Erasmus University Rotterdam
Erasmus School of Economics
Master Thesis Data Science and Marketing Analytics

Improving Hybrid Recommendation Systems for Solving the Cold Start User Problem.

Tycho Bakker (506027)

Supervisor:	dr. A Alfons
Second assessor:	prof.dr. ACD Donkers
Date final version:	13th October 2023

Abstract

Recommendation systems are systems that are increasingly used by companies and within more and more different industries. Almost everyone encounters them daily, such as on social media or when choosing a product online. The systems are created by utilizing data filtering which has seen its rise. With the great interest in recommenders, the need for research and improvement is undeniable. This paper examines one of the biggest problems with recommenders, namely Cold Start (CS) users. CS users are those new to the system with no or little available data. This research proposes a dynamic weighted combination of two recommendation models to solve this problem. The combination is weighted with a rule based on a user's available movie ratings. The proposed method performs slightly better than the two models individually. This paper also zooms in on the change at low amounts of available ratings and finds that an improvement becomes apparent only from 15 ratings onward. Finally, it looks at different users by genre and finds that the system does not yet optimally adapt to the differences between groups of users.

Contents

Chapter 1

Introduction

Recommendation systems (RSs) have become increasingly prevalent in various industries, including entertainment, e-commerce, and social media. There are a lot of benefits of RSs that exist today. They provide a quick, efficient, automated process to give recommendations, they can be highly personalized, increase engagement, boost sales, and increase customer satisfaction. Probably the most known recommenders are present on social media platforms like Instagram, YouTube, and TikTok. These RSs also bring up a lot of concerns about for example body image (Liu, 2021), discrimination (Amarikwa, 2023) or addictions (Petrescu & Krishen, 2020) as a result of the algorithms. The negative side of this system can be addressed by ethical consideration and improvements in the algorithmic design.

With the rise of interest in recommendation methods, some problems in making recommendations will occur. In this paper, one of the biggest problems of RSs will be tackled, namely the 'cold start' problem. Cold start (CS) problems occur with the lack of information. It is a problem that is most common in collaborative filtering methods. In a paper called "Facing the cold start problem in recommender systems" (Lika et al., 2014) this problem is broadly discussed. They divide the problem into three categories. The first is recommendations for new users, the second recommendations for new items and the third is a combination of the two. This specific paper puts the focus on the user side, but there are a lot of other papers focusing on the item side.

The relevance of research on the CS problem is more than the scientific or practical implications. There is a wide range of stakeholders, including researchers, businesses, consumers, and policymakers. This paper provides a valuable contribution to the existing knowledge about RSs in the interest of researchers. It sheds light on one of the biggest challenges and proposes a solution to improve accuracy. For the fast-increasing number of businesses that rely on recommendation systems, the relevance lies in its potential to enhance the accuracy and effectiveness of the systems. In papers that face the CS problem like the one mentioned above (Lika et al., 2014), the authors understood the need of tackling this problem, because of the significant effect on the performance of RSs. By addressing the CS problem, businesses give better, more personalized recommendations at the start of a customer journey. This results in customers that feel more valued and get more personalized recommendations. These customers are more likely to convert, to become loyal customers, and eventually, the business will have increased sales. In addition to that, consumers who receive improved recommendations could save time, discover new products and services, and make better-informed decisions. So consumers also benefit from research towards solving the CS problem.

Finally, this research could also interest policymakers, as the RSs' ethical concerns continue to arise. By addressing the CS problem, policymakers can make sure this problem does not exacerbate the existing ethical problems like discrimination and addiction, and promote a more ethical algorithmic design. RSs that suffer from the CS problem may rely on default or generalized assumptions, which could result in these discriminating outcomes. Personalizing from the start could also help solve the addiction of customers when used in the correct way. Policymakers could prioritize user well-being and discourage addictive content from the beginning, set personal limits when the risk of addiction is present, and promote well-being and balance.

There has been extensive research conducted within the field of Recommendation Systems. The most known and researched solution for most of the common problems is the "Hybrid Recommendation System". Within hybrid RSs, the strengths of multiple recommendation strategies are combined in different ways. In "Hybrid recommender systems: A systematic literature review" (Çano & Morisio, 2017) a comprehensive review of existing literature in the field of Hybrid RSs is presented. There is an overview of the most relevant studies, the problems, and challenges that are faced by the researchers, which data mining and machine learning techniques are used, which datasets are used, and more. Out of the 76 papers that are reviewed, 23 tackle the cold start problem, 22 the data sparsity problem, and 16 the accuracy of the RSs. K-NN and clustering are the most used Machine Learning methods in the papers. The majority of the papers used movie data (MovieLens by Grouplens [1]) to conduct their research.

While the work discussed in this literature review has contributed to the development and implementation of hybrid recommender systems, one crucial aspect remains to be thoroughly researched: optimizing weights for content-based and collaborative filtering to provide the best recommendations for CS users. Despite the numerous advances in hybrid models, this specific research question remains unanswered in the current literature. Therefore, it is necessary to explore this topic further to determine how these weights can be effectively optimized. This study attempts to answer the research question:

[1]grouplens.org collected multiple datasets called MovieLens that exist of millions of ratings and tagging activities since 1995

How can the weightings for content-based filtering and collaborative filtering be optimized to enhance recommendation accuracy for cold start users?

To address and research this question, several sub-questions are formulated. First, the performance of the hybrid recommendation system needs to be investigated. The hypothesis is that hybrid recommendation systems outperform content-based or collaborative filtering in addressing the CS problem. This is essential for the research because if the hybrid recommender does not top a single filtering method, it does not make sense to use a hybrid system. Existing research compared the performance of the three options above, but not with the specific case of CS users. To investigate the performance, the following sub-question is formulated:

Can the hybrid recommendation system outperform content-based filtering and collaborative filtering alone in addressing the cold start problem?

Second, this research explores how the performance of the hybrid system varies with the amount of data available for new users. This sub-question is crucial because the performance of the system is likely to depend on the availability of data, and understanding the relationship is essential for improving the system and future practical implementation. Previous research focused on this topic within recommendation systems, but this research will specifically focus on the hybrid approach. To investigate this matter, the following sub-question is posed:

How does the performance of the hybrid system vary with the amount of data available?

The investigation into optimizing the hybrid recommendation system for different types of users is a crucial aspect of this research. This question holds significance because user preferences can vary significantly, and tailoring the system to different user types can enhance its overall performance. What sets this research apart is its focus on delving into the nuances of different user types rather than solely concentrating on overall optimization.

To categorize users into different types, the preferred genres by a user are employed as a distinguishing factor. Analyzing the genres enables the identification of user patterns, which has been extensively studied in the context of music recommendation systems (Hu & Ogihara, 2011). Leveraging these patterns allows for personalized recommendations based on individual user characteristics and behaviors. In light of this, the following sub-question is formulated to guide the research:

Does the performance of hybrid recommender vary for different user types?

By addressing this sub-question, the recommender system checks to see if any improvements can be made for additional personalization of specific needs and patterns exhibited by various user types. The ultimate goal is to enhance the system's performance by tailoring recommendations to the unique characteristics of each user group.

Chapter 2

Literature Review

With the birth of the internet in 1983 not only did the world get access to endless amounts of information, but also an abundance of information to collect. This came with opportunities that needed to be investigated. Around 1990 a lot of research was done on information storage, processing, filtering, and other aspects of dealing with information. A good example is an article about the distinction between collecting and filtering of information (Belkin & Croft, 1992).

The rise of information literature has given way to the development of recommendation systems (RSs), which serve a vital role in suggesting content to users. RSs found their earliest applications in the realm of email systems. In 1992, the creation of "Tapestry" marked a pioneering recommender system that combined both content-based filtering and collaborative filtering (Goldberg et al., 1992). An influential paper titled "Recommender Systems" (Resnick & Varian, 1997) acknowledged Tapestry as the first RS. This paper explored five different RSs, examining their unique features and exploring their social implications.

Since the exploration of these five systems, the field of RSs has experienced significant growth, with substantial research conducted in the years following. The objective of this chapter is to comprehensively review relevant literature to address the main research question and its associated sub-questions, providing insightful perspectives into the domain of recommendation systems.

2.1 Filtering methods

Recommender systems (RSs) can enhance the user experience by providing personalized recommendations. Key to their functioning are filtering methods, which determine how much weight is given to different factors in the recommendation process.

The following sections provide an in-depth analysis of diverse filtering methods, covering their theoretical foundations, practical implementations, and impact on the recommendation process.

2.1.1 Content-based filtering

The emergence of filtering methods in response to the exponential increase in information gathering is a key aspect discussed in this literature review. In parallel with the growing usage of the internet, the research on content-based filtering (CBF) has also seen significant development. An example of a personalized RS utilizing CBF is the PRES (Personalized Recommender System), which takes advantage of one of CBFs benefits by recommending small, niche news articles that may not be popular (Meteren, 2000). However, the paper acknowledges that PRES, relying solely on CBF, is not an accurate recommender. One of the major reasons for this limitation is the ambiguity of terms, which can have multiple meanings. Additionally, this type of recommender fails to consider the user's future preferences. To address these challenges, the authors propose the integration of collaborative filtering into the PRES system to enhance its performance.

