Movie Recommendation Using Machine Learning Techniques: Comparing K-Nearest Neighbors and Singular Value Decomposition

Yoon-Teck Bau1, a), Nur Irdina Sofea binti Mohd Mahadir1, b) and Kian-Chin Lee1, c)

1Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia.

b) Corresponding author: irdnasfea0408@gmail.com  
a)ytbau@mmu.edu.my   
c)kclee@mmu.edu.my

**Abstract.** With the explosion of movie releases and the broad variety videos by online streaming platforms have resulted in an age of unparalleled choice for consumers in the quickly expanding entertainment environment. The purpose of this research is to implement and comparing two machine learning techniques for movies recommendation in data science domain, with the aim of improving user satisfaction by providing personalized choices. The movies dataset from Data.world is utilized, containing approximately 45,000 movies. Next, six data preprocessing methods are applied to transform the raw dataset into a suitable format for analysis. These methods involve handling null values, checking uniqueness of values in the dataset, removing irrelevant features, replacing zero with not a number to indicate missing values, converting features to numeric data type and extracting the years. Subsequently, exploratory data analysis methods are utilized to visually enhance our comprehension of the dataset by generating word clouds for movie titles and movie overviews to discern common themes, creating world map for movies production countries, and plotting charts for number of movies by its original languages, number of released movies in a particular month, number of released movies on a particular day, number of released movies by year, and number of movies across different genres. To fulfil research gaps, two comparable machine learning techniques, which are K-Nearest Neighbors (KNN) and Singular Value Decomposition (SVD), were chosen for recommendation towards personalized movie selection. SVD outperforms KNN based on a comparative analysis of three performance metrics, which are Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE), showing the lowest values in all metrics.

# INTRODUCTION

The explosion of movie production and the broad variety of streaming platforms have resulted in an age of unparalleled choice for consumers in the quickly expanding entertainment environment. The sheer volume of contents available, cut across different genres and platforms, has made it challenging for viewers who desire specialized movie suggestions to suit their specific needs. This sheer abundance tends to make consumers miss out of their interested movies, resulting in frustration and information overload.

To tackle this severe problem, data science techniques are to be utilized to create a movies recommendation. By analyzing user preferences, watch history, and learning from comparable user profiles, this recommendation techniques endeavors to present consumers with precise and personalized movie recommendations. This recommendation technology hopes to speed up the movie discovery process by harnessing the power of data to give a curated list of movies that resonate with individual interests.

# LITERATURE REVIEW

A number of machine learning approaches to solving movies recommendation issues have been proffered in the past. [1] researched the creation of a personalized movie recommendation system by way of the integration of user preferences and movie contents. Their approach uses the MovieLens dataset to build user and movie profiles, as well as weekly movie ranking data. Issues they encounter in their research are with scalability and incorporating real-time input data, reflecting scope for future improvement to deal with large-scale real-world applications.

[2] did a thorough evaluation of movie recommender systems, concentrating on different recommendation strategies. The authors cited [3] worked on improving movies recommendation using a combination of KNN and collaborative filtering, highlighting the relevance of movie characteristics and attributes.

[4] investigated a hybrid technique using collaborative filtering and content-based filtering for recommendation system. The emphasis is on tailored movies recommendation to address the information overload that consumers are experiencing. The researchers combine content-based and collaborative filtering techniques and provide a hybrid strategy to improve the precision of movies recommendation. The MovieLens dataset is used to collect important data such as popularity and attractiveness. Collaborative filtering examines user behavior, whereas content-based filtering considers qualities to provide more specific recommendations. The limitation of the study is the subjective nature of movie choices, which makes objectively evaluating system performance difficult.

[5] investigated movies recommendation systems in comparative by incorporating SVD in user-based collaborative filtering technique. The results demonstrate the persuasive effectiveness of SVD in improving the user-based collaborative filtering method, using a dataset obtained from undergraduate students’ movie evaluations to solve issue associated with small amounts of information.

[6] investigated movie genre classification by employing collaborative filtering and KNN on the MovieLens dataset. They utilize Pearson correlation and Jaccard index as similarity measures to create a movie-similarity network, aiming to improve classification performance. Their approach, which relies heavily on user ratings, achieves an F1-score of 0.70. However, it may not fully consider the content’s depth and encounters scalability issues.

