

***A Mini-Project Report On***

**“Movie Recommendation”**

***Submitted By***

**Zalak Karnik**

**PRN: 1132220291**

**Harshwardhan Tambe**

**PRN: 1132220274**

**Ankita Kothari**

**PRN: 1132220973**

**Sahil Shah**

**PRN: 1132221070**

**F.Y. M.Sc. (Data Science and Big Data Analytics)**

**School of Computer Science**

**Faculty of Engineering and Technology**

**Dr. Vishwanath Karad MIT – World Peace University**

**Pune - 411038**

**Academic Year 2022-2023**

**Nov - 2022**

**Dr. Vishwanath Karad MIT WORLD PEACE UNIVERSITY, PUNE**

**SCHOOL OF COMPUTER SCIENCE**

**Certificate**

This is to certify that

**Ankita Kothari**

**PRN: 1132220973**

Of **M.Sc. (Data Science and Big Data Analytics)** successfully completed his/her

Mini-Project in

**“MOVIE RECOMMENDATION SYSTEM”**

to our satisfaction and submitted the same during the academic year 2021 - 2022 towards the partial fulfilment of degree of  **Master of Science in Data Science and Big Data Analytics** of Dr Vishwanath Karad MIT World Peace University under the School of Computer Science, MIT WPU, Pune.

**Prof. Dr. Shubhalaxmi Joshi Prof. Surabhi Thatte Prof. Surabhi Thatte**

**Associate Dean Program Head Assistant Professor**

**Faculty of Science School of Computer School of Computer**

**Science Science**

**MITWPU MIT WPU MIT WPU**

**ACKNOWLEDGEMENT**

In the accomplishment of this project, I would like to express my special thanks of gratitude to my teachers **Prof. Surabhi Thatte**, School of Computer Science, Dr. Vishwanath Karad MIT World Peace University whose valuable guidance has been the ones that helped me patch this project and make it full proof success. His/Her suggestions and instructions have served as the major contributor towards the completion of the project.

As we were working in a group, I would like to thank my group members for their fabulous support throughout the completion of the project. We learned a lot of things during this period, as it was hard to work in this time of adversity; we were in touch with each other throughout the period and shared everything which was important from the aspect of our project. As this project was completed by staying at home, I would also like to thank our families for their cooperation and for providing facilities to us.

**Student Name**

**PRN.**

|  |  |
| --- | --- |
| **Introduction** |  |
| Domain Name | 4 |
| Motivation | 4 |
| Problem Statement | 4 |
|  |  |
| **Literature Survey** | 5 |
|  |  |
| **Solution Design** |  |
| Solution Approach | 7 |
| Technology Stack | 7 |
| Design Model | 8 |
|  |  |
| **Solution Implementation and Results** |  |
| Obtaining Data | 11 |
| EDA | 13 |
| Pre-Processing | 14 |
| Algorithms Used | 16 |
| Results | 20 |
|  |  |
| **Conclusion and Future Work** |  |
| Conclusion | 21 |
| Future Work | 21 |
|  |  |
| **References** | 22 |

**Contents**

**1) Introduction:**

* 1. **Domain Name:**

Nowadays, the recommendation system has made finding the things easy that we need. Movie recommendation systems aim at helping movie enthusiasts by suggesting what movie to watch without having to go through the long process of choosing from a large set of movies which go up to thousands and millions that is time consuming and confusing. In this article, our aim is to reduce the human effort by suggesting movies based on the user’s interests. To handle such problems, we introduced a model combining both content-based and collaborative approach. It will give progressively explicit outcomes compared to different systems that are based on content-based approach. Content-based recommendation systems are constrained to people, these systems don’t prescribe things out of the box, thus limiting your choice to explore more. Hence, we have focused on a system that resolves these issues.

