorithm's proficiency in pinpointing movies with shared thematic and stylistic elements. The recommendations underscore the algorithm's capability to both cater to a user's broad preferences and identify films with closely aligned characteristics.

The SVD algorithm's top predictions for User_1 encompass a diverse range of genres, from the thriller "Eye of the Beholder" to the drama "I Shot a Man in Vegas." This variety suggests that the algorithm recognizes User_1's eclectic taste, offering recommendations that span different cinematic themes and styles.

Top Similar Movies (SVD)	
movie_name	**genre**
Toy Story (1995)	Animation|Children's|Comedy
Aladdin (1992)	Animation|Children's|Comedy|Musical
Beauty and the Beast (1991)	Animation|Children's|Musical
Bug's Life, A (1998)	Animation|Children's|Comedy
Toy Story 2 (1999)	Animation|Children's|Comedy

Top 5 Predictions for user_1 (SVD)	
movie_name	**genre**
Eye of the Beholder (1999)	Thriller
Bullets Over Broadway (1994)	Comedy
Thinner (1996)	Horror|Thriller
Six-String Samurai (1998)	Action|Adventure|Sci-Fi
I Shot a Man in Vegas (1995)	Comedy

Table 7.2 (a): Top Similar Movies & Top Predictions (SVD)

The KNN algorithm's top predictions for User_1 present an intriguing mix of films, ranging from the historical drama "I Am Cuba" to the romantic drama "Firelight." This selection suggests that the KNN model has identified a nuanced palette of User_1's preferences, offering recommendations that touch upon various cinematic narratives and periods.

Top 5 Predictions for user_1 (KNN)	
movie_name	**genre**
I Am Cuba (Soy Cuba/Ya Kuba) (1964)	Thriller
Criminal Lovers (Les Amants Criminels) (1999)	Comedy
Dangerous Game (1993)	Horror|Thriller
Smashing Time (1967)	Action|Adventure|Sci-Fi
Firelight (1997)	Comedy

Table 7.2 (b): Top Predictions (KNN)

Unlike the SVD algorithm, we did not derive a list of movies similar to "Toy Story". This is primarily due to the inherent differences in the two algorithms. While SVD can easily generate item-based similarities based on latent factors, the KNN model relies on explicit feature similarities. Therefore, For the user-based KNN approach, the concept of item similarity doesn't exist.

Chapter 8 Content-Based Filtering

Content-based filtering is a recommendation approach that leverages descriptive attributes of items to recommend additional items similar to what a user likes, based on their previous actions or explicit feedback. Unlike collaborative filtering, which is based on user-item interactions, content-based filtering focuses on the properties of the items themselves. In the context of movie recommendations, this study employs movie plots and user data (age and gender) as the primary attributes for generating recommendations. The underlying principle is that if a user has shown interest in a particular movie, they are likely to be interested in other movies with similar content or themes, or if users of similar profiles have shown certain preferences, they might have similar tastes [1]. This chapter delves into the methodologies and results of applying content-based filtering on the MovieLens 1M dataset, offering insights into its efficacy and challenges.

8.1 Feature Extraction and Preprocessing

In the realm of content-based filtering, the quality and nature of features play a pivotal role in determining the efficacy of recommendations. For this study, two primary sets of features were extracted and preprocessed: movie plot summaries and user data.

8.1.1 Movie Plots

- Text Cleaning: Before any meaningful analysis could be conducted on the movie plot summaries, it was imperative to clean the text data. This involved removing any special characters, numbers, and converting all text to lowercase to ensure uniformity.

- Stemming: To reduce words to their base or root form, stemming was employed. This process aids in consolidating words of similar meaning under a common base, thereby reducing the dimensionality of the data and improving the efficiency of subsequent processes.

- Vectorization using TF-IDF: The Term Frequency-Inverse Document Frequency (TF-IDF) technique was utilized to convert the cleaned and stemmed plot summaries into numerical vectors. TF-IDF not only considers the frequency of a word in a particular document but also offsets this frequency against the number of documents containing the word. This results in a matrix where each movie is represented as a vector of numbers, capturing the essence of its plot in relation to other movies.

