Conestoga College

School of Applied Computer Science & Information Technology

**SENG8080 - Case Studies Big Data**

Section – 2

Group – 9

Movie Ratings Analysis Project

Akashraj Kalathingal (8834315)  
Vivek Subramaniyan (8831747)

Shameem Begum (8893341)

**Abstrac****t:**

This project revolves around analyzing and harnessing insights from the TMDB ratings of movies and TV series by aggregating a diverse range of data from multiple online sources.

The project utilised online scraping techniques through the usage of Google Chrome's WebDriver to gather information from many movie databases and streaming services about different genres, ratings, reviews, cast, crew, and user preferences. After the data was painstakingly cleaned up and combined into a Microsoft SQL Server database, an application was created for further research and visualisation. With the intention of reinventing the process of finding films and television shows and providing viewers with a more rich and customised experience, this tool offers individualised and customised insights based on user behaviour and interests. By combining web scraping, data processing, and skillful database administration, this project creates an analytical framework around the TMDB ratings that goes beyond conventional methods of analysing films and TV shows.

**Table of contents**

[Introduction 4](#_Toc150024627)

[Data Research and Integration 4](#_Toc150024628)

[Data Collection 4](#_Toc150024629)

[Data Storage and Maintenance 4](#_Toc150024630)

[Data Quality – Work in Progress 4](#_Toc150024631)

[Data Analysis and Visualization – Work in Progress 4](#_Toc150024632)

[Extension – Work in Progress 5](#_Toc150024633)

[Proposed Allocation Project Team Roles 5](#_Toc150024634)

[Project Timeline 5](#_Toc150024635)

[References ………………………………………………………………………………………...7](#_Toc150024636)

# Introduction

This project represents a significant endeavour in the field of digital movie consumption, with the goal of revolutionising the way in which viewers interact with films according to TMDB ratings. Using cutting-edge web scraping technology, this project compiles information from many internet sites and movie databases. The project carefully collects a vast range of data on genres, TMDB ratings, reviews, cast, crew, release dates, and user preferences—with an emphasis on movies and TV shows—using Google Chrome's WebDriver.

After going through a thorough process of refining and consolidation, the data is finally combined into a single dataset that is stored in a Microsoft SQL Server database. This makes the data easily accessible and well-organized.

Furthermore, the project goes beyond simple data collection to provide an advanced application designed for detailed analysis and visualisation. With a focus on TMDB ratings, this programme provides users with an easy-to-use interface for investigating the nuances of the film industry. It does this by revealing patterns, relationships, and trends in the field of films and TV shows.

Mostly, the app is essential because it makes tailored suggestions for films and TV shows according on user tastes, past viewing habits, and popular TMDB rating trends. Through the integration of web scraping, data processing, and skillful database administration, this project seeks to go beyond the traditional movie-watching experience, establishing a user-focused platform that reinterprets the meaning of cinematic interaction via TMDB ratings-based analysis.

# Data Research and Integration

The project chose two data sources as the main source for data research and focused on TMDB ratings of films and TV shows. Following collection, the data is collated and saved as a CSV file. After that, the datasets are subjected to data cleansing processes using Python scripts. The project utilizes publicly accessible and regularly updated data from Kaggle and data.world, meeting specific criteria: availability, regular updates, diverse content, and a minimum of two distinct data sources, ensuring varied and current datasets for analysis.

# Data Collection

# The project explores the analysis of TV shows and films using TMDB ratings. Using websites such as Kaggle and data.world, WebDriver and Google Chrome-specific web scraping are used in the data collecting process to gather a variety of data. This procedure is automated using a Python script that retrieves vital data from the vast Internet Movie Database (IMDB), including movie ID, title, release year, and ratings.

# This dataset, which includes metadata for 45,000 films published by July 2017, is a mirror of the Full MovieLens Dataset. Cast, crew, narrative keywords, budget, revenue, posters, release dates, languages, production companies, nations, and average votes on TMDB are just a few of the numerous data pieces it contains. Furthermore, the collection contains 26 million ratings from 270,000 individuals who rated films on a scale of 1 to 5. These ratings were obtained from Kaggle. This enormous dataset provides a solid basis for in-depth study and analysis of TMDB-rated motion pictures and television shows.