CBF can be implemented in various ways, but they all follow a common guideline. A typical CBF system comprises two primary data sources (Aggarwal, 2016). The first source is the description or features of the items within the recommender system. This could include item descriptions provided by the manufacturer, textual descriptions of movie content, or genre information associated with movies. These features serve as valuable indicators for determining item similarity and relevance.

The second data source in CBF is the user profile, which is constructed based on user feedback (Aggarwal, 2016). User feedback can be either implicit or explicit. Implicit feedback refers to the actions or behaviors exhibited by users, such as click-through rates, browsing history, or time spent on certain items. Explicit feedback, on the other hand, involves explicit indications of user preferences, such as ratings assigned to movies, which are used for building the model of this paper. By collecting and analyzing user feedback, CBF systems are able to understand user interests and preferences, and subsequently generate personalized recommendations.

It is important to note that while CBF has its strengths, such as the ability to provide recommendations for niche items and rely on item characteristics, it also has limitations. One of the major challenges is the reliance on item descriptions and features, which may not fully capture the user's preferences or the complexity of their interests (Lops et al., 2011). Additionally, CBF systems may suffer from the "filter bubble" effect, where users are recommended items similar to what they have interacted with in the past, potentially limiting their exposure to diverse content.

In conclusion, CBF has gained prominence in the field of RSs as a means to address the challenges posed by the vast amount of available information. The PRES system demonstrates the utilization of CBF for personalized recommendations while acknowledging its limitations and proposing the incorporation of collaborative filtering to improve accuracy. By considering item descriptions and user profiles, CBF systems can capture item similarity and user preferences, enabling personalized recommendations. However, challenges such as the ambiguity of terms and the lack of consideration for future preferences remain areas of ongoing research in the field.

2.1.2 Collaborative filtering

Collaborative filtering (CF) has been extensively researched and widely used in RSs due to its effectiveness in generating recommendations. CF is often incorporated as one of the methods in hybrid recommender systems, leveraging its strengths alongside other techniques.

One notable early application of CF can be seen in the work of the Grouplens project, which developed the Movielens datasets widely utilized in RS research (Konstan, 1997). The Grouplens team integrated CF into a news system called "Usenet", where they utilized the ratings provided by 250 users to make predictions. They observed that CF exhibited higher accuracy, scalability with large volumes of data, and fast-paced recommendation generation. However, challenges such as the sparsity of ratings from other users and the data scarcity for individual users were prominent issues. The system heavily relied on user input for optimal performance. These challenges remain major concerns in CF research. The issue of data availability, which will be further discussed in this literature review, forms the central focus of this study.

Over the years, numerous methods have been proposed and tested for CF. The methodology section of this paper provides a comprehensive overview of the specific methods employed, which can be further contextualized within the broader taxonomy of CF recommender systems (Papadakis et al., 2022). This taxonomy categorizes methods into memory-based and model-based approaches.

Memory-based methods rely on the direct usage of user-item interaction data, such as ratings or purchase history, to compute similarities or relationships between users or items (Papadakis et al., 2022). These methods include user-based CF, item-based CF, and their variants. User-based CF identifies similar users to make recommendations based on the preferences of other users with similar tastes. On the contrary, item-based (CF) operates by identifying similar items for recommendation based on a user's positive interactions with other items.

In contrast to memory-based methods, model-based methods employ statistical or machine-learning techniques to construct a model from the available data (Papadakis et al., 2022). These models capture underlying patterns, relationships, and preferences to generate recommendations. Among the model-based CF methods are matrix factorization, latent factor models, Bayesian models, and other advanced techniques. These methods aim to create robust models that can better understand the nuances of user preferences and item characteristics, thus leading to more accurate and personalized recommendations.

The literature review conducted by Papadakis et al. (2022) provides a valuable and up-to-date resource for understanding the various CF methods proposed. It offers insights into different techniques' strengths, limitations, and advancements, enabling researchers to choose appropriate methods for their specific contexts.

In summary, CF has been a central focus in RS research, and its integration into hybrid recommender systems has proven fruitful. The challenges of sparse data and user-specific scarcity remain important areas of investigation. By categorizing CF methods into memory-based and model-based approaches, researchers have been able to explore and compare different techniques effectively.

2.1.3 Weighted combination in Hybrid Recommendation Systems

Each of the known recommendation techniques possesses its unique strengths and weaknesses. In response to the desire for improved performance and to address the most prominent issues that may arise, researchers have explored the integration of techniques (Burke, 2002). This led to the development of hybrid recommendation systems (RSs), which generally fall into one of three categories (Kim et al., 2006): the linear combination model, sequential combination model, and mixed combination model. In the context of weighted combinations, the linear combination approach is commonly used. In this approach, both models in the hybrid RS make predictions simultaneously, and their predictions are then combined in a weighted manner. This method allows for a flexible and balanced integration of the different models, leveraging their respective strengths to improve the overall recommendation quality.

The advantage of the linear combination model is its simplicity and interpretability. It allows researchers and practitioners to control the influence of each recommendation technique by adjusting the assigned weights. This flexibility makes it possible to adapt the hybrid system to the specific requirements and characteristics of the recommendation domain.

Moreover, the linear combination model can take advantage of the complementary strengths of different techniques (Kim et al., 2006). For example, collaborative filtering methods excel at capturing user preferences based on historical interactions, while content-based filtering methods effectively consider item characteristics and user profiles. By combining these approaches, the hybrid system can potentially provide more accurate and diverse recommendations, addressing the limitations of the individual techniques.

However, the linear combination model also has challenges. Determining the optimal weights for each technique is a nontrivial task. The weights must be carefully assigned to ensure that the combined predictions reflect the strengths of the individual models.

Moreover, dealing with the potential differences in prediction scale and biases between the different techniques requires pre-processing steps or normalization techniques to achieve a fair combination.

2.2 Influence of data availability

The emergence and development of recommendation systems (RSs) have introduced several challenges, among which the cold start (CS) problem and data sparsity stand out as the most prominent issues (Çano & Morisio, 2017). Addressing and mitigating these problems have become key areas of interest for stakeholders involved in RS research and application.

The CS problem manifests itself in two distinct forms: the item cold start problem and the user cold start problem (Lam et al., 2008). The item CS problem arises when a new item, which lacks any ratings or relevant data, is introduced into the system. On the other hand, the user CS problem occurs when a new user joins the system and has not yet provided any ratings or preferences. While both sides of the CS problem present challenges, this paper specifically focuses on the less-researched issue of CS users.

Over the past two decades, researchers have proposed numerous solutions to address both the item and user CS problems. As early as 2002, a highly influential paper explored the item CS problem and proposed an aspect model latent variable method (Schein et al., 2002). The authors tested their approach on Naïve Bayes and several heuristic recommenders, inspiring further investigations and the development of subsequent models by researchers in the field. The item CS problem has been tackled through various strategies. Some approaches leverage content-based filtering, utilizing item features or descriptions to make initial recommendations for new items. Others incorporate demographic or contextual information to infer user preferences and make informed suggestions. Hybrid approaches that combine multiple techniques, such as content-based and collaborative filtering, have also been effective in addressing the item CS problem.

Regarding the user CS problem, researchers have explored techniques that involve knowledge transfer from existing users to new users (Lika et al., 2014). This can be achieved through group-based recommendations, where similar user profiles are identified, and recommendations are made based on the preferences of those similar users. Additionally, active learning methods, where the system actively seeks feedback from new users, have shown promise in mitigating the user CS problem. The advancements made in addressing the CS problem are closely tied to the availability and accessibility of data. As data availability has increased over the years, primarily driven by the expansion of online platforms and the internet, new opportunities have arisen for RSs to overcome the CS challenge (Melville & Sindhwani, 2010). The ability to collect and analyze diverse data types, such as user demographics, contextual information, and item features, has enhanced the effectiveness of recommendation algorithms in handling CS scenarios.

In conclusion, the challenges posed by the CS problem have spurred extensive research and the development of various approaches to mitigate its impact on recommendation systems. The availability of diverse data sources and the advancements in recommendation algorithms have led to significant progress in addressing both the item and user CS problems. The evolving data landscape continues to shape and influence the nature of recommendations, enabling more accurate and personalized suggestions for users, even in the absence of historical data.

2.3 User types in Hybrid Recommendation Systems

That people are different from each other is a long-known fact. With this difference, there is also a difference in people's needs. Recommendation systems (RSs) try to respond to those needs by making the best possible personalized recommendations. Therefore, understanding user differences is important. This importance within RSs has been noticed before (Knijnenburg et al., 2011). Identifying users based on different characteristics is key to getting the best performance.

Another paper found that movie preference is highly dependent on a user's personality (Golbeck & Norris, 2013). They found a correlation between personality traits and opinions about recommendations, how often they were used, and the ratings of items that were recommended to them.

The aim of this research is to investigate whether there are differences in recommendation-predicting performance between different user types based on genre preferences. The question arises if the hybrid recommender can effectively personalize recommendations for different user types or if improvements are needed.

Chapter 3

Data

3.1 Dataset

In this research, a combination of content-based filtering (CBF) and collaborative filtering (CF) is employed, which requires the use of different types of variables. The data utilized for this study is sourced from Grouplens (Harper & Konstan, 2016), and it can be directly downloaded from the provided link: https://grouplens.org/datasets/movielens/. Specifically, the 25M dataset is utilized, comprising 25,000,095 movie ratings and 1,093,360 tag applications across 62,423 movies. This dataset covers the period from January 9, 1995, to November 21, 2019, and involves 162,541 users. To facilitate analysis, the data is segregated into distinct datasets, with this research focusing on the movies and ratings datasets.