movie_name	processed_plot
Toy Story (1995)	toy stori american media franchis own the walt disney company. it center toy that, unknown humans, secret living, sentient creatures. it began 1995 releas anim featur film name, focus divers group toy featur classic cowboy doll name sheriff woodi modern spaceman action figur name buzz lightyear. the toy stori franchis consist main five cgi-anim featur films: toy stori (1995), toy stori 2 (1999), toy stori 3 (2010), toy stori 4 (2019), spin-off prequel film within film lightyear (2022). a fifth film recent announced. it also includ 2d-anim direct-to-video spin-off film within film buzz lightyear star command: the adventur begin (2000) anim televis seri buzz lightyear star command (2000–01) follow film. the first toy stori first feature-length film made entir use computer-gener imagery. the first two film direct john lasseter, third film lee unkrich (who act co-director second film alongsid ash brannon), fourth film josh cooley, lightyear angus maclane. produc total budget $720 million, toy stori film gross $3.3 billion worldwide, becom 20th highest-gross film franchis worldwid third highest-gross anim franchise. each film main seri set box offic records, third fourth includ top 50 all-tim worldwid films.
Jumanji (1995)	jumanji 1995 american urban fantasi adventur film direct joe johnston screenplay jonathan hensleigh, greg taylor, jim strain, base 1981 children pictur book name chris van allsburg. the film first instal jumanji film series. it star robin williams, kirsten dunst, david alan grier, bonni hunt, jonathan hyde, bebe neuwirth. the stori center supernatur board game releas jungle-bas hazard upon player everi turn take. jumanji releas decemb 15, 1995, soni pictur releasing. the film receiv mix review critics, box-offic success, gross $263 million worldwid budget approxim $65 million. it tenth-highest-gross film 1995. the film spawn anim televis series, air 1996 1999, follow spin-off film, zathura: a space adventur (2005), two indirect sequels, jumanji: welcom jungl (2017) jumanji: the next level (2019). == plot == in 1969, alan parrish live parents, sam carol, brantford, new hampshire. one day, escap group bulli retreat sam shoe factory.

Table 8.1.1: Dataframe Showing the Processed Movie Plot Summaries

8.1.2 User Data (Age and Gender)

- Gender Encoding: Gender, being a categorical variable, was encoded to a numerical format. 'Male' was represented by 0 and 'Female' by 1, facilitating its use in mathematical computations.

- Age Normalization: Age, a continuous variable, was normalized to ensure it's on a similar scale as other features. This step is crucial to ensure that no particular feature disproportionately influences the model due to its scale.

The meticulous preprocessing of these features set the foundation for the subsequent steps in content-based filtering, ensuring that the data fed into the models was of the highest quality.

8.2 Implementation of Content-Based Filtering Algorithms

Content-based filtering, at its core, revolves around the idea of using item attributes or user attributes to recommend additional items similar to what the user likes, based on their previous actions or explicit feedback. In this study, two primary approaches were explored:

8.2.1 Item-based Approach Using Movie Plots

- Cosine Similarity Matrix: With the movie plots vectorized using TF-IDF, a cosine similarity matrix was computed. This matrix captures the cosine of the angle between two vectors, representing how similar two movies are in terms of their plots.

- KNN with Movie Plots: The K-Nearest Neighbors (KNN) algorithm was employed using the cosine similarity matrix. Two variations were explored:

    a. Using the complete TF-IDF matrix.
    b. After feature selection, where certain terms like names were removed to see if it improved recommendation quality.

8.2.2 User-based Approach Using User Data:

- Cosine Similarity Matrix for Users: A cosine similarity matrix was also computed for users based on their age and gender. This matrix captures the similarity between users.

- KNN with User Data: The KNN algorithm was applied again, but this time using the user similarity matrix. This approach aimed to find users who are similar and then recommend movies based on what those similar users liked.

Both these approaches, item-based and user-based, offer unique perspectives. While the item-based approach focuses on the content of the items (movies in this case), the user-based approach leverages user attributes to find similarities.

8.3 Evaluation

8.3.1 Metrics Comparison

Algorithm	RMSE	Hit Rate (%)
Normal Predictor (Baseline)	1.5047	1.32
Movie Plots using the complete TF-IDF matrix (Item-Based)	0.9087	0.68
Movie Plots after Feature Selection (Item-Based)	0.9136	0.69
User Age+Gender (User-Based)	0.9939	0.05

Table 8.3: RMSE & Hit Rate of the Implemented Algorithms (Part 2)

Movie Plots (Item-Based): This method, which utilizes movie plots for recommendations, offers a balanced approach. Using the complete TF-IDF matrix, it achieved a hit rate of 0.68% and an RMSE of 0.9087. This indicates its capability to provide relevant recommendations based on content similarity.

Movie Plots with Feature Selection (Item-Based): By applying feature selection to the movie plots, there was a slight improvement in the hit rate, which was recorded at 0.69%. However, this came at the cost of prediction accuracy, with an RMSE of 0.9136. This suggests that while feature selection might help in improving relevance, it might not always enhance prediction accuracy.

