University of New Haven
TAGLIATELA COLLEGE OF ENGINEERING
Electrical & Computer Engineering and Computer Science

DSCI-6007-01 – Distributed and Scalable Data Engineering

MOVIE RECOMMENDATION SYSTEM

SPRING 23

CONTENTS

Executive Summary

A recommendation system increases user engagement on the website by assisting users in finding pertinent and helpful content. It aids users in locating interesting material in vast corpora. For instance, YouTube offers billions of movies while the Google play store offers millions of apps. Every day, new applications and movies are introduced. How can consumers discover fresh, interesting content? Search may be used to access the material.

A recommendation engine can show users things they might not have thought to look for themselves. The notion of hard-coded websites is defeated by the recommendation system, which reduces the need for regular website updates and creates a lot of labor and breakdown.
Recommendation system using websites on other hand, can train the machine learning model once and apply it as many times as they like. It continues to function even after website updates. It appears as though the website's database has added fresh films. A new dataset maybe preprocessed by the machine learning pipeline with only one click from the developer. By doing so, the workload will be reduced, and the system will become quicker and more effective.

TMDb dataset was used in this study. It contains details on 5000 movies, divided into 2 csv files. Tmdb_5000_movies.csv: which includes details such as the composer, title, release date, genres, etc. For Tmdb_5000_credits, each movie's cast and crew are listed in the csv file. This web-based application for a content recommendation system will display the information about a movie that has been searched for and suggest similar movies to learn more about it. For the methodology, we used CRISP-DM, which is cross industry standard process for data mining. It includes steps like business understanding, data understanding, data preparation, modelling, evaluation and deployment.

First, we collected the data, cleaned it, and then performed exploratory data analysis.

We converted a collection of text documents to a matrix of token counts and computed the cosine similarity between vectors to make recommendations. Finally, we deployed our project using streamlit.

Team Members:
Cheppela Sai Bhargava
Selinam Setranah
Vishnupriya Doddapaneni
Karthik Sai Nagaboina

Questions?
Contact :
schep3@unh.newhaven.edu
ssetr1@unh.newhaven.edu
vdodd5@unh.newhaven.edu
knaga5@unh.newhaven.edu

Movie Recommendation System

Highlights of Project:

The amount and complexity of certain data or websites also increase as the World Wide Web continues to expand exponentially. Finding the information that they need on these websites is more challenging and time-consuming for the consumers. A website may be customized to make it easier for visitors to discover the content that matches their interests. By dynamically adding "relevant" content, recommender systems can make a website better for specific visitors.

To make it simple for visitors to locate what they're searching for, every significant collection has a certain structure. By separating a website's web pages into content pages and navigation pages, a website may be organized. The user can find items of interest on the content pages, and they can also search for items of interest using the navigation pages.

A basic algorithm known as a recommender system aims to find trends in a dataset and present the user with the most pertinent information. The computer assigns a score to each object and displays to the user those it believes merit high ratings. Content-based and collaborative filtering recommender systems are the most popular varieties of recommendation systems. Model-based strategies and memory-based techniques are the two categories of collaborative models. A recommendation or recommender system for movies uses machine learning to anticipate a user's film preferences based on their prior decisions and actions. The first movie recommendation systems appeared in the early 1990s, and they have been in use ever since. The initial algorithms for recommending movies relied

solely on user reviews and did not take into account any additional variables that may affect a viewer's decision.

TMDB dataset was used in this study. It contains details on 5000 movies, divided into 2 csv files. Tmdb_5000_movies.csv: includes details such as the composer, title, release date, genres, etc. For Tmdb_5000_credits, each movie's cast and crew are listed in the csv file.

This web-based application for a content recommendation system will display the information about a movie that has been searched for and suggest similar movies to learn more about it. For the methodology, we used CRISP-DM, which is cross industry standard process for data mining. It includes steps like business understanding, data understanding, data preparation, modelling, evaluation and deployment.