Due to the demanding computational power required to process a model on 25 million ratings, a practical approach involves conducting tests using a 2 percent sample, which results in approximately 480 thousand ratings.

Employing this smaller dataset for testing makes it feasible to evaluate and compare the model's performance without overwhelming the computational resources. This streamlined approach ensures valuable insights and results can still be obtained while maintaining a reasonable computational load.

The movie's dataset contains various features, including the movie title, release year, and genre. Meanwhile, the rating dataset presents the movie ratings alongside corresponding user and movie IDs. By utilizing the movie ID as a common identifier, the movie features can be effectively joined with the rating dataset. For content-based filtering (CBF), the genre variable assumes crucial importance. This content feature is employed in making recommendations, taking into account the genre preferences of users. On the other hand, collaborative filtering (CF) primarily relies on the rating and movie ID variables to generate personalized recommendations.

3.2 Data exploration

To gain initial insights into the data, an exploratory analysis was conducted, starting with the distribution of ratings. The ratings in the dataset are either whole numbers or half numbers. Figure 3.1 presents a visualization of the rating distribution, indicating a left-skewed pattern with a higher frequency of high ratings.

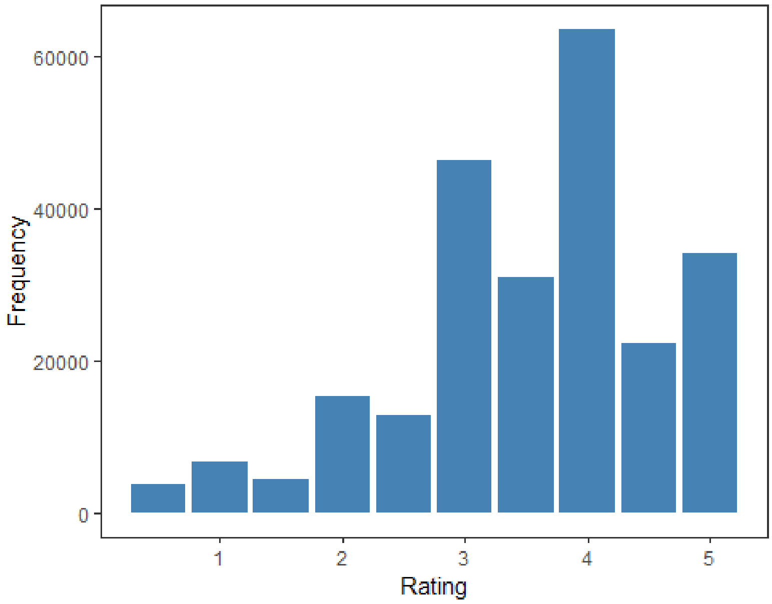


Figure 3.1: Distribution of Ratings

Table 3.1 displays the genre statistics, encompassing both the count of movies per genre and the corresponding average rating for each genre. This table provides valuable insights into the distribution of movies across different genres and sheds light on the average rating received by movies within each genre. The average ratings across genres show a relatively small variation, indicating a similarity in overall ratings. However, when examining individual genres, distinct differences emerge. The genres with the highest average ratings are Film-Noir, War, and Documentary. The genres with the lowest average ratings are horror, movies without a specific genre, and comedy. These findings imply a potential disparity in the quality or appeal of movies across different genres.

Table 3.1: Genre Statistics

		Ratings	
Genres	**Number of Movies**	**Average Rating**	**Standard Deviation**
Drama	25606	3.68	1.00
Comedy	16870	3.42	1.08
Thriller	8654	3.52	1.04
Romance	7719	3.54	1.05
Action	7348	3.47	1.07
Horror	5989	3.29	1.14
Documentary	5605	3.71	1.02
Crime	5319	3.69	1.01
No_genres	5062	3.33	1.16
Adventure	4145	3.52	1.07
Sci-Fi	3595	3.48	1.09
Children	2935	3.43	1.10
Animation	2929	3.61	1.04
Mystery	2925	3.67	1.01
Fantasy	2731	3.51	1.09
War	1874	3.79	0.99
Western	1399	3.59	1.02
Musical	1054	3.55	1.06
Film-Noir	353	3.93	0.91
IMAX	195	3.60	1.05
Total	62423	3.53	1.06

Note: The total number of movies is lower than the sum in the "Number of Movies" column due to movies having multiple genres, resulting in a double count for those movies.

In Figure 3.2, the presented graph depicts the popularity trends of the 5 most popular genres over time. The popularity is measured based on the number of ratings per year, providing insights into the audience engagement with each genre. Notably, the results reveal a peak in popularity for all genres after the year 2000.

The x-axis represents the timeline spanning several years, while the y-axis denotes the number of ratings per release year, serving as a proxy for genre popularity. The graph captures the preferences of audiences and the evolving landscape of the entertainment industry.

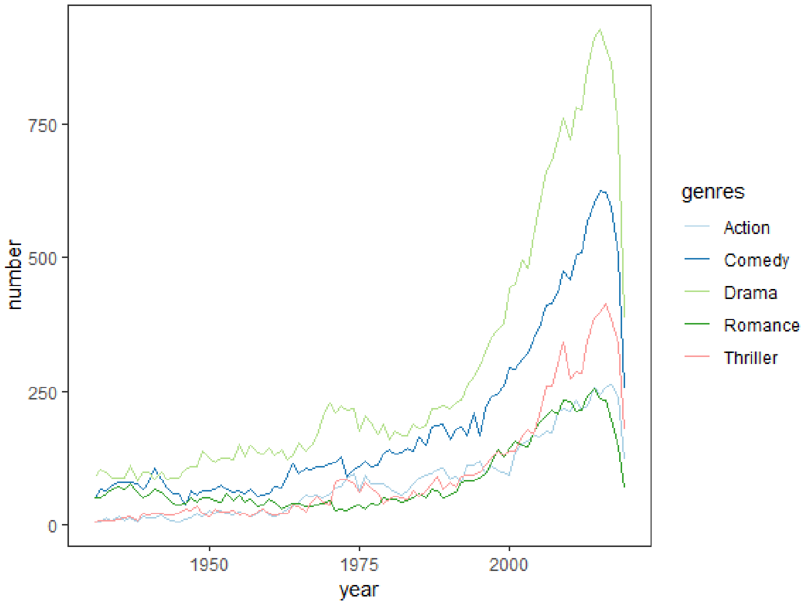


Figure 3.2: Top 5 Genre Popularities Over the Years

Chapter 4

Methodology

Content-based and collaborative filtering are two of the most widely used filtering methods within hybrid recommendation systems (B.Thorat et al., 2015). For building hybrid recommenders, the combination of these two is also the most common hybrid approach.

4.1 Content-based Filtering

Content-based filtering (CBF) is a technique based on the content of an item. It filters based on similarities of the features of an item a user interacted with or liked (Geetha et al., 2018). In the "Recommender Systems Handbook" (Lops et al., 2011) a difference is being made between finding similarities in CBF on the keywords and finding similarities based on the semantics. The authors present two types of semantic indexing techniques: top-down and bottom-up. CBF is simpler than collaborative filtering and consists of only 3 steps:

1. Feature extraction: identifying relevant features or characteristics of the items.
2. User profile creation: based on the features of the items the user has rated positively in the past, a user profile is created that captures a user's preferences for different features.
3. Recommendation generation: Using the user profile and the features of the remaining items to compute a similarity score between the user profile and the items.

Most similar items are recommended to the user. In the specific case of movie data, these features include genre, description, cast, release year, and more. CBF is good at recommending niche or long-tail items that have few ratings or interactions in the system because it compares item features instead of for example user interactions. The downside is the problem of overspecialization, where there is a lack of diversity in the recommendations and they become too similar to a user's past interactions.

4.1.1 Construction of the Content-Based Filtering Model

In this research, the movie genres are utilized as features to measure similarity among movies. A genre movie matrix is constructed, comprising 20 distinct genres as columns and movie IDs as rows. Each movie in the matrix is represented by a binary value: a "1" indicates the presence of a particular genre in the movie, while a "0" indicates its absence. This binary representation allows for quantitative analysis of genre similarities between movies. To generate personalized recommendations, a user profile is created. This profile is calculated with the formula:

$$\text{User Profile} = \text{User Movies} \cdot \text{User Ratings}. \tag{4.1}$$

The dot product combines the genre information of the watched movies with the user's ratings, resulting in a weighted score for each genre in the user profile. The next step is to calculate the scores for each movie of interest by replacing the score where the movie includes a genre with the genre score of the user profile. The rankings for each movie are calculated with:

$$\mathrm{R_M} = \frac{\sum \text{Genres Scores}_\mathrm{M}}{\sum \text{User Profile}}, \tag{4.2}$$

where the R is the ranking for each movie (M).

To compare the rankings obtained from the original ratings and merge them with the predictions from collaborative filtering, it is necessary to normalize the rankings. Normalization ensures that the rankings are on a consistent scale, allowing for meaningful comparisons and combinations.