[7] addressed the topic of movie recommendation using a machine learning-based approach. Their hybrid method, which combines collaborative and content-based approaches, appears to be a viable direction towards solving the problem of recommendation accuracy improvement. The prevalence of the data sparsity problem, however, indicates a challenge in making effective recommendations for new or unrated items, demonstrating an intrinsic limitation of their methodology.

In [8] they investigated the possibility of improving movies recommendation systems through hybridizing content-based, collaborative filtering and neural network methods. It can be seen from their research that, by using user and item collaborative filtering with content-based based on cosine similarity, they were capable of improving the diversity and accuracy of recommendations. Additionally, to address the issue of data in user-item interactions, they incorporated network embeddings. This approach sheds light on the challenges faced by recommendation systems when striving for personalized recommendation. It highlights the need for technique capable of effectively handling sparse data while improving personalization in movies recommendation.

[9] applied collaborative filtering technique for the movies recommendation. For technique evaluation, they use two performance metrics which are RMSE and Mean Absolute Percentage Error (MAPE). Their technique did not fully explore how to optimize for scalability or how to deal with the data sparsity issue, which is crucial for new users using recommendation systems.

[10] investigated expert system for movies recommendation. Their technique employs fuzzy model to provide a final list of selected movies considering the user's movie favorites and least preferred movie genres. While their proposed technique succeeds at recommending relevant movies, it has limits when dealing with user' favorite movies and least favorite movie genres, particularly, if a user defines a genre as unpopular, then their proposed technique may provide movies recommendation that are not aligned to the user's preferences.

[11] investigated hybrid movies recommendation models that utilize machine learning to improve the customization of movie characteristics and customer preferences. It utilizes the MovieLens dataset to develop and evaluate techniques such as content-based, collaborative filtering, and latent factor-based models, with a focus on SVD. The study presents a weighted linear combination model that outperforms individual models in predicted accuracy, overcoming the constraints of single approach systems. The work acknowledges the computing requirements, particularly for complex models such as SVD, indicating a possible need for research to enhance efficiency.

[12] investigated the creation of a movies recommendation system using deep learning technique. The study used collaborative filtering technique using user attributes to supply users with the right information and speed up decision making. The study also recorded an accuracy rate of 65.8%, indicating promising results. Their research processes included data preprocessing through feature engineering, movie recommendation through KNN, and validation through a deep learning-based matrix factorization model. Their validation faces challenge which was associated with untrusted internet data.

[13] conducted a survey of movie recommendation systems based on machine learning approaches. The paper plunges into datasets such as MovieLens and IMDb covering techniques such as content-based, collaborative filtering as well as hybrid approaches that combine the two for better accuracy. The paper also mentions clustering algorithms such as k-means for partitioning items and predicting user ratings. The paper also points out challenges such as scalability issues, the start problem for new users with no past history, as well as overfitting in content-based approaches. These problems lead to constraints on predictability and system performance.

[14] experimented with machine learning based recommendation system on movie reviews using the KNN algorithm. They used term frequency with inverse document frequency and doc2vec for content-based technique to handle the scarcity in user interactions and enhance recommendation accuracy. Their methodology involved preprocessing the data and enhancing content-based technique using text processing methods. They also point out challenges such as text processing methods and content-based technique limits, in properly capturing various user desires.

[15] investigated the development of a movies recommendation that combined self-organizing map based neural network, collaborative filtering, and content-based techniques. By using complex techniques integration, their research employed a hybrid technique to reduce the drawbacks of separate stand-alone techniques. Their proposed technique performance is evaluated using accuracy and precision.

[16] investigated movies recommendation with content-based technique incorporating user demographics. Two datasets from Kaggle are used. The two datasets contained movie information such as cast, crew, and popularity. Based on content analysis, movies are recommended using term frequency with inverse document frequency and cosine similarity. However, their research noted that catering to new users was difficult [17].