Collections gathered via data.world and Kaggle may have different formats, data types, and column names. Cleaning up data before to merging include fixing duplicates and outliers, standardising formats, and addressing missing values to ensure consistency. It is ensured that the column names for comparable data are uniform by renaming 'name' to 'Title.' Data is cleaned and standardised before being merged based on a shared identity. This process may involve data transformation or aggregation to create a coherent, single dataset.

Combining data from many sources or contributors is known as "merging datasets" from sites like Kaggle and data.world. This process frequently calls for correcting inconsistencies in column names, formats, or content. This procedure entails gathering, organising, and integrating data as well as standardizing column names and combining datasets using shared IDs. The objective is to bring the data together cohesively for thorough analysis and investigation.

# Data Storage and Maintenance

Data needs to be stored somewhere, not just today but for the future. Some considerations here are regulatory, and some are practical (i.e. how long to store the data).

**Data Storage:** MySQL, Google Collab (csv filed saved in GC).

In order to evaluate data storage choices for the project that analyses TMDB ratings of films and TV shows, platforms such as Google Colab and MySQL are evaluated.

1. **Google Colab** (CSV files saved in Google Colab):

GPUs and TPUs may be accessed for free via Google Colab, a cloud-based platform. It makes working together possible and makes running Python programmes easier.

The Google Colab environment makes storing CSV data accessible, sharing simple, and allowing for easy interaction with Python scripts for processing and analysis.

However, scalability and long-term data preservation may be limited with Google Colab storage, particularly for large datasets or long-term storage requirements.

1. **MySQL**

Strong relational database management system (RDBMS) MySQL provides organised data management, retrieval, and storing features.

Scalability, effective query processing, and strong data security characteristics are offered by using MySQL as a data storage solution.

It can handle massive amounts of movie and TV series data since it makes organised data storage easier, especially when managing several qualities and connections.

MySQL has customisable data retention rules that may be tailored to meet future scalability requirements and regulatory compliance.

**Tools for Storage:**

Google Cloud Storage is going to be our data repository. This platform provides robust, safe, and scalable storage options. Utilising Google's infrastructure helps with future storage and retrieval requirements by ensuring dependable data management, accessibility, and smooth connection with other Google services.

# Data Quality

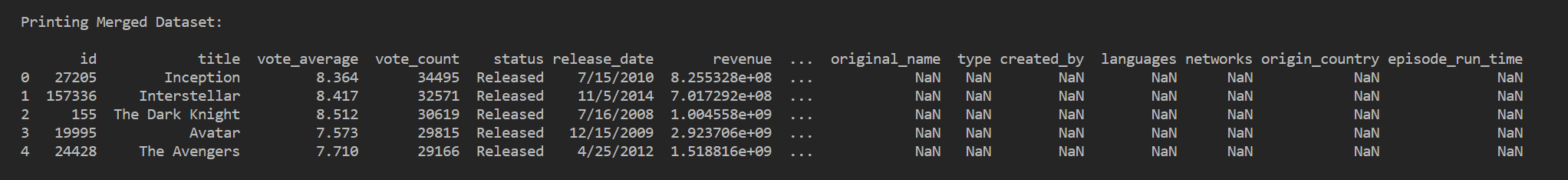
# The project that examines movie and TV show ratings on TMDB guarantees the upkeep of high-quality data. The produced datasets are subjected to extensive data cleaning techniques in order to improve the accuracy and organisation of the database. With the help of specialised Python scripts, this approach expedites and simplifies database maintenance. By methodically addressing mistakes, duplications, and inconsistencies in the datasets, these programmes guarantee a dependable and well-organized database. The automatic Python scripts maintain the correctness of the database and improve overall data maintenance efficiency.