To identify the optimal CBF recommender, various normalization techniques are being evaluated. The objective is to maintain the consistency of ranking distributions while comparing actual ratings with the predictions generated by CBF. The initial method under consideration is Min-Max Normalization (Patro & Sahu, 2015), which facilitates a linear transformation of the original data range of the ratings. The formula for this is the following:

$$\mathrm{R_{M\ normalized}} = \frac{\mathrm{R_M} - \mathrm{R_{min}}}{\mathrm{R_{max}} - \mathrm{R_{min}}} \times (\mathrm{TR_{max}} - \mathrm{TR_{min}}) + \mathrm{TR_{min}}. \tag{4.3}$$

Here, R is the ranking of the CBF recommender of movie M, TR is the actual rating, min is the minimum value and max is the maximum value.

Additionally, multiplication factor normalization, which involves a trial-and-error process, is proposed by this research and explored. By systematically adjusting the multiplication factor, the impact on performance metrics can be observed. In essence, this approach increases the scale of the ranking with the multiplication factor.

$$\mathrm{R_{M\ normalized}} = \mathrm{R}_M \times \mathrm{MF}, \tag{4.4}$$

where R is the ranking of the CBF recommender of movie M and MF is the multiplication factor, which is adjusted until the most suitable results are achieved.

4.2 Collaborative Filtering

Collaborative filtering (CF) is likely the most researched filtering method. In a recent paper: "Collaborative filtering recommender systems taxonomy" (Papadakis et al., 2022) the authors give an overview of the approaches proposed in the entire research area of CF. The paper gives an overview of the types with two categories: memory-based and model-based. Memory-based (neighborhood-based) CF relies on calculating similarities between users or items. Typically it involves a similarity metric such as cosine similarity or Pearson correlation. It consists of two phases: preference similarity computation and predicting the rating of a target item based on neighbors who are similar users (Nam, 2022). Model-based CF exists to build a statistical model that can learn the relationships between users and items. The authors of the taxonomy also provide a table where the known drawbacks and the types of CF that have the best results for these drawbacks are shown. For the cold start (CS) problem, neural networks seem to have the best performance (Papadakis et al., 2022), but since this research is going to examine the best weighting of the hybrid approach for CS users, and since the neural networks model is going to over-complicate this research, neural networks will not be used as the CF method. Similarity scoring is one of the best interpretable methods within CF. Therefore, similarity scoring will be used in this research to conduct CF.

4.2.1 Construction of the Collaborative Filtering Model

The recommender system model utilized for collaborative filtering (CF) recommendations is implemented through the Recommenderlab package (Hahsler, 2022). This package requires the data to be structured as a user-item matrix, where the matrix captures the ratings that users have assigned to movies. The selection of movies to include in the matrix is based on a minimum threshold of available ratings. The user-item matrix is split into train, test, and validation sets to evaluate the three recommender model's performance. This split is conducted considering the available ratings for each user. Approximately 60% of the ratings are allocated to the train set, 20% to the test set, and the remaining 20% are assigned to the validation set. This division enables the assessment of the model's effectiveness in generating accurate predictions and recommendations on unseen data. The train set is used to train the CF model. The validation set is to validate and tune the model to get the best-performing model. Eventually, the test set is used to test the performance of our final model, test two proposed accuracy metrics, and test the sub-questions about the change in data availability and the different types of users.

The first step for creating recommendations is finding a neighborhood of similar users. This is done using a similarity measure. Within the collaborative filtering (CF) model of Recommenderlab, two similarity measures can be used. One of them is cosine similarity, which measures the similarity between two non-zero vectors in an inner product space (Li & Han, 2013). It calculates the cosine of the angle between the vectors, indicating the similarity in their orientations. The range of cosine similarity is from -1 to 1, with values closer to 1 indicating higher similarity. The formula for cosine similarity, using vectors A and B, is given as:

$$\text{Cosine Similarity}(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|}, \tag{4.5}$$

here $A \cdot B$ denotes the dot product of vectors A and B, and $\|A\|$ and $\|B\|$ represent the Euclidean norms of vectors A and B, respectively.

Another similarity measure is the Pearson correlation coefficient, also known as Pearson's r (Benesty et al., 2009). It assesses the linear correlation between two variables and indicates the strength and direction of their relationship. As in cosine similarity, the coefficient ranges from -1 to 1. The formula for the Pearson correlation coefficient, considering variables X and Y with n data points, is given as:

$$\rho(X, Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}, \tag{4.6}$$

where X_i and Y_i represent the individual data points of X and Y, respectively, and $\bar{X}$ and $\bar{Y}$ denote the means of X and Y, respectively.

Both Cosine similarity and Pearson correlation coefficient serve various applications. Cosine similarity is especially useful for measuring similarity between vectors in high-dimensional spaces, making it popular in tasks like text analysis and recommendation systems. On the other hand, the Pearson correlation coefficient finds extensive use in statistical analysis and data modeling for evaluating relationships between variables.

Once the neighborhood is established, the ratings of the neighborhood are aggregated to form a predicted rating for the user. This is done by averaging the ratings of the neighbors. To optimize the model's performance, parameters can be adjusted. The two most important parameters are the similarity method and "NN".

Tuning the similarity method is the choice between one of the methods explained above, Cosine or Pearson similarity. The difference in performance will be extensively tested and the best-performing similarity method will be used in building the Hybrid Recommender.

"NN" represents the number of neighbors taken into account when calculating a user's rating. The default value for NN is 25, meaning that ratings are predicted based on the ratings of the 25 most similar neighbors. To keep in line with the research goals and focus on improving the combination of CF and CBF rather than optimizing CF alone, the default value for NN remains unchanged.

4.3 Hybrid Recommendation Systems

Hybrid recommendation systems (HRSs) are available in different forms. A recommendation system (RS) is called a hybrid when there are at least two different filtering methods combined. The combining of methods can be done in different ways. The first one is a weighted combination (Burke, 2002). Here, both filtering methods make recommendations and are then combined using a weighted average. This approach is simple to implement and effective when both filtering methods are complementary. In most cases, the weighted method is used(Çano & Morisio, 2017).

The second method of combining is called a cascade (Burke, 2002). In this approach, one filtering method is used to make an initial set of recommendations and then the second method is used to filter these recommendations. This can be done by using CBF to make a first set, which is then filtered using CF.

The last method is switching (Burke, 2002). Here, the system switches between two methods of filtering, depending on the characteristics of the user and the item that is being recommended.

This paper proposes a novel approach to address the cold start (CS) user problem by introducing a weighted combination method. The main objective of the research is to optimize the weights associated with different recommendation techniques to determine the most effective combination that yields optimal performance. The weights are dynamically adapted based on the available data for each user.

A rule-based approach is employed to determine the combination of recommendation techniques, where the weights are adjusted dynamically depending on the number of available ratings in the train set. This adaptive strategy ensures that the weights are tailored to the specific user's data.

Table 4.1: Rules for Hybrid Recommender

Rule Multiplier	Available Ratings	CBF Weight	CF Weight
x	< 1x	0.9	0.1
x	1x - 2x	0.8	0.2
x	2x - 3x	0.7	0.3
x	3x - 4x	0.6	0.4
x	4x - 5x	0.5	0.5
x	5x - 6x	0.4	0.6
x	6x - 7x	0.3	0.7
x	7x - 8x	0.2	0.8
x	8x - 9x	0.1	0.9
x	9x >	0.0	1.0

Note: The rule multiplier is a factor that scales the required available ratings for each rule. As the multiplier increases, the threshold for utilizing collaborative filtering also rises, leading to more frequent usage of content-based filtering.

By implementing this dynamic weighting strategy, as proposed by this research and presented in Table 4.1, the aim is to overcome the challenges posed by CS users. The research endeavors to identify the most suitable weight combination for each user, ultimately leading to improved performance in predicting user preferences and generating personalized recommendations.

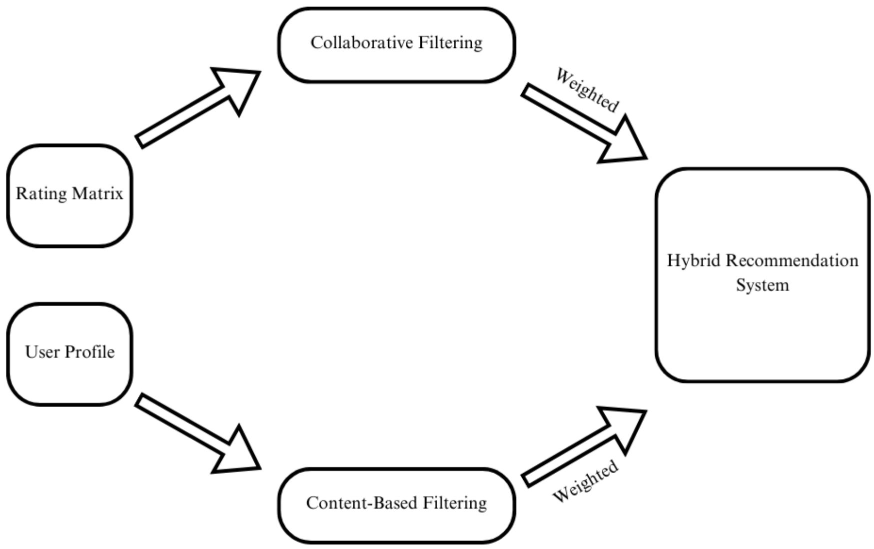


Figure 4.1: Weighted Hybrid Recommendation Model

In Figure 4.1, a comprehensive schematic walkthrough of the final model employed for this research is depicted. This flowchart explains the sequential process, starting from the input of the filtering methods, followed by the filtering itself, and finishing with the weighting process to integrate the outcomes within the hybrid recommendation system.