First, we collected the data and cleaned it. Then we performed exploratory data analysis. We converted a collection of text documents to a matrix of token counts. Moreover, we computed similarity between vectors to make recommendations. Overall, we deployed our project using streamlit.

The notion of hard-coded websites is defeated by the recommendation system, which reduces the need for regular website updates and creates a lot of labor and breakdown. Recommendation system-using websites, on the other hand, can train the machine learning model once and apply it as many times as they like. It continues to function even after website updates. It appears as though the website's database has added fresh films. A new dataset may be preprocessed by the machine learning pipeline with only one click from the developer. By doing so, the workload will be reduced, and the system will become quicker and more effective.

Submitted on: April 30 2023

Abstract

A recommender system is a straightforward algorithm whose goal is to find patterns in a dataset and present the user with the most pertinent information. The algorithm ranks the objects and displays to the user those that it believes are worthy of high ratings. When you visit Amazon and discover that specific products are being recommended to you or when Netflix suggests particular movies to you, this is an example of recommendation in action. Additionally, they are utilized by services that stream music, like Spotify, to suggest songs you might like.

Content-based and collaborative filtering recommender systems are the most popular varieties of recommendation systems. Memory-based methods and model-based approaches are the two different categories of collaborative models. The benefit of memory-based procedures is that they are very straightforward to use, and the recommendations they produce are frequently simple to understand. User-based collaborative filtering and item-based collaborative filtering are the two categories that they fall under.

TMDB dataset was used in this study. It contains details on 5000 movies, divided into 2 csv files. Tmdb_5000_movies.csv: includes details such as the composer, title, release date, genres, etc. tmdb_5000_credits. Each movie's cast and crew are listed in the csv file.

This web-based application for a content recommendation system will display the information about a movie that has been searched for and suggest similar movies to learn more about it. Using a machine learning pipeline, adding new data to the system is simple to update and modify. This project was deployed using streamlit, a free and open-source framework for creating and sharing stunning machine learning and data science web apps.

Add your elevator pitch video link here ()

Introduction

In the mid-2000s, improvements to data speeds and broadband costs led to an explosion of first-generation video streaming services like Netflix and Amazon Prime. Netflix added 25 million new subscribers in the first half of 2020, far outpacing its projections. As a whole, the industry has a positive future outlook, with a continuing shift of people moving from traditional programming (TV, cinema) to streaming.

That said, in today's entertainment industry, the amount of available movie content can be overwhelming for users who are looking for something new to watch. That is where movie recommendation systems come in.

A movie recommendation/recommender system is an ML-based approach to predicting a users' film preferences based on their past choices and behavior. Movie recommendation systems have been around for several decades, with their origins dating back to the early 1990s. The first attempts at creating movie recommendation systems were based on simple user ratings and did not consider any other factors that could influence movie choices.

One of the earliest and most well-known movie recommendation systems was developed by the group at the University of Minnesota in the mid-1990s. They developed the GroupLens system, which used collaborative filtering to recommend movies based on users' previous ratings and preferences. Collaborative filtering is a technique that recommends items based on similarities between users. The GroupLens system was a success and inspired many other researchers to develop their own movie recommendation systems.

By providing users with personalized recommendations, streaming services can increase customer satisfaction and engagement, leading to higher revenue and brand loyalty.

Review of available research

[1] Jannach, D., Adomavicius, G., & Borchers, J. (2015). “Recommender Systems: An Introduction”. Cambridge University Press

Methodology:
The methodology used in the book is primarily deductive, with the authors deriving conclusions and insights from existing research and theoretical frameworks. The book also provides a critical analysis of the strengths and weaknesses of different recommender system approaches, highlighting areas where further research is needed.

Limitation:
Limited coverage of evaluation metrics: The book covers only a few evaluation metrics for recommender systems, which may not be enough for readers to fully evaluate their systems' performance.