Dataset Head from two Sources:

A black background with many small squares

Description automatically generated with medium confidence

Combined Dataset:



# Data Cleaning and Validations

# Handling Missing Values: The mean is used to imputation numerical columns, while the mode is used to impute categorical columns, with the ability to modify based on needs and data.

# Dropping Duplicates: To enhance data quality, duplicate rows are eliminated.

# Data Validation: Validation procedures are included in the code.

# Saving Cleaned Data: After being cleaned, datasets are stored as fresh CSV files, which serve as checkpoints for quality enhancements and post-missing value handling. This makes future data analysis easier.

Cleaned Dataset:

# 

# Data Analysis and Visualization

In this section you will discuss the findings and answers to the questions you set out to answer in your introduction and abstract.

1. Displaying the Total Percentage of Popularity, Vote Average, and Vote Count Based on Movie Status

**Insights:**

Analyze the distribution of popularity, vote average, and vote count for movies based on their status (e.g., Released, Post Production, Rumored).

Understand the overall distribution of these metrics for different movie statuses.

**Recommendations**:

Use pie charts or bar plots to visually represent the percentage distribution.

Provide insights into how popular and well-received movies are at different stages, helping stakeholders understand audience engagement during the various phases of movie production.

A screen shot of a graph

Description automatically generated

2. Distribution of Top 10 Production Companies in 10 Different Countries Based on Popularity

**Insights:**Explore the distribution of the top 10 production companies in different countries based on the popularity of their movies.Identify which production companies are most successful in specific regions.

**Recommendations:** Utilize a stacked bar chart or a heatmap to represent the popularity of top production companies in different countries.

Offer recommendations on potential partnerships or collaborations based on the success of production companies in specific regions.A screenshot of a computer

Description automatically generated

3. Top 10 Movies Based on Highest Vote Count

**Insights**:

Identify and showcase the top 10 movies with the highest vote counts.

Understand which movies have garnered the most engagement from the audience.

**Recommendations**:

Create a bar chart or a table to display the top movies based on vote count.

Provide insights into the characteristics or commonalities among these top-voted movies, aiding in the understanding of audience preferences.

A blue squares with black text

Description automatically generated

4. Relationship Between Popularity and Vote Count Based on Genres

**Insights**:

Investigate the relationship between the popularity and vote count for movies based on different genres.

Understand if certain genres tend to have a higher correlation between popularity and audience engagement.

**Recommendations**:

Use scatter plots or bubble charts to visualize the relationship between popularity and vote count for each genre.

Offer recommendations for genres that exhibit a strong correlation, potentially guiding content creators on genre-specific strategies for increasing audience engagement.

A graph with blue circles and white text

Description automatically generated

# Extension

The project now uses a small amount of data, which is kept in manageable storage systems like MySQL or Google Collab. But a scalable solution like Hadoop becomes essential with an envisaged growth and the expected rise in data volume owing to the continuous aggregation of varied TMDB ratings of films and TV shows from multiple internet sources.

Large datasets may be handled with flexibility and scalability thanks to Hadoop, a distributed processing and storage system. Hadoop's distributed file system (HDFS) enables smooth scalability as data grows, handling the inflow of information without sacrificing efficiency. Its MapReduce programming approach also makes parallel processing easier, which makes it possible to run queries on large-scale databases quickly.

The project's plan to migrate to Hadoop for storage is in line with its goal of increasing the number and variety of data sources. The project can manage the growing amount of data, carry out intricate analytics, and run big data queries by utilising Hadoop's capabilities. This ensures scalability and resilience while managing large and varied TMDB ratings datasets of films and TV shows. This change makes it possible to store, process, and analyse massive amounts of data efficiently, which helps the project achieve its goals of gaining insights and giving consumers a better cinematic experience.