4.4 K-Means Clustering

To address the sub-question concerning the different types of users, clustering the users is necessary. Clustering is the most popular unsupervised learning technique (Patel & Thakral, 2016). Among various clustering techniques, this paper employs the K-Means clustering approach. In addition to being recognized as one of the most widely-used clustering methods (Ashabi et al., 2020), K-means is appreciated for its simplicity, speed, and straightforward implementation (Yuan & Yang, 2019). The K-means algorithm operates as follows (Likas et al., 2003):

1. **Initialization**: To start the clustering process, 'K' cluster centroids are selected from the data points.

2. **Assignment**: Once the initial centroids are defined, each data point is assigned to the nearest centroid based on the Euclidean distance metric.

3. **Update Centroids**: After the initial assignment, the centroids are recalculated as the mean of the data points within each cluster.

4. **Reassignment**: The assignment and centroid update steps are iteratively repeated until convergence or a predetermined number of iterations is reached.

5. **Final Clustering**: The outcome of the algorithm is a partitioning of the data into 'K' distinct clusters represented by their respective centroids.

The objective function of K-Means is given by:

$$J = \sum_{i=1}^{k} \sum_{j=1}^{n} \|x_j - \mu_i\|^2 , \tag{4.7}$$

where:

- J represents the objective function.
- k is the number of clusters.
- n is the number of data points.
- x_j represents a data point.
- μ_i represents the centroid of cluster i.

- $\|x_j - \mu_i\|$ represents the Euclidean distance between data point x_j and centroid μ_i.

To determine the optimal value of 'K,' the Elbow rule is utilized (Bholowalia & Kumar, 2014). The Elbow rule involves plotting the within sum of squares for varying numbers of clusters. The "Elbow" point in the plot indicates the value of 'K' where the within sum of squares starts to level off or form an elbow-like bend.

4.5 Evaluation metrics

Performance measurement is a crucial aspect of this research. To evaluate the performance of the collaborative filtering (CF) model, the rating matrix is initially split, as explained in the CF section mentioned earlier. This division allows the model to be trained on a training set, tested on a validation set for tuning the parameters of the models, and tested on the test set to assess its final performance. In the case of content-based filtering (CBF), the recommendation process differs slightly. As CBF relies on content features and user preferences for specific genres, a score can be calculated for each movie and user, effectively ranking the movies based on their relevance to the user's genre preferences.

4.5.1 Root Mean Squared Error & Mean Absolute Error

To measure performance, evaluation metrics are needed. The two most commonly used metrics for calculating the performance of predictive models are root mean squared error (RMSE) and mean absolute error (MAE).

RMSE measures the average deviation of the predicted values and the true values. It takes the square root of the average squared difference (Chai & Draxler, 2014). The formula of RSME is the following:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{4.8}$$

MAE on the other hand is also a good metric of performance. It calculates the average of the absolute difference between predicted and true values. The formula is the following:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{4.9}$$

4.5.2 Weighted Mean Squared Error

Since RMSE and MAE are symmetric evaluation metrics, which means that they treat overestimation and underestimation equally, this research is going to use a third, non-symmetric, evaluation metric. Overestimating a movie rating could lead to disappointed users. Getting an item recommended that you do not like is worse than getting an item not recommended that you would have liked. Therefore, this research proposes a weighted method to account heavily for overestimation. The current literature is not adequate when it comes to non-symmetric evaluation methods. An example of a paper that uses this method is the paper written by Almeida et al. (Almeida et al., 2018). The method proposed in this research uses the Mean Squared Error and adds a weight depending on whether the predicted value is higher or lower than the actual value.

$$\text{WMSE} = \frac{1}{n}\sum_{i=1}^{n}\begin{cases} \text{OW} \times (y_i - \hat{y}_i)^2 & \text{if } \text{OE}_i = 1 \\ \text{UW} \times (y_i - \hat{y}_i)^2 & \text{if } \text{UE}_i = 1. \end{cases} \tag{4.10}$$

Where:

- $WMSE$ = Weighted Mean Squared Error
- n = number of observations,
- y_i = actual value of the ith observation,
- $\hat{y}_i$ = predicted value of the ith observation,
- OE_i = binary variable indicating whether the prediction for the i observation is greater than the actual value,
- UE_i = binary variable indicating whether the prediction for the i observation is greater than the actual value,
- OW = weight for overestimations,
- UW = weight for underestimations.

4.5.3 Precision & Recall

In addition to these three evaluation metrics precision and recall are used as evaluation metrics. These metrics are binary evaluation metrics that rely on positive (recommend) and negative (not recommend) ratings. A movie is recommended to a user when the rating is 3 or higher. This means that a rating or predicted rating is positive if the rating is greater or equal to 3. In Table 4.2 presents the core principle of binary classifiers, which is crucial for calculating precision and recall.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Table 4.2: Confusion Matrix

Precision measures the proportion of true positives, in the case of movie recommendations the rightful recommended movies among all the movies that are recommended. In other words, it measures the accuracy of positive predictions.

$$\text{Precision} = TP/(TP + FP), \tag{4.11}$$

where TP is True Positives and FP is False Positives. Recall, also known as sensitivity or true positive rate, is a measure of how many relevant movies are selected. It calculates the proportion of true positive results among all the actual positive items. In other words, it measures the ability to find all the positive instances. Recall is computed using the formula:

$$\text{Recall} = TP/(TP + FN), \tag{4.12}$$

where TP is True Positives and FN False Negatives.

4.5.4 Weighted Accuracy

There are two methods introduced for calculating binary accuracy. Again, both methods are symmetric, where overestimation has the same weight as underestimation. For that, a non-symmetric variant is also proposed by this research for the binary metrics. This is the weighted accuracy where a weight is given to false negatives and false positives.

$$\text{WA} = (TP + wFN)/(TP + wFP + wFN), \tag{4.13}$$

where:

- WA = Weighted Accuracy
- TP = True Positives
- wFN = Weighted False Negatives
- wFP = Weighted False Positives.

Chapter 5

Results

5.1 Performance of the models

To address the research question regarding the effectiveness of the hybrid recommendation system in overcoming the cold start (CS) problem, it is essential to evaluate the performance of both content-based filtering (CBF) and collaborative filtering (CF) individually. For tuning the models, the validation dataset is used.

The first sub-question about whether the performance of the CBF and CF models is less than the combination in the hybrid recommendation system is an essential basis for this study and is addressed here.

5.1.1 Content-Based Recommender

In the context of content-based filtering (CBF), this recommendation model requires minimal adjustments. A score can be assigned to each user-movie pair, utilizing the user profile. The only modifiable aspect of CBF lies within the normalization of ranking values. To achieve optimal performance, various normalization methods are examined and compared.

Table 5.1: CBF Normalizations

Normalization Method	**RMSE**	**MAE**	**Precision**
Min-Max Real Rating	2.09	1.86	0.89
8 Multiplication	1.88	1.59	0.87
9 Multiplication	1.75	1.45	0.88
10 Multiplication	1.65	1.35	0.89

Table 5.1 presents the results of various normalization methods tested for evaluating rankings. Among the different approaches, it becomes evident that using a 10 times multiplication factor yields the most favorable outcomes in terms of both RMSE and

MAE. Consequently, the hybrid recommender will adopt the 10 times multiplication as the preferred normalization method.

However, it is worth noting that a higher multiplier leads to increased accuracy, but it also causes a significant shift in the distribution of predicted ratings. As a result, a majority of the predictions end up being the maximum possible value of 5. This seemingly improved accuracy might be attributed to the distribution of actual ratings, which tends to peak at higher ratings. The imbalance in the distribution could potentially explain the inflated accuracy achieved with higher multipliers.

To provide visual insights, Figure 5.1 showcases the distributions of the four normalization methods described in Table 5.1.

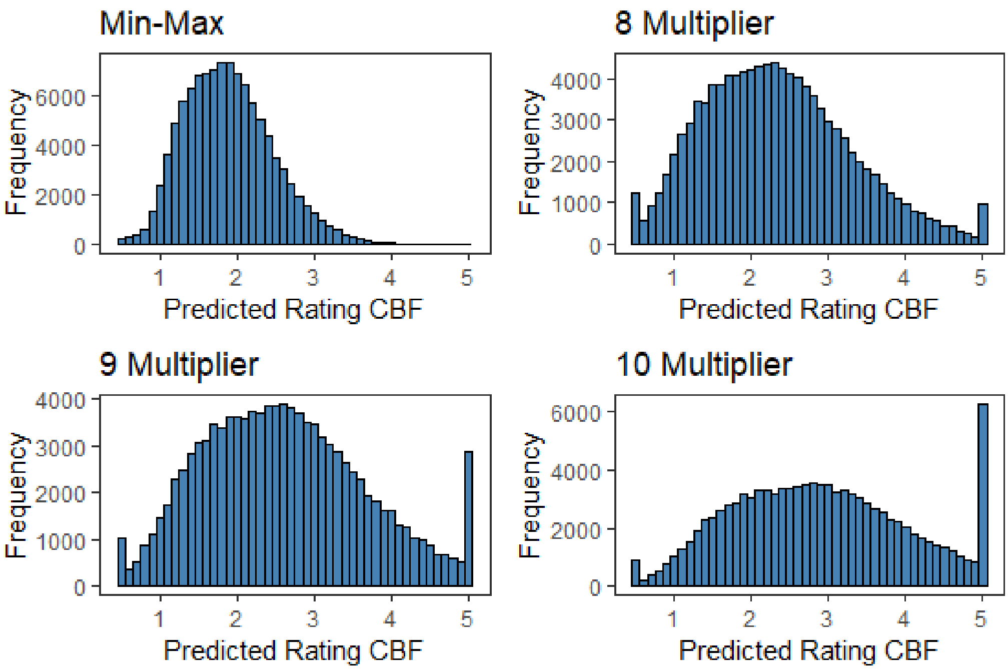


Figure 5.1: Distribution with different Normalizations

5.1.2 Collaborative Recommender

To achieve optimal results, one must focus on fine-tuning the parameters of the user-based collaborative filtering (CF) model. Among these parameters, the choice of a similarity measure holds the most significant influence over the CF model's performance. To this end, two built-in similarity measures, namely cosine similarity and Pearson correlation coefficient, have undergone thorough testing.