KNN (User-Based with User Data): This method, which incorporates user data into the KNN algorithm, achieved a significantly lower hit rate of 0.0497%. Its RMSE was 0.9939. The results suggest that relying solely on user attributes might not be as effective for recommendations, especially when used in isolation.

8.3.2 Further Analysis

In the realm of content-based filtering, using movie plots as features yielded promising results. The method that utilized the complete TF-IDF matrix of movie plots achieved a commendable balance between prediction accuracy (RMSE) and recommendation relevance (Hit Rate). However, when feature selection was applied to the movie plots, there was only a marginal improvement in the hit rate, suggesting that the complete set of features in the TF-IDF matrix might be essential for optimal performance.

The user-based KNN approach that incorporated user data (age and gender) into the recommendation process did not fare as well. Its hit rate was significantly lower, indicating that relying solely on user attributes might not be sufficient for making relevant recommendations.

In conclusion, while content-based filtering methods offer a unique approach to recommendations by leveraging item or user attributes, their performance can vary based on the features used and the algorithms employed. The comparative analysis underscores the importance of selecting the right combination of features and algorithms to achieve optimal recommendation results

Chapter 9 Conclusion

The study began with a meticulous exploratory data analysis, setting the stage for subsequent investigations. The collaborative filtering approach, particularly the SVD algorithm, showcased its prowess by outperforming other methods in both prediction accuracy and recommendation relevance. On the other hand, the content-based filtering methods, while offering a unique perspective by leveraging item or user attributes, demonstrated varied performance based on the features and algorithms employed.

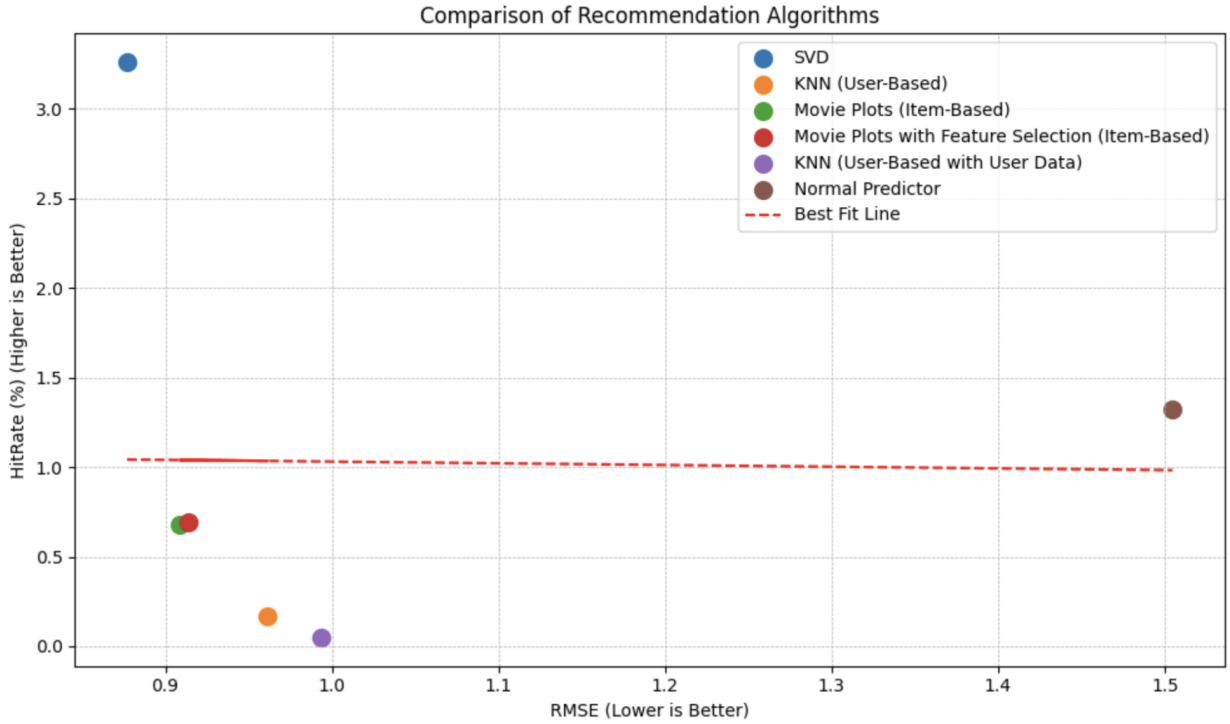

Figure 9.0: Scatter Plot Comparing the Performace of all the Recommendation Algorithms

Considering the the position of each recommendation algorithm and the best-fit line in the scatter plot visualized in Figure 9.0, it is evident that the ideal method is most likely going to be a hybrid one, harnessing the best of both Collaborative and Content-based techniques.