* 1. **Motivation:**

Movie recommendation system is to provide personalized recommendations to users based on their previous viewing history, preferences, and behaviors. With the increasing availability of vast amounts of data, including user ratings, reviews, and browsing history, machine learning algorithms can effectively analyze and understand user behavior patterns to recommend movies that match their preferences. A good movie recommendation system can enhance user experience and engagement, increase customer loyalty and retention, and generate revenue for streaming platforms. By suggesting movies that match users' interests, the system can help users discover new content and keep them engaged on the platform for longer periods

**1.3 Problem Statement:**

Given a dataset of user ratings and viewing history, develop a movie recommendation system that can effectively suggest relevant movies to users based on their preferences and behavior patterns. The system should consider various factors such as movie genres, cast, crew, release year, and user ratings to generate personalized recommendations..

**Solution Design**

* 1. **Solution Approach**

We Will do both content based filtering and collaborative based filtering First, we have performed EDA on all dataset to understand the data if it contains any null value, missing value, any outliers, etc. Then we will perform pre-processing and data cleaning. We have done PCA also on the Dataset and after that We have applied various algorithms to predict

**Cosine similarity** is another distance metric commonly used in movie recommendation systems to measure the similarity between two movies

To calculate cosine similarity, the features of the movies are first converted into vectors. The cosine similarity between two movie vectors A and B is calculated using the following formula:

**cosine\_similarity(A, B) = (A . B) / (||A|| ||B||)**

Then we have used another similarity algorithm **Euclidean similarity** is a distance metric used in movie recommendation systems to measure the similarity between two movies.

* 1. **Technology Stack**

We have used Jupyter notebook and python for EDA, pre-processing as well as for model building. In short we have done all our machine learning algorithm on jupyter Notebook

* 1. **Designing model**

Data collection and preprocessing: The first step in designing a movie recommendation system is to collect and preprocess data. This may involve collecting data on movie titles, genres, cast and crew, ratings, and other metadata from various sources, such as movie databases and user reviews. The data may need to be cleaned, formatted, and standardized to ensure consistency and accuracy.

Feature engineering: Once the data has been collected and preprocessed, the next step is to extract features that can be used to identify patterns and make recommendations. This may involve creating feature vectors that represent each movie, based on the metadata collected in the previous step. Feature engineering may also involve extracting features from user profiles, such as viewing history and preferences.

Model selection: The next step is to select a suitable model for the recommendation system. There are several types of models that can be used, including collaborative filtering, content-based filtering, matrix vectorization, and KNN algorithm ,SVD algorithm. The choice of model will depend on the specific requirements of the system and the available data.

Training the model: Once the model has been selected, the next step is to train it using the feature vectors and other relevant data. This may involve optimizing hyperparameters, such as the learning rate and regularization, to improve the accuracy of the model.

**4. SOLUTION IMPLEMENTATION AND RESULT**

* 1. **Obtaining Data**

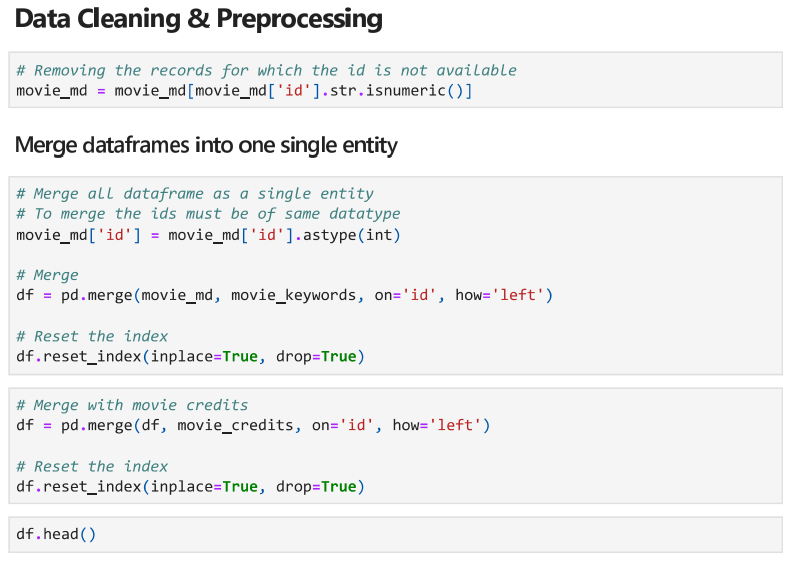
These files contain metadata for all 45,000 movies listed in the Full MovieLens Dataset. The dataset consists of movies released on or before July 2017. Data points include cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, TMDB vote counts and vote averages.