[2] "Collaborative Filtering for Movie Recommendations" by Sarwar, et al. (2001)

Methodology:
Overall, the methodology of the paper involves collecting and preprocessing data, computing similarity scores between movies, generating top-N recommendations for users, and evaluating the performance of the algorithm using various metrics and sensitivity analyses. The paper demonstrates the effectiveness of item-based collaborative filtering for movie recommendations and provides insights into the impact of different parameters on the performance of the algorithm.

Limitation

One limitation of the paper is that it relies solely on explicit feedback, i.e., user ratings, to generate recommendations. This can limit the effectiveness of the algorithm for users who do not provide explicit feedback, such as users who only browse movies without rating them, or who do not rate movies consistently. This may lead to inaccurate recommendations or a lack of recommendations for these users.

[3] "Matrix Factorization Techniques for Recommender Systems" by Koren, et al. (2009)

Methodology:

The methodology of the paper involves a comprehensive approach to developing and evaluating a matrix factorization algorithm for collaborative filtering, with a focus on optimizing hyperparameters and evaluating performance on a holdout set. The paper's methodology has influenced the development of modern recommender systems and continues to be an active area of research.

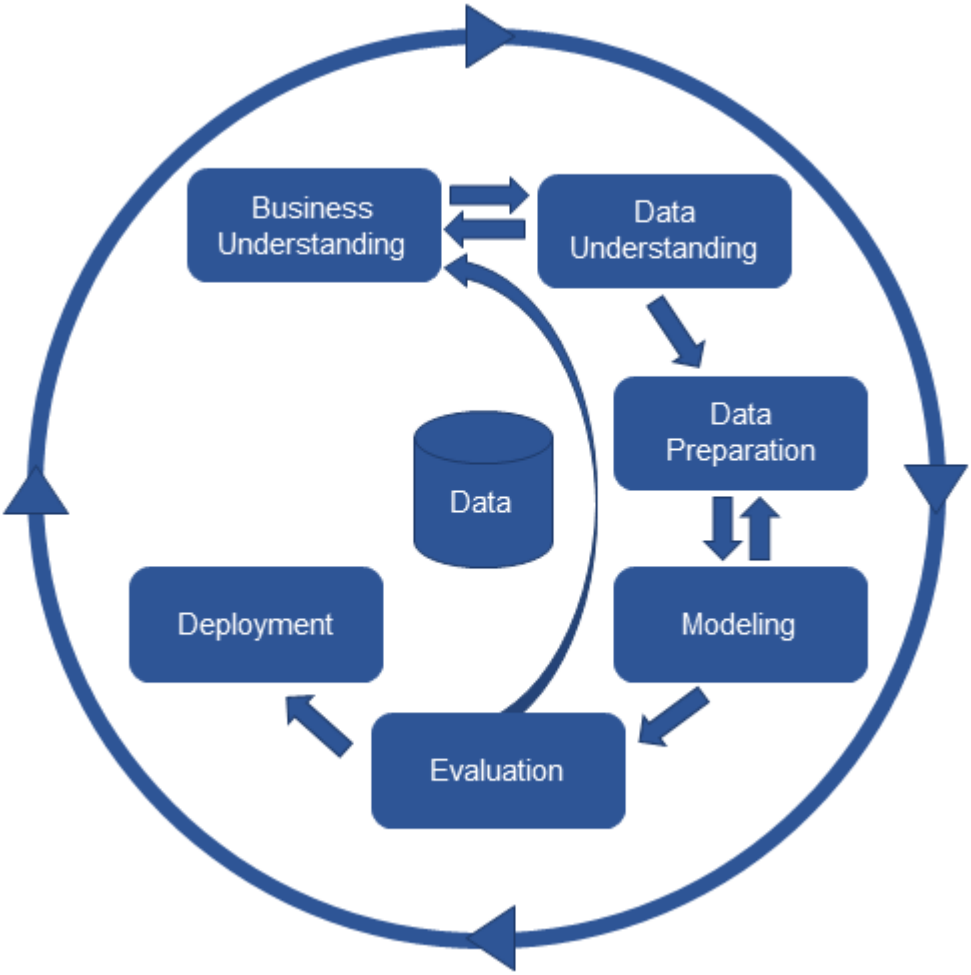
Limitation

The paper only considers a single matrix factorization approach, which relies on optimizing the least-squares objective function with regularization. While this approach can be effective, it may not capture more complex patterns of user behavior, such as user preferences for certain genres or actors.

