# Movie Recommendation System

## Milestone I report

### Objective:

The project objective is to provide movie recommendations to use based on the large volume of data collected from different sources. The project may include a conversation agent, using which a user can get insights into the large volume of data.

### Tool Type:

Recommendation System

### Data Used:

- IMDB movie datasets (by genre) (https://www.kaggle.com/datasets/rajugc/imdb-movies-dataset-based-on-genre)
- MovieLens Datasets (https://grouplens.org/datasets/movielens/)
- TMDB Dataset (https://www.kaggle.com/datasets/asaniczka/tmdb-movies-dataset-2023-930k-movies)

### Tech Stack:

I have listed the tech stack which I used, along with the tech stack which I will be using for future work.

- Programming Language: Python
- Data storage: SQLite
- Data manipulation: Pandas, NumPy
- Visualization: Matplotlib, Seaborn

### Project Timeline:

- Milestone 1: Data Collection, Preprocessing, and Exploratory Data Analysis (EDA) Timeline: February 5, 2025 – February 23, 2025 (2 weeks)
  - Feb 5 – Feb 7: Identify and acquire datasets
  - Feb 8 – Feb 10: Verify dataset accessibility, licensing, and documentation
  - Feb 11 – Feb 15: Data preprocessing (handling missing data, outliers, scaling)
  - Feb 16 – Feb 19: Exploratory Data Analysis (EDA) (statistical summaries, visualizations)
  - Feb 20: Finalizing the EDA report and project documentation
  - Feb 23: Submit Milestone 1 deliverables

- Milestone 2: Feature Engineering, Feature Selection, and Data Modeling Timeline: February 21, 2025 – March 21, 2025 (5 weeks)
    - Feb 22 – Feb 26: Feature engineering (creating new features)
    - Feb 27 – Mar 2: Feature selection
    - Mar 3 – Mar 6: Splitting the dataset and initial model training
    - Mar 7 – Mar 12: Model tuning and hyperparameter optimization
    - Mar 13 – Mar 18: Model evaluation and comparison
    - Mar 19 – Mar 20: Preparing Milestone 2 report
    - Mar 21: Submit Milestone 2 deliverables
- Milestone 3: Evaluation, Interpretation, Tool Development, and Presentation Timeline: March 24, 2025 – April 23, 2025 (5 weeks)
    - Mar 24 – Mar 28: Evaluate model performance on test data
    - Mar 29 – Apr 2: Interpret model results and address biases
    - Apr 3 – Apr 7: Develop an interactive dashboard for visualizations
    - Apr 8 – Apr 12: Implement the final recommendation system
    - Apr 13 – Apr 17: Tool testing and debugging
    - Apr 18 – Apr 20: Prepare final report and GitHub repository updates
    - Apr 21 – Apr 22: Record demo video and finalize presentation
    - Apr 23: Submit Milestone 3 deliverables

## EDA report:

The datasets which are used by me are relatively huge. However, I believe that these datasets are perfect for movie recommendations. As this data is coming from 2 of the most popular and reliable sites (IMDB and TMDB) and shares common data among them up to some extent.

The 3rd dataset (movielens) is also quite important, as it provides me with the direct mapping of the first 2 datasets. In addition, it also provides user tags for different movies, which could be more useful for training my model during upcoming milestones of the project.

Below you will find some analysis about each dataset. Various steps of EDA (such as min-max scaling, statistic analysis, removing outliers, etc.) have been performed on the dataset and respective stages can be found while running the program (through print statements)

### IMDB Dataset

- Dimension (After merging all genre files):
    - Rows: 368300, Columns: 14
    - This data had some redundant columns, which I dropped (description, gross (with majority value as NA)

- Some columns also had outlier (negative values for runtime) which were handled using min imputation or other methods
- Some columns contained Null or NA values, which were filled by “Unknown” keyword for strings and also some rows were filtered out (for e.g. the rows having 0 ratings and vote counts)
- After removing duplicates, handling and removing noisy data, the dataset statistics were observed using the describe function.

Column Names:

| **file_name** | **number_of_rows** | **number_of_cols** |
|---|---|---|
| action.csv | 52452 | 14 |
| adventure.csv | 25664 | 14 |
| animation.csv | 8419 | 14 |
| biography.csv | 8289 | 14 |
| crime.csv | 35852 | 14 |
| family.csv | 17095 | 14 |
| fantasy.csv | 17163 | 14 |
| film-noir.csv | 986 | 14 |
| history.csv | 8996 | 14 |
| horror.csv | 36682 | 14 |
| Mystery.csv | 18960 | 14 |
| romance.csv | 52617 | 14 |
| scifi.csv | 16557 | 14 |
| sports.csv | 5292 | 14 |
| thriller.csv | 53365 | 14 |
| war.csv | 9911 | 14 |

TMDB Dataset

- Dimension
  - Rows: 1180936, Columns: 24

- This dataset contains some similar and some different fields from the IMDB dataset.
- The data also contains information about directors and actors, which could be used for more insightful movie recommendations.
- Also, it contains a “popularity” column. Which can be used alongside “ratings” to draw correlation and predict if the movies met the expectations or not.