A few key takeaways from the study include:

- The significance of data preprocessing and feature extraction in shaping the performance of recommendation algorithms.
- The SVD algorithm's superiority in the collaborative filtering realm, both in terms of predicting user ratings and recommending relevant movies.
- The potential of content-based filtering, especially when using rich features like movie plots. However, the importance of feature selection was also underscored, as it can influence the recommendation quality.
- The challenges and limitations of relying solely on user attributes for recommendations, as evidenced by the lower hit rate of the user-based KNN approach.

9.1 Futurework

As with any research, this study has its limitations and areas for improvement. Incorporating these advancements and improvements in future studies can further improve recommendation systems, offering users an even more tailored and enriched experience.

1. Enhanced Web Scraping: The current study utilized a foundational approach to unpack the MovieLens 1M dataset. However, there's potential to enrich this dataset further. One avenue to explore is the integration of movie posters through advanced web scraping techniques. Visual elements like posters can offer a unique perspective, potentially aiding in the development of more visually-informed recommendation systems. Incorporating such visual data could pave the way for more interactive and visually appealing recommendations, enhancing user engagement and experience.

2. Dimensionality Reduction: As datasets grow in size and complexity, the curse of dimensionality becomes a pressing concern. In this study, while various features were extracted and utilized, the potential of dimensionality reduction techniques was not fully explored. Techniques such as Principal Component Analysis (PCA) or t-SNE could be employed in future works to reduce the feature space while retaining the most critical information. This not only can improve the efficiency of algorithms but also potentially enhance the accuracy of recommendations by focusing on the most salient features.

3. Hybrid Approaches and Deep Learning: While this study delved deep into collaborative and content-based filtering methods, there's a vast landscape of recommendation algorithms yet to be explored. A promising direction is the hybrid approach, which combines the strengths of both collaborative and content-based methods. Moreover, the advent of deep learning offers exciting possibilities. Neural networks, especially architectures like autoencoders or recurrent neural networks, could be employed to capture intricate patterns in the data, potentially leading to more accurate and personalized recommendations.

Appendix

The Source Code is available at this GitHub link

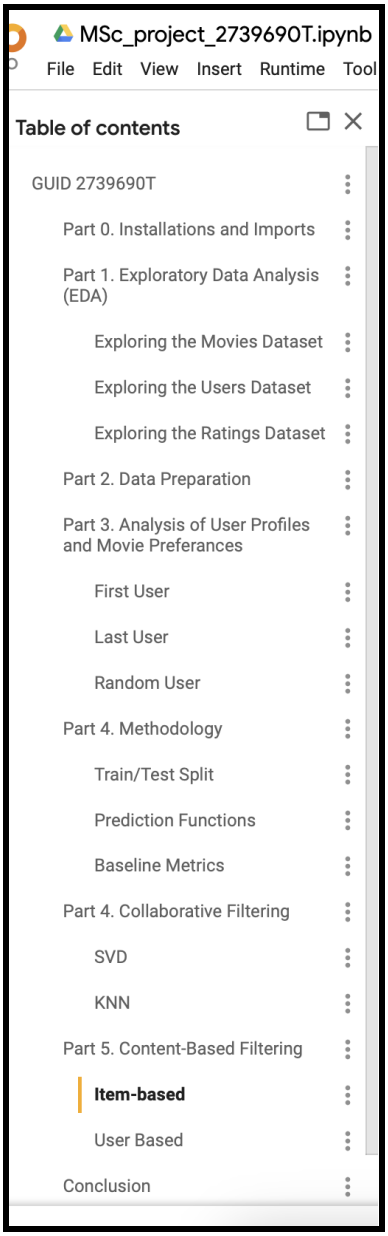

Figure A.0: Table of Contents of the Google Colab Notebook

Bibliography

[1] Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. In Recommender Systems Handbook (pp. 1-35). Springer, Boston, MA. DOI: 10.1007/978-0-387-85820-3_1.

[2] Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM, 35(12), 61-70.

[3] Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. In The adaptive web (pp. 325-341). Springer, Berlin, Heidelberg.

[4] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction, 12(4), 331-370.

[5] Mello, Arthur. “How Do Netflix and Amazon Know What I Want?” *Medium*, Towards Data Science, 17 Mar. 2020, towardsdatascience.com/how-do-netflix-and-amazon-know-what-i-want-852c480b67ac.