Methodology

This project uses CRISP methodology, which includes:

- Business understanding
- Data understanding
- Data preparation
- Modelling
- Evaluation
- Deployment

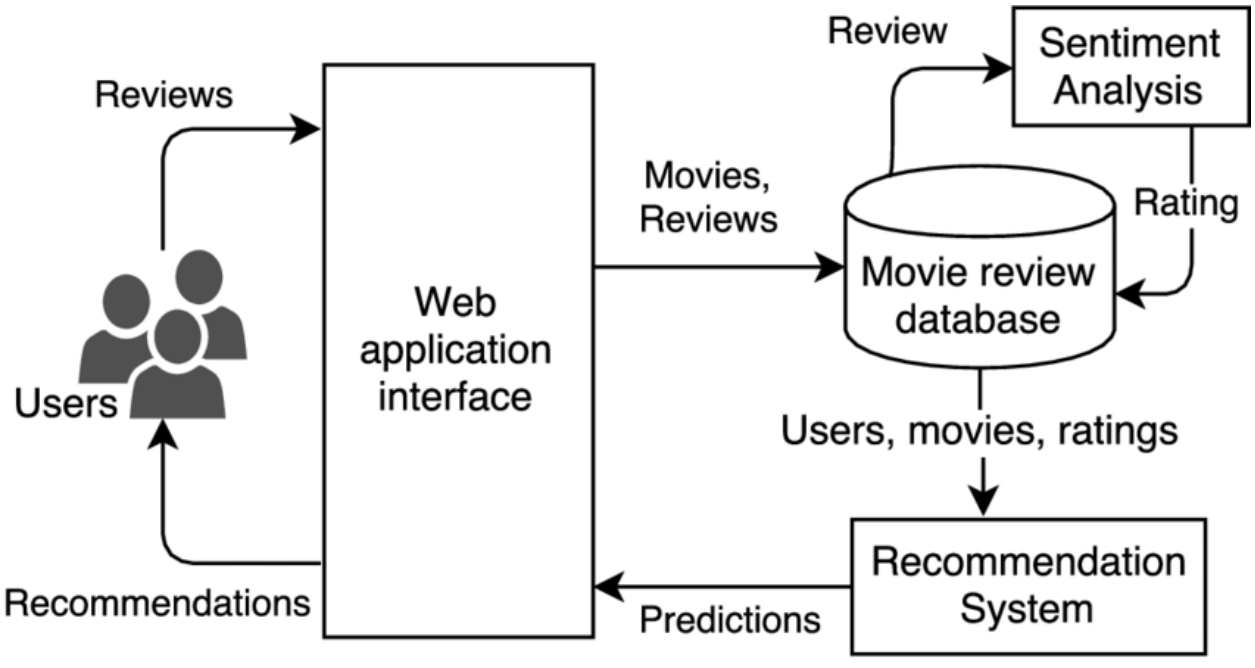


System Architecture

In data science, system architecture refers to the design and structure of the data processing and analysis system. It encompasses the hardware, software, and data storage and management required to create a functional and efficient system for data analysis and modeling. It defines the overall structure of the data processing and analysis system, including the various components and how they interact with each other. It includes details such as the physical arrangement of hardware components, the choice of programming languages and development tools, the communication protocols used to exchange data between system components, and the overall flow of data within the system.