# IMPLEMENTATIONS

The dataset is obtained from the Data.world dataset which contains approximately 45,000 movies. The dataset has 24 columns, each containing essential information for our study. Most of the columns in the dataset are attributes generated from the movie’s metadata. Six data preprocessing methods are applied to transform the raw dataset into a suitable format for analysis. These methods involve handling null values, checking uniqueness of values in the dataset, removing irrelevant features, replacing zero with not a number to indicate missing values, converting features to numeric data type and extracting the years. Possible exploratory data analysis methods are utilized to visually enhance our comprehension of the dataset by generating word clouds for movie titles and movie overviews to discern common themes, creating world map for movies production countries, and plotting charts for number of movies by its original languages, number of released movies in a particular month, number of released movies on a particular day, number of released movies by year, and number of movies across different genres.

Many studies have explored content-based and collaborative techniques in movie recommendations, but there is still a need to address the contradiction of choice, which occurs when users feel overwhelmed by the wide selection of movies in the movie industry. This is demonstrated by the ongoing issue of user dissatisfaction and decision fatigue, even though there are numerous suggestions available. Although current techniques have their advantages, concerns such as inadequate personalization and managing contents complexity remain key issues. This research aims to reduce the gap by utilizing two comparable machine learning techniques to personalize recommendation of movies.

## K-Nearest Neighbors (KNN)

Machine learning is part of data science techniques. An important machine learning technique known for its ease of implementation and effectiveness in recommendation applications is KNN. KNN stands out as an instance-based, non-parametric learning technique. This indicates that KNN do not create a generalizable model from the training data, compared to many other machine learning methods. Rather, it saves the whole data and uses it for calculations to generate predictions as needed. The technique works based on feature similarity, averaging the scores of the ‘k’ closest neighbors to a query point or using the majority vote to determine the output. Recommendation technique using user-based collaborative filtering and KNN Equation (1) as follows.

|  |  |
| --- | --- |
|  | (1) |

The predicted rating, *r̂ui*, that a user *u* would give an item *i* is calculated using user-based method in the KNN technique. In user-based collaborating filtering, the recommendation is calculated by total of the multiplying user *u* by other users *v* in terms of similarity *sim*(*u*, *v*) and the rating given by other user to item *rvi*. To normalize this total, it is then divided by the total of the similarities.

Pseudocode of recommendation technique using KNN as follows.

-----------------------------------------------------------------------------------------------------------------------------

Algorithm: KNN

-----------------------------------------------------------------------------------------------------------------------------

1. sum1 = 0

2. sum2 = 0

3. for each user *u* and user *v* that has rated item *i*:

4. sum1 += similarity(*u*, *v*) \* rating(*v*, *i*)

5. sum2 += similarity(*u*, *v*)

6. prediction = sum1 / sum2

7. return prediction

-----------------------------------------------------------------------------------------------------------------------------

## Singular Value Decomposition (SVD)

In data science, SVD is a widely used machine learning technique. In matrix factorization, the Equation (2) represents the user's rating prediction for an item in collaborative filtering using SVD.

|  |  |
| --- | --- |
|  | (2) |

The rating that user *u* is expected to assign to item *i* is represented by the symbol *r̂ui*. The average rating for every user and item in the collection is represented by the variable *μ*. Bias factors for user *u* and item *i* are denoted as *bu* and *bi*, respectively. Furthermore, the product of the transposed latent factor vector for item *i* and the latent factor vector for user *u*, represented by and *pu*, respectively. The predicted rating by summing all four components indicates the likelihood that the item will satisfy the user which represents the relation between the user’s preferences and the item’s attributes.

Given a matrix *A*, which is a *m*×*n* matrix, SVD decomposes *A* into three distinct matrices: *U*, Σ, and *V*\* where the matrix *U* is an *m*×*m* orthogonal matrix, the matrix Σ is an *m*×*n* diagonal matrix, and the matrix *V*\* is the conjugate transpose of an *n*×*n* orthogonal matrix. The columns of *U* and *V* are called the left-singular vectors and right-singular vectors of *A*, respectively. Mathematically, the decomposition is written as: *A* = *U* Σ *V*\*. Both equations in recommendation can be conceptually related to the SVD that involves decomposing a matrix into factors that capture underlying patterns in the data. They play a similar role to the singular vectors in SVD, reconstructing user-item interactions to predict unknown ratings.

Pseudocode of recommendation technique using SVD as follows.