Variable Description:

| **column_name** | **default_type** | **change_to** | **col_use** | **will_be_dropped?** |
|---|---|---|---|---|
| adult | bool | | Checks if a film is adult only film | |
| backdrop_path | object | string | Provides link for backdrop_path | Yes |
| budget | int64 | | Values highlighting movie budget | |
| genres | object | | List of genres associated with movie | |
| homepage | object | string | Link of movie homepage | Yes |
| id | int64 | | Represents movie id | |
| imdb_id | object | int64 | Can be useful for imdb dataset mapping | |
| keywords | objects | | Provides other keywords for movies | Yes (This might be included later to add more robustness to recommendation. But at initial stage want to |

| | | | | |
|---|---|---|---|---|
| | | | | remove it) |
| original_language | object | string | Original language in which the movie was produced | |
| original_title | object | string | Original title of the movie | |
| overview | object | string | A short overview about the movie | |
| popularity | float64 | | Represents popularity of the movie among audience | |
| poster_path | object | string | Link for fetching poster path | Yes |
| production_companies | object | | List of production companies involved | |
| production_countries | object | | List of production countries | |
| release_date | object | date | Can be used to derive movie release year | |
| revenue | int64 | | Revenue generated can be key factor in movie recommendation | |
| runtime | int64 | | Runtime of the movie | |
| spoken_languages | object | | List of languages in which movie is | |

| | | | | |
|---|---|---|---|---|
| | | | available | |
| status | object | category | Contains movie status | |
| tagline | object | string | Tagline of the movie | |
| title | object | string | Represents movies title | |
| vote_average | float64 | | Average vote (aka ratings) given by audience | |
| vote_count | int64 | | Size of audience, who rated the movie | |

Movielens Dataset

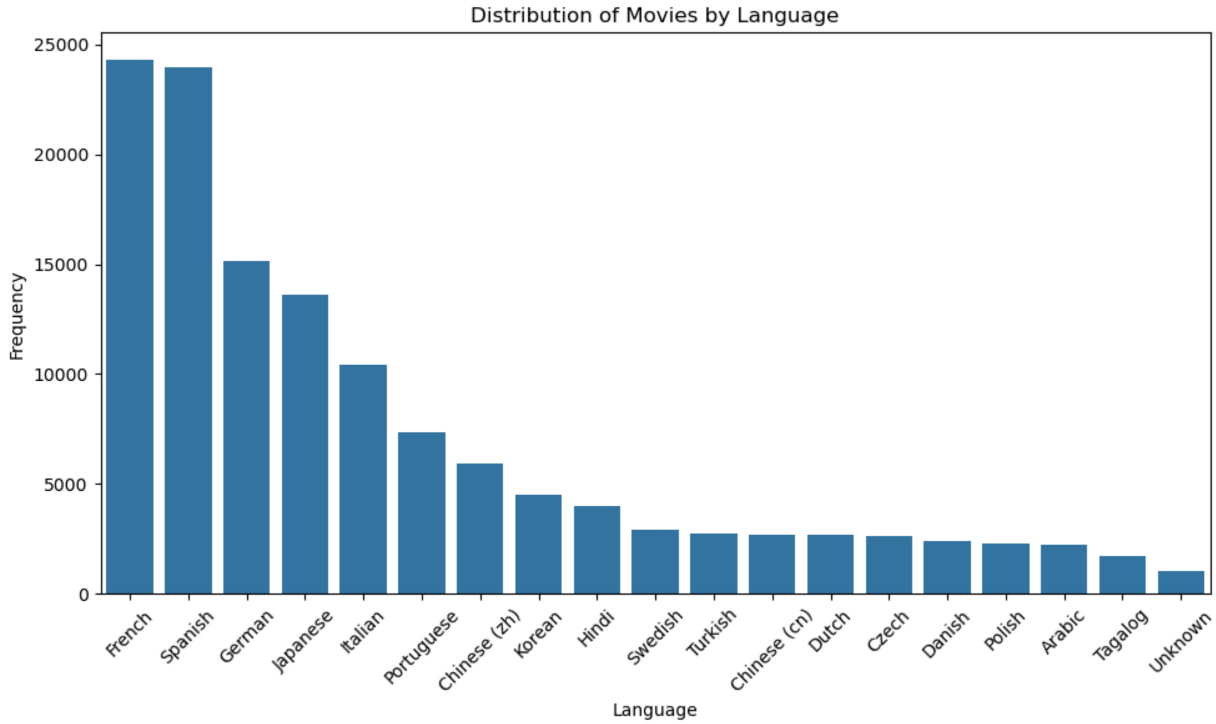
- Dimension
  - Rows: 2000072, Columns: 4
  - This datasets main purpose is to establish link between other 2 datasets and more insights on user data and preference (based on user tags)
  - This data was pretty much cleaned already. I just had to do some joins on it, to combine different csv files into one.