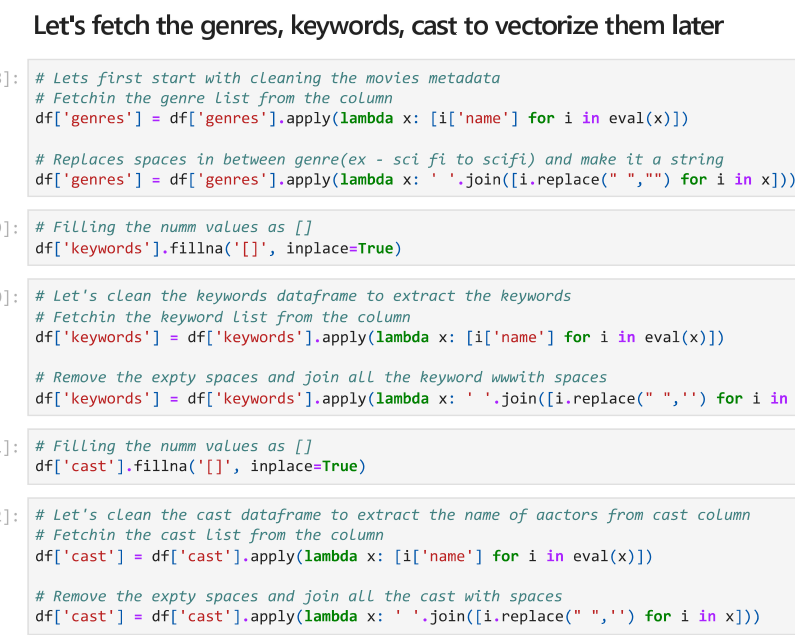
This dataset also has files containing 26 million ratings from 270,000 users for all 45,000 movies. Ratings are on a scale of 1-5 and have been obtained from the official GroupLens website.

We have collected data from online source i.e

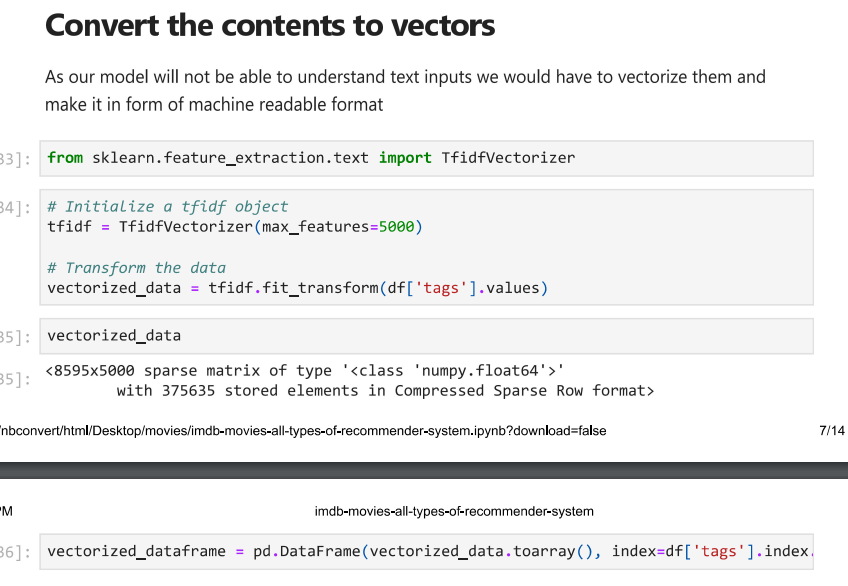
<https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset>