-----------------------------------------------------------------------------------------------------------------------------

Algorithm: SVD

-----------------------------------------------------------------------------------------------------------------------------

1. eigenvalues\_U, eigenvectors\_U = eigen(A \* transpose(A))

2. eigenvalues\_V, eigenvectors\_V = eigen(transpose(A) \* A)

3. sorted\_indices\_U = argsort(eigenvalues\_U, descending=True)

4. sorted\_indices\_V = argsort(eigenvalues\_V, descending=True)

5. U = eigenvectors\_U[ : , sorted\_indices\_U]

6. singular\_values = sqrt(eigenvalues[sorted\_indices\_U])

7. Ʃ = diagonal(singular\_values)

8. V = eigenvectors\_V[ : , sorted\_indices\_V]

9. return U, Ʃ, transpose(V)

-----------------------------------------------------------------------------------------------------------------------------

# RESULTS AND DISCUSSION

A comparative evaluation of two machine learning techniques which are KNN and SVD using three performance metrics were analyzed. By analyzing MAE, MSE and RMSE the goal is to offer an evaluation of each technique performance and dependability. The results indicate that the SVD technique outperforms KNN technique across three performance metrics.

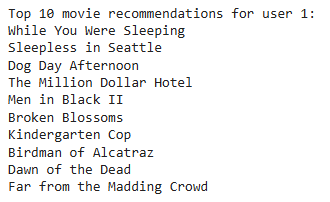
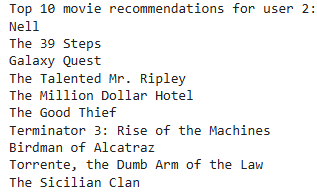
Table 1 shows the comparison of three performance metrics across KNN and SVD. The MAE for KNN technique is 0.7499 while SVD is 0.6968. The SVD technique shows a lower MAE value which shows a better result compared to the KNN. Next, for the MSE values attained for KNN and SVD are 0.9504 and 0.8146, respectively. The smaller value of MSE shows that SVD technique has a better result and relative infrequency of significant errors in prediction when compared to KNN. Lastly, the result of RMSE for KNN technique is 0.9749 while the SVD is 0.9025. This shows that the SVD has a lower value of RMSE compared to KNN technique which conveys greater consistency in prediction.

**TABLE 1.** Performance metrics across KNN technique and SVD technique

|  |  |  |
| --- | --- | --- |
| Performance metrics / Technique | KNN | SVD |
| MAE | 0.7499 | **0.6968** |
| MSE | 0.9504 | **0.8146** |
| RMSE | 0.9749 | **0.9025** |

To sum up, the SVD technique has demonstrated better performance across all three performance metrics as compared to the KNN technique. It provides precise and consistent predictions. This shows that the SVD technique is more suitable for this dataset.

Since SVD is better than KNN, the technique is used for recommending movies. The reason SVD works so well is that it carefully considers user preferences. By analyzing all the movie ratings submitted by users, it identifies patterns that can predict what content a user might like. Through SVD it is ensured that each person received a tailored list of their recommended movies. Figure 1 shows that SVD offers tailored recommendations of movies differently for each user making it an excellent option to help everyone discover movies they truly enjoy.

**Figure 1.** List of movies recommendation for user 1 and user 2

# CONCLUSION

In conclusion, the paper is concentrated on the examination of two machine learning algorithms which include the KNN and SVD. The aim was to evaluate the performance of these algorithms by comparing their performance against three metrics which are MAE, MSE and RMSE. Further, the results analysis revealed that the SVD performed better than KNN algorithm. It was also established as the superior method for movie recommendation because it has the ability to personalize recommendations to suit user tastes. This creates a diverse list of suggested movies for each user and therefore enhances the user satisfaction of the recommendation system. For future plan, in order to improve the performance of the current model, priority will be given to the exploration of ensemble method and hyperparameter tuning. Furthermore, there is an opportunity to implement deep learning technique, which can handle massive amounts of data and discover complex patterns.