| **file_name** | **number_of_rows** | **number_of_cols** |
|---|---|---|
| links.csv | 86537 | 3 |
| tags.csv | 2328315 | 4 |

Plots

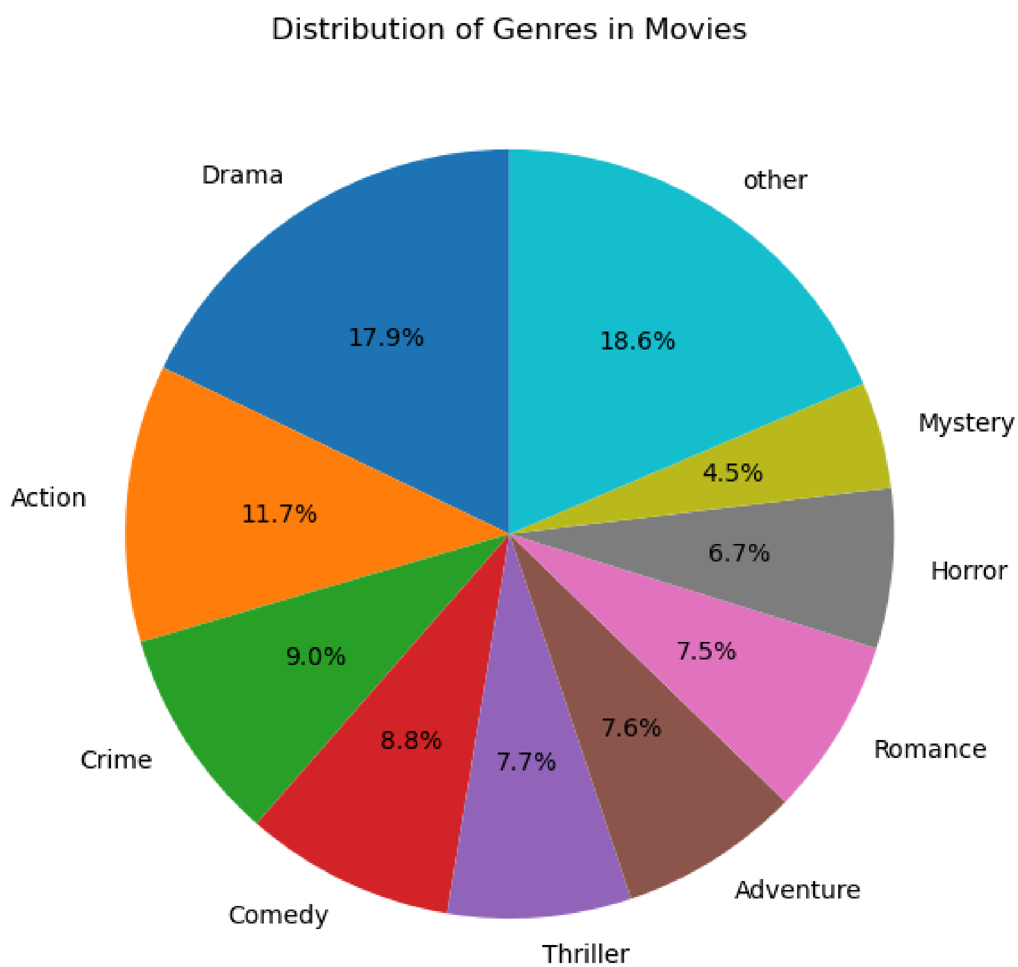
- **Most popular languages among all movies**

  This graph showcases which languages are most popular in movies. (English tops the list of course, but it was creating an imbalance bar chart). To wanted to have a good overview of it, I have plot the graph for top 20 languages (except english)