[6] Spoorthy G., Sanjeevi Sriram G. (2021). "Hybrid model for movie recommendation system using content K-nearest neighbors and restricted Boltzmann machine". [Link](https://dx.doi.org/10.11591/IJEECS.V23.I1.PP445-452)

[7] S. Fong, Yvonne Ho, Yang Hang (2008). "Using Genetic Algorithm for Hybrid Modes of Collaborative Filtering in Online Recommenders". (https://dx.doi.org/10.1109/HIS.2008.59)

[8] M. Muhammad (2021). "User-Based Collaborative Filtering Using Agglomerative Clustering on Recommender System".(https://dx.doi.org/10.4108/eai.17-7-2021.2312410)

[9] Tianyu Li, Yukun Ma, Jiu Xu, B. Stenger, Chen Liu, Yu Hirate (2018). "Deep Heterogeneous Autoencoders for Collaborative Filtering". [Link](https://dx.doi.org/10.1109/ICDM.2018.00153)

[10] Kyung-Yong Chung (2008). "Personalized Item Recommendation using Image-based Filtering". [Link](https://dx.doi.org/10.5392/JKCA.2008.8.3.001)

[11: J. Salter, N. Antonopoulos (2006). "CinemaScreen recommender agent: combining collaborative and content-based filtering". (https://dx.doi.org/10.1109/MIS.2006.4)

[12] Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 52(1), 1-38.

[13] Chen, C., Zhang, M., Liu, Y., & Ma, S. (2020). Neural attentional rating regression with review-level explanations. Proceedings of the World Wide Web Conference.

[14] Covington, P., Adams, J., & Sargin, E. (2016). Deep neural networks for YouTube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 191-198).

[15] Álvaro González, Fernando Ortega, Diego Pérez-López, Santiago Alonso (2022). "Bias and unfairness of collaborative filtering based recommender systems in MovieLens dataset". (https://dx.doi.org/10.1109/ACCESS.2022.3186719)

[16] F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS) 5, 4, Article 19 (December 2015), 19 pages. DOI=http://dx.doi.org/10.1145/2827872

[17] Rapaport, Elad. “Movielens-1m Deep Dive - Part I.” Medium, Towards Data Science, 8 June 2022, towardsdatascience.com/movielens-1m-deep-dive-part-i-8acfeda1ad4.

[18] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection was conducted. Proceedings of the 14th international joint conference on Artificial intelligence - Volume 2, 1137-1143.

[19] Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection was presented. Statistics surveys, 4, 40-79.

[20] Cha, Gi-Wook & Moon, Hyeun & Kim, Young-Min & Hong, Won-Hwa & Hwang, Jung-Ha & Park, Won-Jun & Kim, Young-Chan. (2020). Development of a Prediction Model for Demolition Waste Generation Using a Random Forest Algorithm Based on Small DataSets. International journal of environmental research and public health. 17. 10.3390/ijerph17196997.

[21] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 22(1), 5-53.

[22] Chai, T., & Draxler, R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. Link to the paper. Publication ID: 10.5194/GMD-7-1247-2014.

[23] Gunawardana, A., & Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. Journal of Machine Learning Research, 10(Dec), 2935-2962

[24] Cremonesi, P., Koren, Y., & Turrin, R. (2010, April). Performance of recommender algorithms on top-n recommendation tasks. In Proceedings of the fourth ACM conference on Recommender systems (pp. 39-46)

[25] Ricci, F., Rokach, L., & Shapira, B. (2015). Recommender systems: introduction and challenges. In Recommender Systems Handbook (pp. 1-34). Springer, Boston, MA

[26] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, (8), 30-37.

[27] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 22(1), 5-53.

[28] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web (pp. 285-295).

[29] Hug, N. (2017). Surprise: A Python scikit for building and analyzing recommender systems. Available at: http://surpriselib.com

[30] Koren, Y., Bell, R., & Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, 42(8), 30-37

[31] Aha, D. W., Kibler, D., & Albert, M. K. (1991). Instance-based learning algorithms. Machine learning, 6(1), 37-66.

[32] Dr. Ganesh, D., & Bhansali, Y. (2023). Movie Recommendation Systems Using Content-Based Filtering. Publication ID: 10.56726/irjmets42626.

[33] Iliopoulou, K., Kanavos, A., Ilias, A., Makris, C., & Vonitsanos, G. (2020). Improving Movie Recommendation Systems Filtering by Exploiting User-Based Reviews and Movie Synopses. Publication ID: 10.1007/978-3-030-49190-1_17.

[34] Singh, K. (2021). A Multi-Criteria Movie Recommendation System based on User Preferences and Movie Features. Publication ID: 10.17762/msea.v70i1.2317.