During the evaluation, both similarity measures were carefully analyzed to understand their strengths and limitations. This comprehensive assessment ensures that the CF model utilizes the most suitable similarity measure to deliver superior results. In Table 5.2 the comparison between the two similarity measures within the CF model is made. Both similarities have better results for RMSE and MAE, compared to the results of CBF in

Table 5.1. When comparing Cosine and Pearson, Pearson outperforms Cosine slightly. Therefore, Pearson similarity will be used in the hybrid model.

Table 5.2: CF Optimization

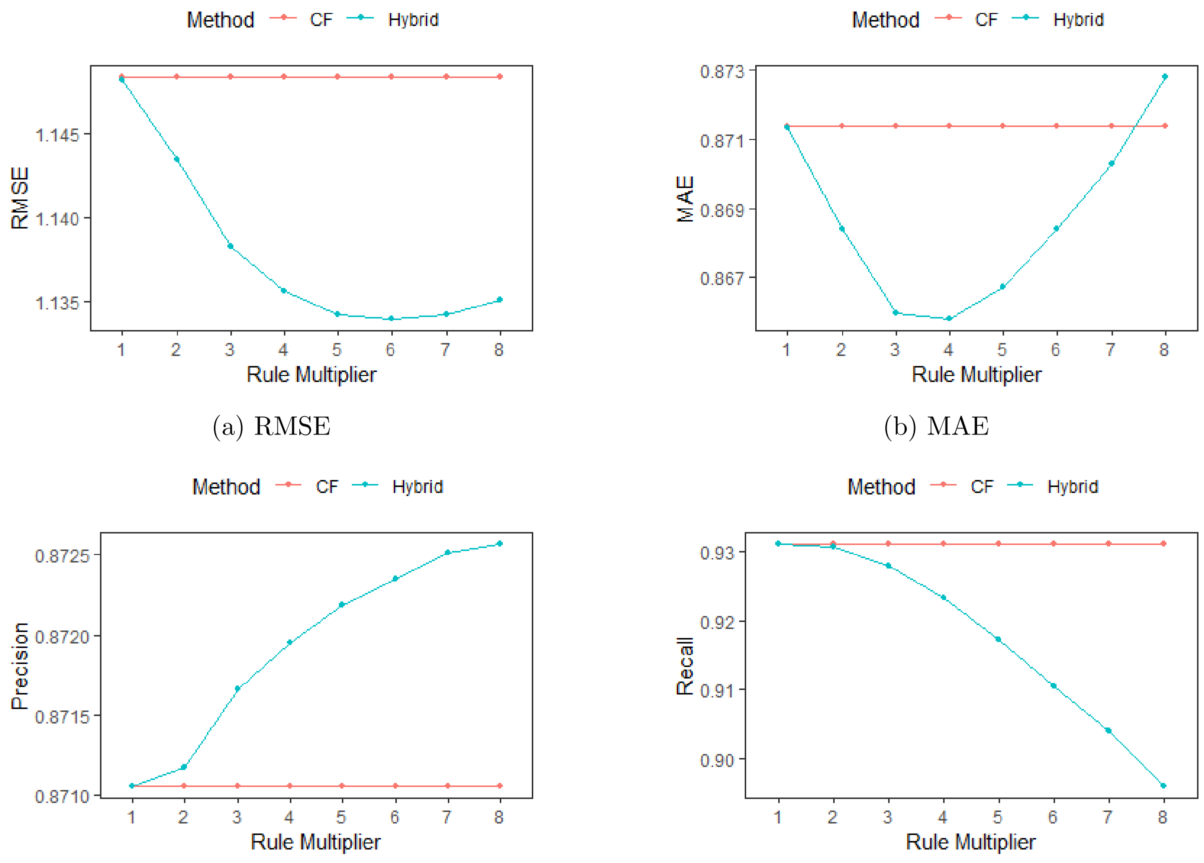
Similarity Measure	**RMSE**	**MAE**	**Precision**
Cosine	1.16	0.88	0.87
Pearson	1.15	0.87	0.87

The other parameters will be kept at their default values since the primary focus of this research is to optimize the combination of CF and CBF in a hybrid approach.

5.1.3 Dynamic Hybrid Recommender

Having established the CF and CBF models, the dynamic hybrid can now be constructed and evaluated. The existing rule governs the allocation of weights to each model based on the number of ratings available in the training data. The rule multiplier plays a crucial role in determining the balance between CBF and CF weights. By multiplying the values, it governs the extent to which the shift from CBF to CF occurs. When the rule multiplier is set to a higher value, it establishes a more stringent requirement for the number of ratings needed to increase the influence of CF while simultaneously reducing the impact of CBF.

The results of the four evaluation metrics: RMSE, MAE, precision, and recall, obtained by testing on the validation set, are shown in Figure 5.2. Since the CBF performance is not close to the performance of CF or the hybrid recommender, only these two methods are compared to see where the hybrid outperforms CF. Note that the evaluation values of the CF stay fixed because the rule does not influence CF.



(a) RMSE (b) MAE

(c) Precision (d) Recall

Figure 5.2: Comparison of Evaluation Metrics for Collaborative Filtering and Hybrid Recommender

Based on the results of RMSE, MAE, precision, and recall plots, it is evident that both RMSE and MAE perform better and achieve lower errors for rule multipliers 3,4 and 5. Precision demonstrates a gradual increase with the rise in the rule multiplier, although the changes are relatively small. On the other hand, recall shows a decrease as the rule multiplier increases. Considering these observations, the most optimal rule multiplier for the data with the CF and CBF models in place appears to be 4. This results in the following rules:

Table 5.3: Best performing rules

Available Ratings	CBF Weight	CF Weight
< 4	0.9	0.1
4 - 8	0.8	0.2
8 - 12	0.7	0.3
12 - 16	0.6	0.4
16 - 20	0.5	0.5
20 - 24	0.4	0.6
24 - 28	0.3	0.7
28 - 32	0.2	0.8
32 - 36	0.1	0.9
36 $>$	0.0	1.0

The results of the best model with the rules above in place are presented in Table 5.4. The results are obtained from testing on the test dataset. The Hybrid model exhibits a marginal improvement in terms of RMSE and MAE compared to CF. Precision remains unchanged, and recall has a slight dip when rounded to two decimal places.

Table 5.4: Best dynamic Hybrid results

Recommender	RMSE	MAE	Precision	Recall
CF	1.16	0.88	0.87	0.93
CBF	1.64	1.35	0.87	0.45
Hybrid	1.14	0.87	0.87	0.92

5.2 Weighted Mean Squared Error & Weighted Accuracy

After identifying the most optimized model for the given data and model, it is now essential to explore the non-symmetric metrics: weighted Mean Squared Error (wMSE) and weighted accuracy (WA). These metrics are introduced in the methodology section and are based on the premise that users would prefer not to receive recommendations for movies they dislike (false positives or overestimation) rather than missing out on movies they would enjoy (false negatives or underestimation). To account for this preference, adjustable weights are assigned, with higher weightage given to false positives or overestimation compared to false negatives or underestimation.

Table 5.6 presents the results of wMSE concerning the variation in overestimation weights. As the weight assigned to overestimation increases while keeping all other factors constant, the error of the model consistently increases. This increase is attributed to the higher penalty imposed on overestimations. A higher weight indicates a worse performance; however, addressing the issue of overestimation with a higher penalty is necessary to reveal the true model performance. By doing so, the model can effectively account for overestimations and potentially improve its overall accuracy.

Table 5.5: Weighted Mean Squared Error

Weight Underestimation	**Weight Overestimation**	**Weighted MSE**
1	1.0	1.30
1	1.5	1.79
1	2.0	2.27
1	2.5	2.75
1	3.0	3.23
1	3.5	3.72
1	4.0	4.20

Additionally, a weighted measure for binary accuracy was introduced. Again, the performance diminishes as the penalty increases. This highlights the trade-off between accuracy and the strictness in handling overestimations.