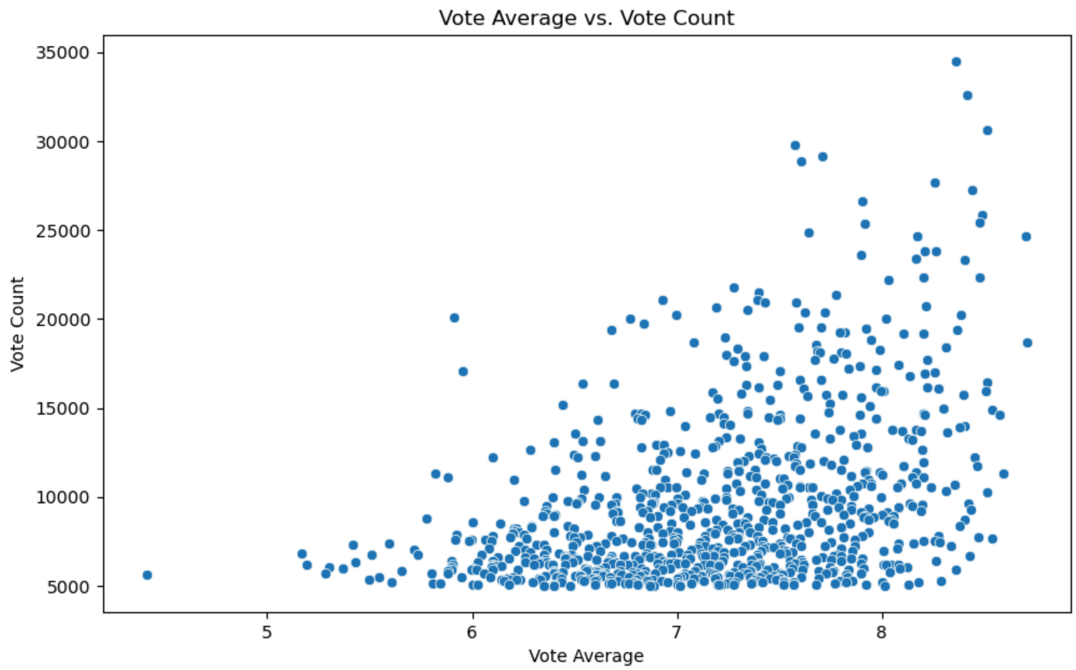



- **Top 10 genres across all movies**

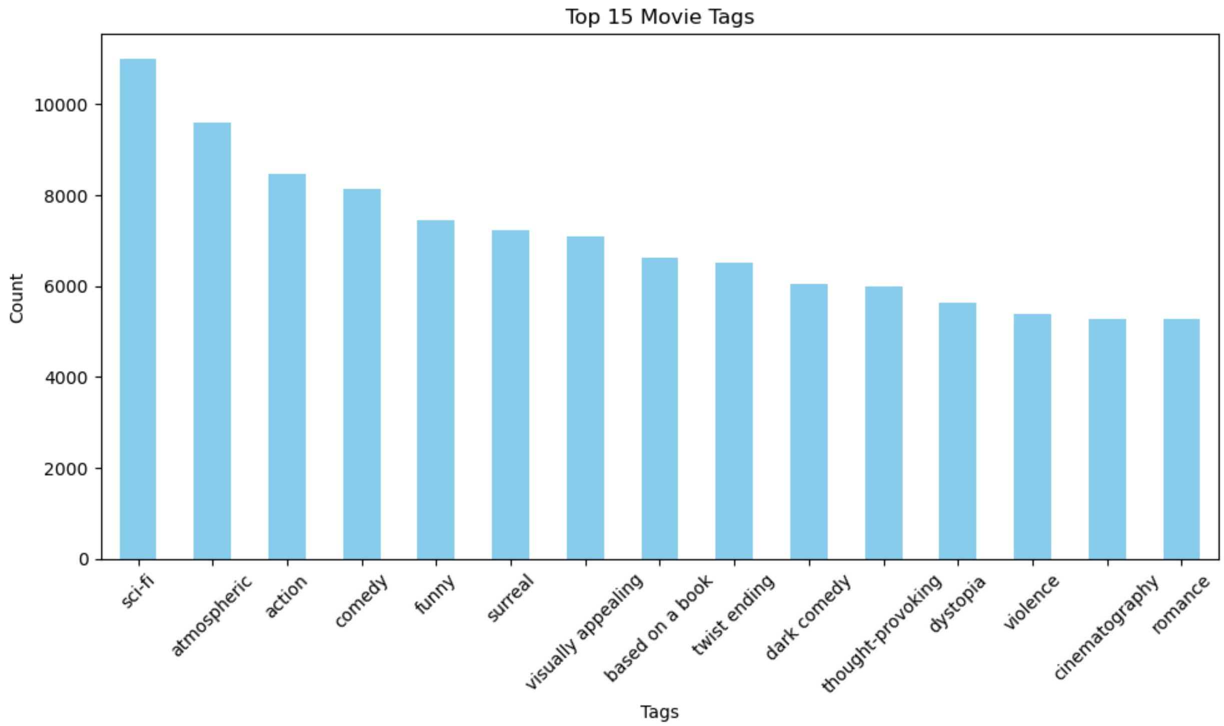
  The pie-chart showcases the distribution of top 10 genres in the data (by movie count). As expected some of the most common genres (Drama, Action, Crime, Comedy) shares nearly 45% of the the total tags,