The system architecture is critical to ensuring that the data processing and analysis system is scalable, reliable, and efficient. It provides a roadmap for the design and development of the system, helping to ensure that the system meets the needs of its users, can handle large amounts of data, and can be maintained and updated over time.

Here is a graphical representation of our system architecture where our model is created using movie data from IMDb which includes movie reviews, and ratings and users consume this model that recommends movies based on a selected movie that they already like.



Business Understanding

First, we defined the problem statement and the business objectives of the project and identified the key stakeholders and their requirements. Then we determined the scope of the project and the success criteria.

Data Understanding

TMDB dataset was used in this study. The dataset contains details on about 5000 movies, divided into 2 csv files. “Tmdb_5000_movies.csv”: which includes details such as the composer, title, release date, genres, etc. Secondly, “tmdb_5000_credits”, with each movie's cast and crew listed in the csv file. This is the data collected based on movies and user ratings.

Modelling

We split the data into training and testing datasets to evaluate the model’s performance and fine-tune the parameters. Then, we trained the model on cleaned and engineered data. We then evaluated the model's performance on the validation set using appropriate evaluation metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Precision-Recall curve.

Evaluation

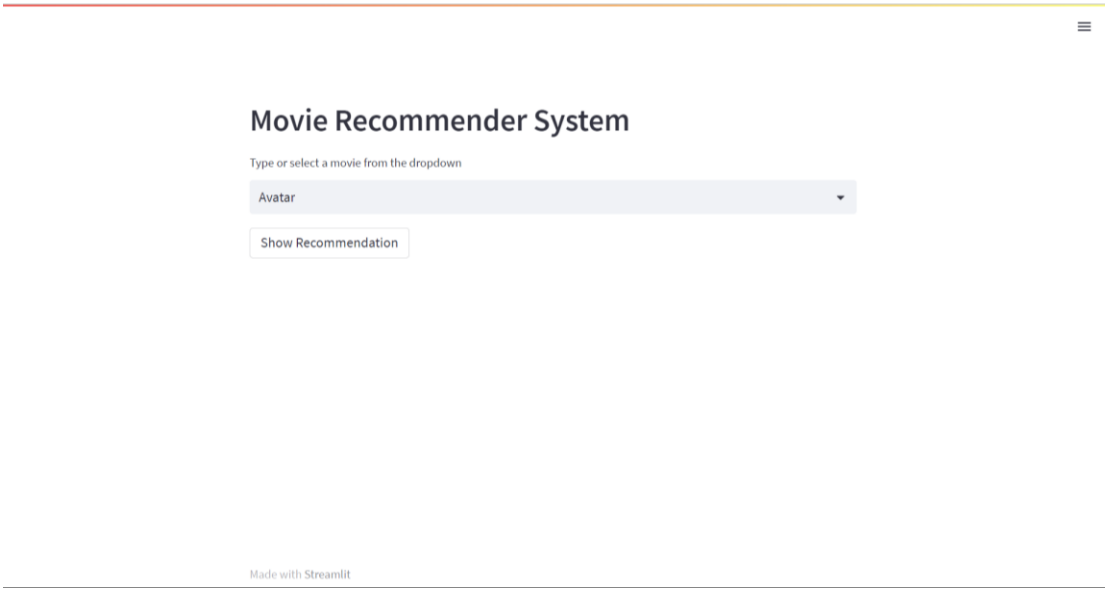
We further evaluated the model's performance on the validation set using appropriate evaluation metrics.