Table 5.6: Weighted Accuracy

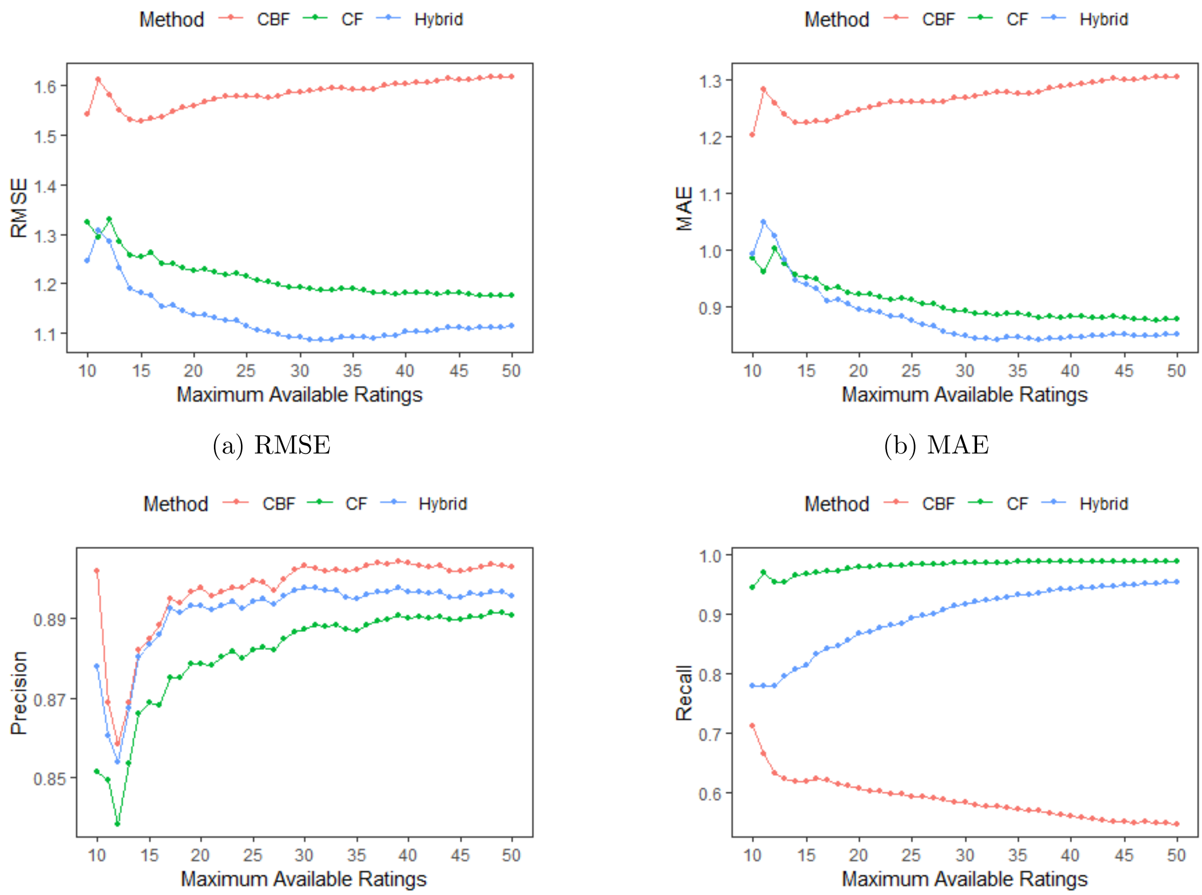
Weight False Negatives	**Weight False Positives**	**Weighted Accuracy**
1	1.0	0.88
1	1.5	0.83
1	2.0	0.78
1	2.5	0.75
1	3.0	0.71
1	3.5	0.68
1	4.0	0.65

5.3 Change of data availability

After identifying the most suitable model, the focus now shifts toward examining the impact of data availability on the model's performance. In Table 5.4, the overall evaluation metric scores were presented, encompassing all user types with varying amounts of available ratings. However, this analysis aims to investigate the results for users with relatively fewer available ratings.

By narrowing the attention to users with limited ratings, insights can be gained into how the model performs under data-scarce, cold start (CS) user scenarios. This examination provides valuable information about the model's robustness and effectiveness when dealing with sparse data.

For comparison reasons, the rule multiplier is maintained at the same value used in the final dynamic model, which is a multiplier of 4.



(a) RMSE (b) MAE

(c) Precision (d) Recall

Figure 5.3: Comparison of Evaluation Metrics for different data available

Figure 5.3 illustrates the variation in evaluation metrics concerning the change in the maximum available ratings within the training data. Across all metrics, the performance shows a decline for maximum ratings below 15. However, as the maximum ratings reach and exceed 15, the RMSE of the hybrid model begins to outperform CF. The MAE metric remains relatively similar for both CF and the Hybrid. In terms of precision, the hybrid demonstrates a slight advantage, while CF excels in the recall measure. These findings suggest that the hybrid approach becomes increasingly advantageous as the number of available ratings increases, particularly when the maximum ratings surpass the threshold of 15.

5.4 Different types of users

To segment the different types of users into groups K-mean clustering is used. The appropriate number of clusters is determined using the elbow rule, as depicted in Figure 5.4.

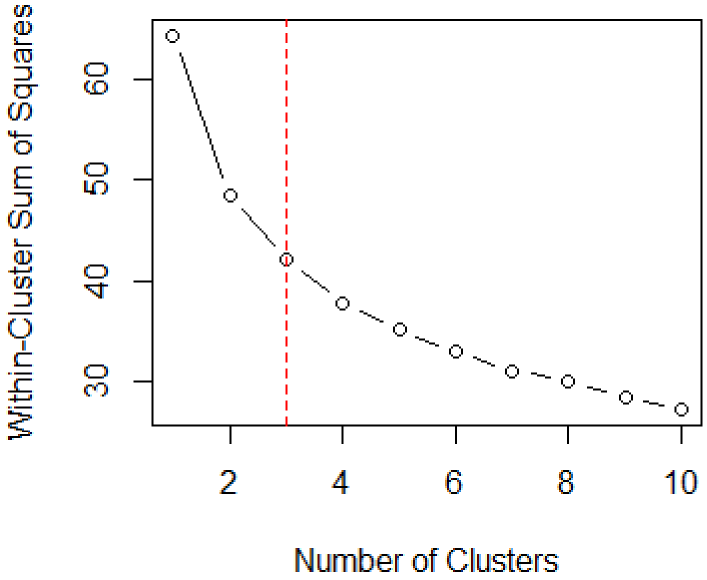


Figure 5.4: Elbow Rule

Three clusters were formed, resulting in the following groups along with their respective values based on the full 2% subset of data used in this research.

Table 5.7: K-Mean Clusters

			Proportion		
Cluster	**No. Users**	**No. Ratings**	**Action**	**Comedy**	**Drama**
1	1088	167604	0.05	0.06	0.78
2	936	135750	0.00	0.37	0.62
3	1227	176971	0.43	0.25	0.13
All users	3251	480325	0.18	0.22	0.49

Note: The proportions of the three most significant genres are shown, as they contribute the most to the formation of the clusters. 'No.' in columns 2 and 3 stands for 'number of'.

The table above provides valuable insights into the three clusters. It is intriguing to observe that each cluster exhibits a distinct preference for specific genres. Cluster 1 predominantly focuses on watching drama content. Cluster 2 displays a preference for drama, while also engaging significantly with drama content. Finally, Cluster 3 shows a preference for action, followed by comedy.

These findings shed light on the viewing behavior patterns within each cluster and highlight the varying genre preferences among different groups of users.

Table 5.8: Recommender Results Clusters

Cluster	**Recommender**	**RMSE**	**MAE**	**Precision**	**Recall**
1	CF	1.15	0.87	0.87	0.93
	CBF	1.61	1.33	0.88	0.44
	Hybrid	1.13	0.86	0.88	0.92
2	CF	1.18	0.90	0.87	0.93
	CBF	1.63	1.32	0.87	0.46
	Hybrid	1.16	0.89	0.87	0.93
3	CF	1.14	0.86	0.87	0.93
	CBF	1.69	1.38	0.86	0.45
	Hybrid	1.13	0.86	0.87	0.92

Utilizing the created clusters, it is possible to evaluate the performance of the recommender systems for each user cluster using the test dataset. The results of these evaluations are detailed in Table 5.8. When comparing the three clusters, there is a noticeable difference in performance, with the second cluster exhibiting slightly higher errors in the quantitative metrics RMSE and MAE compared to the first and third clusters.

These findings imply that CF, CBF, and the hybrid recommender system do not perform equally well in providing personalized recommendations to users across all clusters, considering their distinct genre preferences. This suggests a noteworthy disparity in the predictive performance of the recommendation system when addressing user groups with diverse genre preferences.

Chapter 6

Conclusion

This research proposed a dynamic hybrid recommendation system (RS) to adapt for cold start users. The system combines a content-based filtering (CBF) method that utilizes a user profile and a movie genre matrix and a collaborative filtering (CF) method that uses the Pearson correlation coefficient to calculate similarities between users and predicts movie ratings based on that. The combination of the predicted movie ratings is dynamically weighted based on the available input ratings for each user individually. The rules that adjust the weights are adapted until the recommendation system reaches the, for now, best possible performance. The hybrid RS did eventually outperform both the CF and CBF. However, this performance improvement is so little that the question is whether it is worth the extra computational time. Potential reasons and solutions for this result can be found in the next section.

To answer the sub-questions about how the performance of the hybrid RS varies with the amount of data available, tests of the hybrid with changing data availability were performed. The results showed that for really low numbers of data available the performance is increasingly worse than for higher numbers of available data. From approximately a maximum of 15 ratings available and above, the hybrid system starts to outperform CF based on the evaluation metrics RMSE, MAE and precision.