* 1. **EDA**
  2. **Pre-Processing**







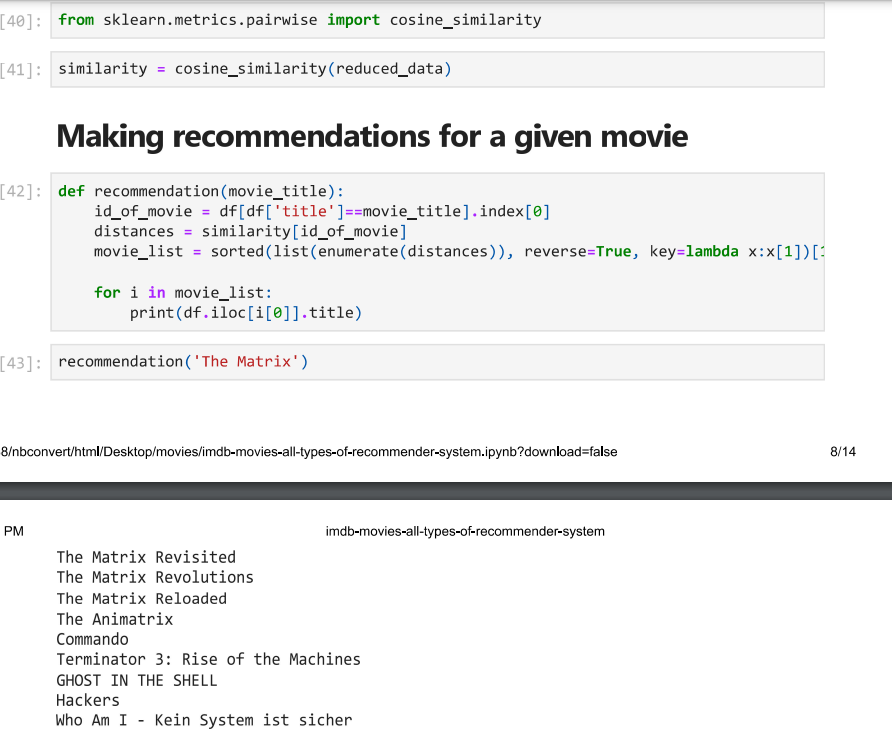


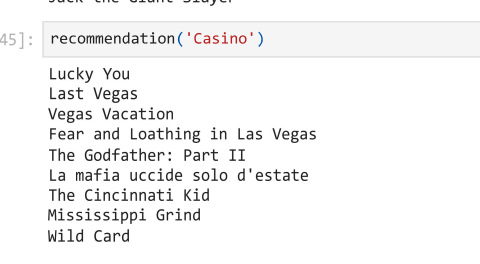
* 1. **Algorithms Used**

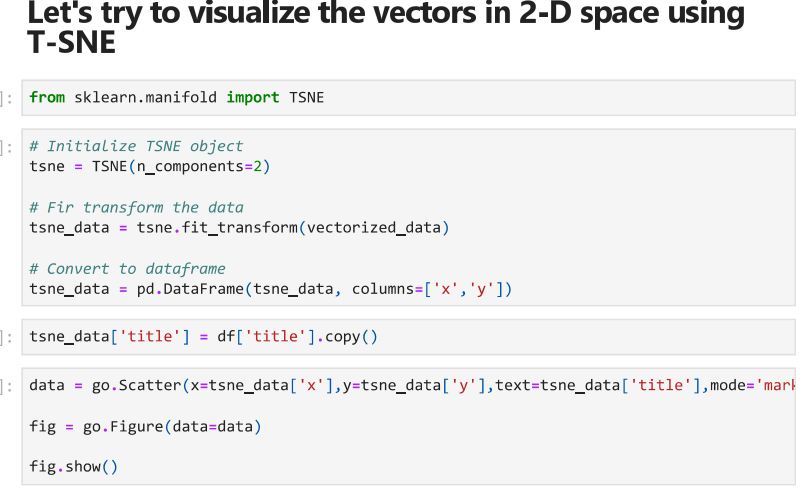
We have used TSNE which is a t-Distributed Stochastic Neighbor Embedding is a non-linear dimensionality reduction algorithm used for exploring high-dimensional data.After that we have also used SVD algorithm which is basically a matrix factorization technique, which decomposes any matrix into 3 generic and familiar matrices.After doing this 2 algorithms we have performed KNN algorithm The KNN algorithm can compete with the most accurate models because it makes highly accurate predictions. Therefore, you can use the KNN algorithm for applications that require high accuracy but that do not require a human-readable model. The quality of the predictions depends on the distance measure.

* 1. **Results**