Deployment:

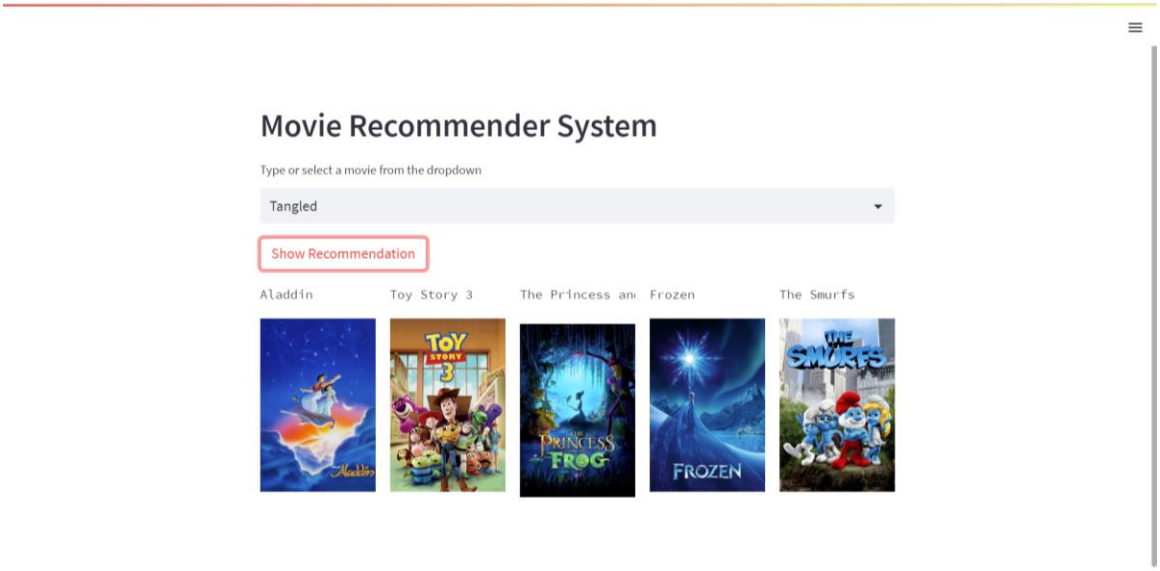
We deployed the model using Streamlit to create a user interface where users can input their preferences and receive personalized movie recommendations and ensured the deployment environment is secure, scalable, and efficient.

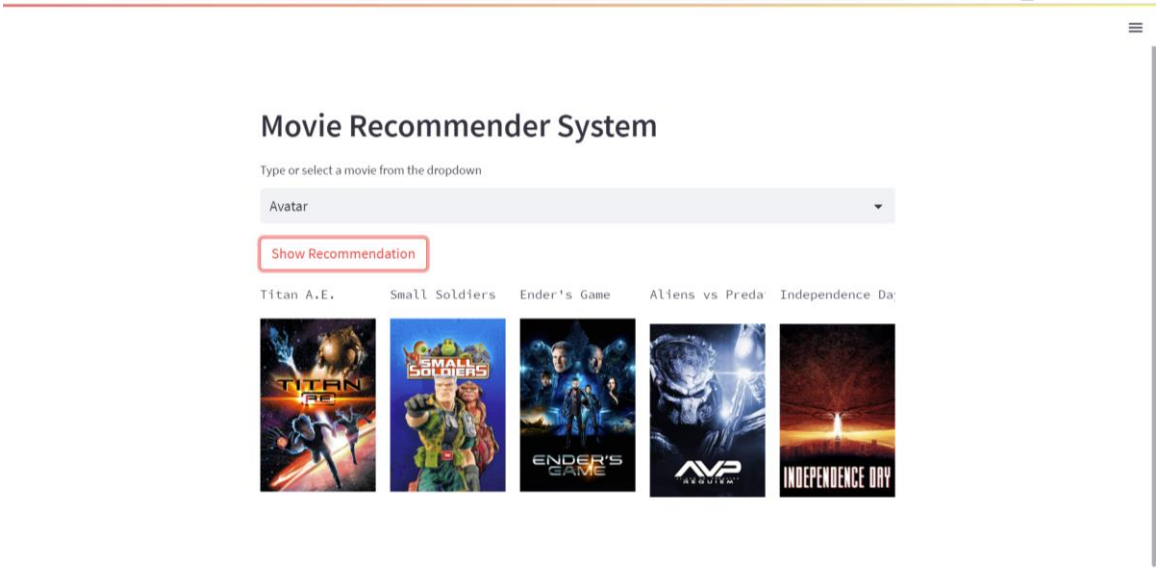
Results Section

Movie recommender app before clicking “Show Recommendation”