At last, this research investigated the performance of the hybrid in personalizing for different user types. The users were clustered into three clusters based on genre preferences. Between the clusters, there was a small difference in performance. This could conclude that the hybrid system is not yet optimal at personalizing recommendations based on genre preference differences and that there is room for improvement.

6.1 Limitations & Future Research

Due to the constrained time span for writing a master's thesis and other factors, such as the computational power of the computer and software used, this research does have some limitations.

For simplicity reasons, this research focuses on utilizing basic models, particularly CF and CBF. While more complex methods exist for both CF and CBF, they may offer improved accuracy. In future research, the exploration of dynamic hybrid recommender models incorporating sophisticated filtering techniques, such as deep neural networks in CF, holds promising potential.

The limitations in the computational power of both the program used for building and testing the models, as well as the computer itself, have imposed certain constraints. Consequently, a sample had to be taken from the complete rating data. Although the sample size was still quite substantial, having access to more data could potentially yield more reliable and robust results. In future research, utilizing more powerful research equipment could facilitate testing with larger datasets, further enhancing the credibility and scope of the findings.

Furthermore, in this research, the tuning and testing were conducted by dividing the data into a validation, test, and training set, enabling the tuning and eventually the comparison of predicted ratings with the existing testing data. However, for more precise and accurate results, future studies could consider experimental testing. Such an approach would allow for a more controlled evaluation, potentially yielding more accurate and reliable outcomes.

Moreover, the dynamic weighted hybrid recommender model is constructed based on a linearly increasing rule. To extend this research in the future, there is a potential for improvement by exploring alternative weighting rules, such as nonlinear approaches. By incorporating nonlinear weighting rules, the model could potentially capture more complex patterns and nuances in the data, leading to even more robust and accurate recommendations.

At last, the research on different user types could be expanded beyond genres. Exploring various factors, like demographics, watch history, engagement behavior, and more, can enhance personalization in recommendation systems, leading to improved user satisfaction and engagement.

Bibliography

Aggarwal, C. C. (2016). Content-based recommender systems. In *Recommender systems* (pp. 139–166). Springer International Publishing. https://doi.org/10.1007/978-3-319-29659-3_4

Almeida, F. A., Gomes, G. F., Paula, V. R., Correa, J. E., Paiva, A. P., Gomes, J. d. F., & Turrioni, J. B. (2018). A weighted mean square error approach to the robust optimization of the surface roughness in an AISI 12l14 free-machining steel-turning process. *Journal of Mechanical Engineering*, *64*(3), 147–156.

Amarikwa, M. (2023). Social media platforms' reckoning: The harmful impact of TikTok's algorithm on people of color. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.4349202

Ashabi, A., Sahibuddin, S. B., & Salkhordeh Haghighi, M. (2020). The systematic review of k-means clustering algorithm. *2020 The 9th International Conference on Networks, Communication and Computing*, 13–18. https://doi.org/10.1145/3447654.3447657

Belkin, N. J., & Croft, W. B. (1992). Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, *35*(12), 29–38. https://doi.org/10.1145/138859.138861

Benesty, J., Chen, J., Huang, Y., & Cohen, I. (2009). Pearson correlation coefficient [Series Title: Springer Topics in Signal Processing]. In *Noise reduction in speech processing* (pp. 1–4). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-00296-0_5

Bholowalia, P., & Kumar, A. (2014). EBK-means: A clustering technique based on elbow method and k-means in WSN [Publisher: Citeseer]. *International Journal of Computer Applications*, *105*(9).

B.Thorat, P., M. Goudar, R., & Barve, S. (2015). Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, *110*(4), 31–36. https://doi.org/10.5120/19308-0760

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, *12*(4), 331–370. https://doi.org/10.1023/A:1021240730564

Çano, E., & Morisio, M. (2017). Hybrid recommender systems: A systematic literature review. *Intelligent Data Analysis*, *21*(6), 1487–1524. https://doi.org/10.3233/IDA-163209

Chai, T., & Draxler, R. R. (2014, February 28). *Root mean square error (RMSE) or mean absolute error (MAE)?* (preprint). Numerical Methods. https://doi.org/10.5194/gmdd-7-1525-2014

Geetha, G., Safa, M., Fancy, C., & Saranya, D. (2018). A hybrid approach using collaborative filtering and content based filtering for recommender system. *Journal of Physics: Conference Series*, *1000*, 012101. https://doi.org/10.1088/1742-6596/1000/1/012101

Golbeck, J., & Norris, E. (2013). Personality, movie preferences, and recommendations. *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 1414–1415. https://doi.org/10.1145/2492517.2492572

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. https://doi.org/10.1145/138859.138867

Hahsler, M. (2022). Recommenderlab: An r framework for developing and testing recommendation algorithms [Publisher: arXiv Version Number: 1]. https://doi.org/10.48550/ARXIV.2205.12371

Harper, F. M., & Konstan, J. A. (2016). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems*, *5*(4), 1–19. https://doi.org/10.1145/2827872

Hu, Y., & Ogihara, M. (2011). Nextone player: A music recommendation system based on user behavior. *Poster Session.*

Kim, B. M., Li, Q., Park, C. S., Kim, S. G., & Kim, J. Y. (2006). A new approach for combining content-based and collaborative filters. *Journal of Intelligent Information Systems*, *27*(1), 79–91. https://doi.org/10.1007/s10844-006-8771-2

Knijnenburg, B. P., Reijmer, N. J., & Willemsen, M. C. (2011). Each to his own: How different users call for different interaction methods in recommender systems. *Proceedings of the fifth ACM conference on Recommender systems*, 141–148. https://doi.org/10.1145/2043932.2043960

Konstan, J. A. (1997). GroupLens: Applying collaborative filtering to usenet news.(special section: Recommender systems). *Communications of the ACM*, *40*(3), 77(11).

Lam, X. N., Vu, T., Le, T. D., & Duong, A. D. (2008). Addressing cold-start problem in recommendation systems. *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, 208–211. https://doi.org/10.1145/1352793.1352837

Li, B., & Han, L. (2013). Distance weighted cosine similarity measure for text classification [Series Title: Lecture Notes in Computer Science]. In H. Yin, K. Tang, Y. Gao, F. Klawonn, M. Lee, T. Weise, B. Li & X. Yao (Eds.). D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi & G. Weikum (**typeredactors**), *Intelligent data engineering and automated learning – IDEAL 2013* (pp. 611–618). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-41278-3_74

Lika, B., Kolomvatsos, K., & Hadjiefthymiades, S. (2014). Facing the cold start problem in recommender systems. *Expert Systems with Applications*, *41*(4), 2065–2073. https://doi.org/10.1016/j.eswa.2013.09.005

Likas, A., Vlassis, N., & J. Verbeek, J. (2003). The global k-means clustering algorithm. *Pattern Recognition*, *36*(2), 451–461. https://doi.org/10.1016/S0031-3203(02)00060-2

Liu, J. (2021). The influence of the body image presented through TikTok trend-videos and its possible reasons: https://doi.org/10.2991/assehr.k.210609.072

Lops, P., De Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In F. Ricci, L. Rokach, B. Shapira & P. B. Kantor (Eds.), *Recommender systems handbook* (pp. 73–105). Springer US. https://doi.org/10.1007/978-0-387-85820-3_3

Melville, P., & Sindhwani, V. (2010). Recommender systems. *Encyclopedia of machine learning*, *1*, 829–838.

Meteren, R. v. (2000). Using content-based filtering for recommendation. Retrieved July 10, 2023, from https://www.semanticscholar.org/paper/Using-Content-Based-Filtering-for-Recommendation-Meteren

Nam, L. N. H. (2022). Towards comprehensive approaches for the rating prediction phase in memory-based collaborative filtering recommender systems. *Information Sciences*, *589*, 878–910. https://doi.org/10.1016/j.ins.2021.12.123

Papadakis, H., Papagrigoriou, A., Panagiotakis, C., Kosmas, E., & Fragopoulou, P. (2022). Collaborative filtering recommender systems taxonomy. *Knowledge and Information Systems*, *64*(1), 35–74. https://doi.org/10.1007/s10115-021-01628-7

Patel, K. M. A., & Thakral, P. (2016). The best clustering algorithms in data mining. *2016 International Conference on Communication and Signal Processing (ICCSP)*, 2042–2046. https://doi.org/10.1109/ICCSP.2016.7754534

Patro, S. G. K., & Sahu, K. K. (2015). Normalization: A preprocessing stage [Publisher: arXiv Version Number: 1]. https://doi.org/10.48550/ARXIV.1503.06462

Petrescu, M., & Krishen, A. S. (2020). The dilemma of social media algorithms and analytics. *Journal of Marketing Analytics*, *8*(4), 187–188. https://doi.org/10.1057/s41270-020-00094-4

Resnick, P., & Varian, H. R. (1997). Recommender systems [Publisher: ACM New York, NY, USA]. *Communications of the ACM*, *40*(3), 56–58.

Schein, A. I., Popescul, A., Ungar, L. H., & Pennock, D. M. (2002). Methods and metrics for cold-start recommendations. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 253–260. `https://doi.org/10.1145/564376.564421`

Yuan, C., & Yang, H. (2019). Research on k-value selection method of k-means clustering algorithm. *J*, *2*(2), 226–235. `https://doi.org/10.3390/j2020016`