Future Work

Future enhancements to the TMDB ratings analysis project might concentrate on a number of important areas. Using cutting-edge big data technologies such as Apache Spark or Apache HBase will meet the processing and storage requirements of the growing dataset. The recommendation system may be improved by using machine learning algorithms to provide individualised suggestions depending on user behaviour. By using natural language processing techniques, the recommendation engine may be improved by analysing the feelings expressed in reviews. The collection will be enhanced by adding social media trends and new platforms, and working with specialists can provide deeper insights into certain genres or demographics. Maintaining data accuracy and scalability requires ongoing optimisation of data processing and storage methods.

Moreover, the dataset may be enhanced by adding information from developing platforms, social media trends, and viewer comments in addition to TMDB ratings. This would provide users a more complete picture of user preferences and market trends.

# Proposed Allocation Project Team Roles

* Data acquisition and integration: duties include managing data collecting from platforms such as Kaggle, data.world, and others, handling data scraping tools, and compiling data from various web sources.

Team Member: Akashraj

* Data Cleaning and Preprocessing: Managing missing values, duplicates, outliers, and standardising formats is your responsibility in order to ensure data correctness, consistency, and cleanliness.

Team Member: Vivek

* Database Management and Storage:

Responsibility: Overseeing data storage strategies, managing databases (MySQL, Google Colab, Google Cloud Storage), and ensuring proper data maintenance.

Team Member: Shameem

* Data Analysis and Visualisation: Accountabilities include analysing TMDB ratings data in-depth, making visualisations, investigating trends, and interpreting results.

Team Member: Shameem

* Machine Learning and Recommendation System:  
  Accountability: Putting machine learning algorithms into practice, improving the recommendation engine by analysing user behaviour, and using natural language processing (NLP) approaches to analyse sentiment in reviews.

Team Member: Akashraj

* Project Management and Coordination

Accountability: Managing project schedules, guaranteeing team communication, setting up gatherings, and monitoring overall development.

Team Member: Vivek

.

# Project Timeline

|  |  |  |
| --- | --- | --- |
| **Date** | **Deliverable** | **Responsible** |
| Oct 20 | Data Collected and planned | Akashraj |
| Oct 20 | Data Extraction | Vivek |
| Nov 4 | Data visualization into Tableau | Shameem |
| Nov 8 | 1st Draft Circulated to Team | Vivek |
| Nov 11 | 1st Draft of Presentation Circulated | Akashraj |
| Nov 11 | User testing by the team and errors/refinements identified. | Shameem and Akashraj |
| Nov 17 | Final Adjustments made and checked | All members |
| Nov 18 | Submit Deliverables | Vivek |

## 

# Git repository

## https://github.com/akashR011/SENG8080-2-field\_project-final

# References

* + - 1. Jayalakshmi, S., Ganesh, N., Čep, R., &amp; Senthil Murugan, J. (2022, June 29). Movie Recommender Systems: Concepts, methods, challenges, and future directions. MDPI. <https://www.mdpi.com/1424-8220/22/13/4904>
      2. "A hybrid recommender system based on link prediction for movie baskets analysis" by A. Pereira, M. R. Araújo, and J. Gama
      3. "An improved collaborative movie recommendation system using computational intelligence" by Z. Wang, X. Yu, N. Feng, and Z. Wang
      4. (n.d.). Welcome. Millions of movies, TV shows and people to discover. Explore now. TMBD. <https://www.themoviedb.org/>
      5. (n.d.). Full TMDb TV Shows Dataset 2023 (150K Shows). Kaggle. <https://www.kaggle.com/datasets/asaniczka/full-tmdb-tv-shows-dataset-2023-150k-shows>
      6. Chen, M., Mao, S., & Liu, Y. (2014). Big Data: A Survey. Mobile Networks and Applications, 19(2), 171–209. <https://doi.org/10.1007/s11036-013-0489-0>
      7. Zikopoulos, P., Eaton, C., Deroos, D., Deutsch, T., & Lapis, G. (2011). Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data. McGraw-Hill Osborne Media.
      8. Jain, S., & Jain, S. (2016). Big Data Analytics Beyond Hadoop: Real-Time Applications with Storm, Spark, and More Hadoop Alternatives. Apress