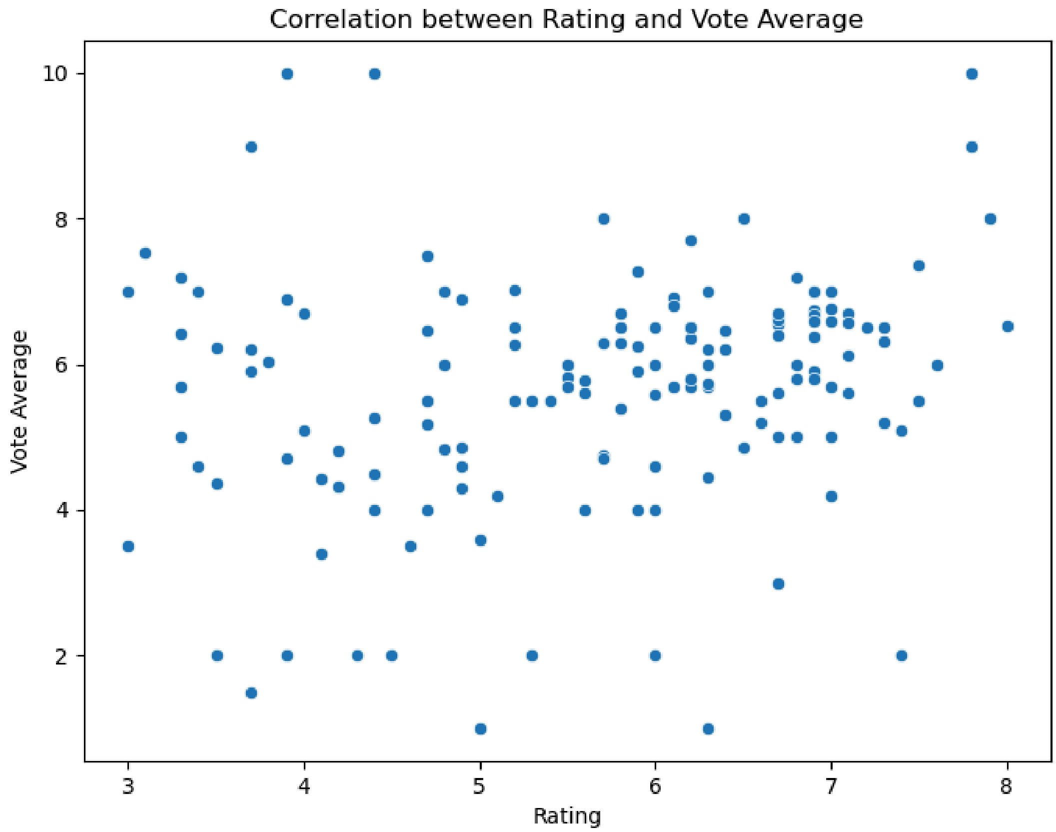

- **Vote average vs Vote count correlation analysis (min. vote count 5000)**
  I wanted to see how the correlation works for vote average based on vote count. Does the vote average fall too much for larger numbers of vote counts? No, it can be observed that the scatter plot is pretty much distributed on the right end, despite considering more than 5000 votes counts only.




- **Most popular user tags (across all movies)**
  The graph represents the top tags given by users across all the movies available. As we can see they differ from the “genres”. The “genres” are pre decided, but these user tags are equally important to understand user preferences against ratings.




- **Vote average (IMDB) correlation with ratings (TMDB) (random 150 records)**
  I was curious to know if the ratings across platforms differ a lot or are they usually the same? Due to the large volume of data, scatter plot don't look good on the whole dataset, so I used scatter plot on sample of 150 records.




## Conclusion:

The IMDB and TMDB ratings sometimes differ a lot and we can analyze those parameters more in depth in upcoming milestones. However, due to the reliable sites the data volume and veracity seems valuable for this project implementation.