**COSINE SIMILARITY**

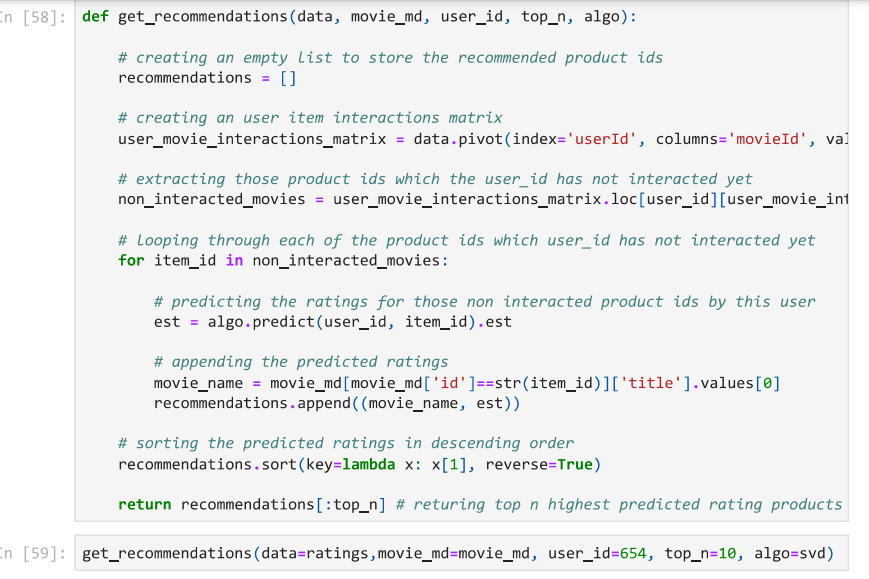
****

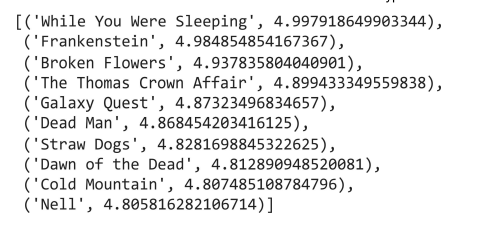
****



**SVD**







**KNN**



1. **CONCLUSION AND FUTURE WORK**
   1. **Conclusion**

movie recommendation systems have the potential to provide users with a personalized and engaging movie viewing experience. By focusing on personalization, data quality, hybrid models, evaluation metrics, and user feedback, developers can create effective and accurate recommendation systems that meet the needs of users.

* 1. **Future Work**

1.Multilingual recommendation: Recommending movies based on user preferences and in their native language could help to increase engagement and satisfaction of users who speak languages other than English.

2. Incorporating temporal dynamics: The patterns of user preferences and interests may change over time, and recommendation systems need to incorporate these temporal dynamics to provide accurate recommendations.