# REFERENCES

1. A. Liu, Y. Zhang, and J. Li, “Personalized movie recommendation,” in *Proceedings of the 17th Association for Computing Machinery International Conference on Multimedia*, (*Association for Computing Machinery*, Beijing China, 2009), pp. 845–848.
2. N. Sharma, and M. Dutta, “Movie Recommendation Systems: A Brief Overview,” in *Proceedings of the 8th International Conference on Computer and Communications Management*, (*Association for Computing Machinery*, Singapore Singapore, 2020), pp. 59–62.
3. B.-B. Cui, “Design and Implementation of Movie Recommendation System Based on Knn Collaborative Filtering Algorithm,” ITM Web Conferences **12**, 04008 (2017).
4. G. Geetha, M. Safa, C. Fancy, and D. Saranya, “A Hybrid Approach using Collaborative filtering and Content based Filtering for Recommender System,” Journal of Physics: Conference Series **1000**, 012101 (2018).
5. V.X. Chen, and T.Y. Tang, “Incorporating Singular Value Decomposition in User-based Collaborative Filtering Technique for a Movie Recommendation System: A Comparative Study,” in *Proceedings of the 2019 the International Conference on Pattern Recognition and Artificial Intelligence*, (*Association for Computing Machinery*, Wenzhou China, 2019), pp. 12–15.
6. R. Ghawi, and J. Pfeffer, “Movie Genres Classification using Collaborative Filtering,” in *Proceedings of the 21st International Conference on Information Integration and Web-Based Applications & Services*, (*Association for Computing Machinery*, Munich Germany, 2019), pp. 35–44.
7. F. Furtado, and A. Singh, “Movie Recommendation System using Machine Learning,” International Journal of Research in Industrial Engineering (Online First), (2020).
8. Prince Praveen, Sagar Parmar, Praveen Goud, and Universal College of Engineering, “Movie Recommendation System Approaches,” International Journal of Engineering & Technology **V9**(04), IJERTV9IS040710 (2020).
9. M.  Chenna Keshava, P.  Narendra Reddy, S.  Srinivasulu, B.  Dinesh Naik, and JNTUA College of Engineering, Pulivendula, “Machine Learning Model for Movie Recommendation System,” International Journal of Engineering & Technology **V9**(04), IJERTV9IS040741 (2020).
10. B. Walek, and V. Fojtik, “A hybrid recommender system for recommending relevant movies using an expert system,” Expert Systems with Applications **158**, 113452 (2020).
11. S. Chawla, S. Gupta, and R. Majumdar, “Movie Recommendation Models Using Machine Learning,” in *2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, (Institute of Electrical and Electronics Engineers, Mathura, India, 2021), pp. 1–6.
12. L. Sama, H. Wang, and A. Makkar, “Movie Recommendation System Using Deep Learning,” in *2021 9th International Conference on Orange Technology (ICOT) (*Institute of Electrical and Electronics Engineers*, Tainan, Taiwan, 2021), pp. 1–4.*
13. N. Smitha, D. Anusha, C. Chaithanya, J. Sindhu, R. Tanuja, and H.S. Hemanth Kumar, “A Review on Movie Recommendation System Using Machine Learning,” in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, (Institute of Electrical and Electronics Engineers, Tirunelveli, India, 2021), pp. 769–773.
14. J. Ananda Babu, D.R. Vinay, B.V. Kumaraswamy, and C.C.S. Basavaraddi, “Machine learning based recommendation system on movie reviews using KNN classifiers,” Journal of Physics: Conference Series **1964**(4), 042081 (2021).
15. Y. Afoudi, M. Lazaar, and M. Al Achhab, “Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network,” Simulation Modelling Practice and Theory **113**, 102375 (2021).
16. G. Sunandana, M. Reshma, Y. Pratyusha, M. Kommineni, and S. Gogulamudi, “Movie recommendation system using enhanced content-based filtering algorithm based on user demographic data,” in *2021 6th International Conference on Communication and Electronics Systems (ICCES)*, (Institute of Electrical and Electronics Engineers, Coimbatre, India, 2021), pp. 1–5.
17. C.-Y. Lai, N. Hashim, and L. D. Krisnawati, "Hybrid-Based Movie Recommender System: Techniques, Case Studies, Evaluation Metrics, and Future Trends," *Journal of Informatics and Web Engineering* **4**, 90–110 (2025). https://doi.org/10.33093/jiwe.2025.4.3.5