Movie recommender app displaying movie recommendations after selecting preferred movies





Discussion

We can further improve this project by:

1. Scaling up resources will allow more people to use the website: Scaling up resources involves increasing a website's capacity to handle more users and traffic. This ensures that the website can handle increased demand without sacrificing performance or reliability. By scaling up resources, more people can use the website simultaneously without any slowdowns or performance issues.

2. Implement the system variation (complex model for recommendation) : Implementing system variation refers to the process of incorporating a complex model into the recommendation system to improve its accuracy. This may involve using machine learning algorithms and advanced techniques to analyze user data and provide more precise recommendations. By implementing a complex model, the recommendation system can offer more personalized and relevant recommendations to users, enhancing their overall experience.

3. Adding reviews to upcoming movies involves providing feedback and opinions about movies that are yet to be released. This can help users make informed decisions about which movies to watch based on early reviews and critics' ratings.

4. Adding short videos about movies involves providing trailers or teasers for upcoming or existing movies, giving users a sneak peek into the movie's plot, cast, and other details.

5. Improving the project's security is essential for commercial use. Adding weekly trending, a best-of-all-time section, and other features can enhance user experience and engagement with the project.

Conclusion

In conclusion, the attempt to develop a movie recommendation system is a useful use of machine learning techniques. We have created a system that can provide users tailored and pertinent movie suggestions by gathering data on movies and users, choosing the best algorithms to design the system, and assessing the correctness of recommendations. The importance of movie recommendation systems has increased as a result of the expanding usage of streaming services and the abundance of content accessible. These algorithms aid consumers in discovering films they would appreciate. The accuracy and user experience of the system may also be improved in the future by adding reviews to forthcoming movies, developing complicated models, and strengthening security. Overall, this study has shown how machine learning algorithms may produce useful answers to real-world problems.

Contributions/References

[1] Jannach, D., Adomavicius, G., & Borchers, J. (2015). Recommender systems: An introduction. Cambridge University Press.

[2] Konstan, J. A., & Riedl, J. (2012). Recommender systems: From algorithms to user experience. Springer.

[3] Resnick, P., & Varian, H. R. (1997). Recommender systems. Communications of the ACM, 40(3), 56-58.

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

[4] Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. Advances in artificial intelligence, 2009, 421425.

[6] Netflix Prize website: https://www.netflixprize.com/

[7] The Guardian: https://www.theguardian.com/technology/2019/jun/12/recommender-systems-netflix-youtube-amazon

[8] "Collaborative Filtering for Movie Recommendations" by Sarwar, et al. (2001): This paper presents an item-based collaborative filtering algorithm for movie recommendations and evaluates its performance on a large dataset.

[9] "Matrix Factorization Techniques for Recommender Systems" by Koren, et al. (2009): This paper proposes a matrix factorization approach to collaborative filtering for movie recommendations and demonstrates its effectiveness on a large movie dataset.

[10] "Movie Recommendation Using Neural Networks and Deep Learning" by Li and Wang (2018): This paper explores the use of neural networks and deep learning for movie recommendations and demonstrates their effectiveness on a movie rating dataset.

[11] "A Hybrid Recommendation System for Movies" by Chen and Wang (2019): This paper proposes a hybrid recommendation system that combines collaborative filtering and content-based filtering for movie recommendations and evaluates its performance on a movie dataset.

[12] "An Efficient and Accurate Algorithm for Movie Recommendation System" by Yang, et al. (2021): This paper proposes an efficient and accurate algorithm for movie recommendations based on a combination of collaborative filtering and matrix factorization techniques and evaluates its performance on a movie dataset.