Project Report For CS661: BIG DATA VISUAL ANALYTICS
2023-2024 Semester II
Project Title: CineScope: Exploring MovieLens Data Through Visual Analytics

Team members: Jeswaanth Gogula (231110020), Manoj Daram (231110029), Durga Sakar Praveen Reddy (231110013), komala Yaramareddy (231110024), Abhishek Dubey (231110003), Naman Baranwal (231110604), Vikram Kumar (231280003), Sovan Sahoo (231280002),

Member emails: jeswaanth23@iitk.ac.in, manojdaram23@iitk.ac.in, durgaspr23@iitk.ac.in, komala23@iitk.ac.in, abhishekd23@iitk.ac.in, namanb23@iitk.ac.in, vikramk23@iitk.ac.in, sovansahoo23@iitk.ac.in,

IIT Kanpur

1. Introduction:

In today's digital age, the internet generates a vast amount of data, offering unique opportunities for exploration and analysis across various domains. Among these, the realm of movies and cinema stands out as a rich source of information ripe for investigation. Leveraging the power of Big Data and advanced analytics techniques, researchers and enthusiasts can delve into movie datasets to uncover patterns, preferences, and trends that were previously inaccessible.

Our project, titled "CineScope: Exploring MovieLens Data Through Visual Analytics", is aimed at harnessing the potential of Big Data and visual analytics to gain comprehensive insights into the world of movies. Using the MovieLens dataset, which contains extensive information on movies, user ratings, genres, tags, and more, we aim to develop a suite of tools and techniques tailored to the exploration of movie-related phenomena.

The objective of our project is to demonstrate the capabilities of visual analytics in uncovering valuable insights from movie data. By building recommendation systems, conducting genre analysis, exploring user preferences, and examining temporal trends, we seek to provide a holistic view of the cinematic landscape. Through interactive visualizations and user-friendly interfaces, we aim to create a platform that enables movie enthusiasts, researchers, and industry professionals to explore and understand the multifaceted world of cinema.

Our endeavor is to contribute to the ongoing dialogue surrounding movie analytics and enhance the overall movie-watching experience for audiences worldwide. By combining cutting-edge technologies with rich movie datasets, we hope to offer new perspectives and insights into the intricate world of movies.

The dataset utilized in this project is "MovieLens 20M" dataset. The "MovieLens 20M" dataset comprises 20,000,263 ratings and 465,564 tag information across 27,278 movies, gathered from 138,493 users between 1995 and 2015. Segmented into six files, including ratings, tags, and movies, each entry adheres to a comma-separated values format. Ratings range from 0.5 to 5 stars, while tags offer user-generated insights. Movie details encompass unique identifiers, titles, and genres. Additionally, the dataset provides links to IMDb and The Movie Database for cross-referencing. Despite anonymized user IDs, the dataset offers a rich resource for exploring movie preferences and user behavior through visual analytics.

2. Tasks:

- **Task1 : Movie Recommendation System**
  - Our objective is to build a movie recommendation system that suggests movies similar to a user's selection based on collaborative filtering with a touch of content-based filtering.
  - A bar chart, allowing you to see how similar each recommended movie is to the user's chosen movie.
  - In a network graph, each node represents a recommended movie, and edges establish connections between movies based on common genres.
- **Task2 : Genre Analysis of Recommended Movies**
  - The task is to create a function that generates a horizontal bar chart to visualize the average ratings of different genres for the recommended list of movies.
  - The function should utilize data from .csv files containing movie information and ratings, calculate genre-specific average ratings, and create a bar chart for each movie in the provided list.
- **Task3 : Distribution of Users for each Movie**
  - The task is to create a sun-burst chart for distribution of user for the selected movie using the movielens dataset.
  - The sun-burst chart should have 3 layers i.e gender, age-group and occupation.
  - The user should have an option to selected the ordering of layers, for example gender will be inner layer, age-group as middle layer and occupation as outer layer.
- **Task4 : Word Cloud for Movie Tags**
  - The task is to create a word cloud of tags that are assigned for a movie. If user selects a movie, then all the tags that are assigned to that movie will be shown as a word cloud.
  - The size of a tag in the word cloud depends on, how much that particular tag is related to the movie. We get this relevance from the relevance score provided in the dataset.
  - By doing this, users will gain insights into the extent to which different tags are associated with the movie. And by looking at the word cloud the user will get an idea of what to expect from the movie.
- **Task5 : User Genre Analysis**
  - The objective is to develop a feature that enables users to analyze their genre preferences based on their movie-watching history.
  - This feature should offer insights into the genres they enjoy the most, ensuring a balanced representation of their preferences

- **Task6 : Genre VS Genre Analysis**
  - Our objective is to create bar charts that compare two genres within the MovieLens dataset.
  - We'll focus on metrics such as movie count, average rating, and viewership to provide a visually clear comparison of their performance.
  - This method will help us gain valuable insights into the popularity and audience engagement levels of each genre.
- **Task7 : Genre wise Movie releases over Time**
  - The Task is to show how the number of movies released is change over time for each genre using combination of line graphs.
  - The Number of movies released over years in each genre is shown by each line graph.
  - There should be an option for the user to select or deselect a particular genre's line graph, so that he can compare b/w multiple genres.
- **Task8 : Movie Genre Distribution by Age Group**
  - The Task is to create a pie chart to show movie genre distribution for different age groups.
  - The User should have an option to select the age group he wants then a pie chart will appear on the screen.
  - This pie chart will show how many movies were watch by the users of the selected age group in each genres.
- **Task9 : Temporal Analysis of Movies and Users**
  - The objective is to create an interactive dashboard for exploring temporal patterns and trends in movie ratings data. Through this dashboard, users can gain insights into how movie ratings vary over time and across different genres.
  - The dashboard features several key functionalities:
    * Firstly, it allows users to visualize the average rating of movies over time, with the ability to select specific genres of interest from a dropdown menu.
    * Secondly, users can examine the number of ratings given to movies over time, providing an understanding of user engagement with movies of different genres.
    * Additionally, the dashboard presents a bar chart displaying the average rating for each genre, enabling users to compare the average ratings across different genres at a glance.
    * Lastly, it illustrates the average rating of movies over the years, categorized by genre, offering a deep understanding of how ratings evolve over time within each genre.
  - This interactive dashboard aims to provide users with a comprehensive and engaging platform for exploring movie rating trends and patterns.

3. **Proposed Solution:** Solutions proposed for each of the above task are

- **Task1 : Movie Recommendation System**
  - Solution was tailored to conduct movie recommendation and network graph analysis using the dependencies pandas, dash, plotly, sklearn, pyvis
  - Functionality:
    - Content of movies (genres and tags) was combined.
    - $vectorizer.fit-transform$: prepared text data for machine learning tasks by converting it to numerical features.
    - CountVectorizer: converted a collection of text documents (like movie tags in code) into a numerical matrix, making it suitable for machine learning algorithms that couldn't directly process text.
    - Cosine similarity was calculated between movies based on these features. This captured how similar movies were based on their tags and genres.
  - Recommendation:
    - Given a movie title (e.g., "Batman"), the system found its index in the data.
    - Cosine similarity scores between this movie and all other movies were retrieved.
    - Movies with the highest similarity scores (most similar tags/genres) were recommended and network graph was also created.
- **Task2 : Genre Analysis of Recommended Movies**
  - Solution was tailored to conduct genre-versus-genre analysis using the dependencies pandas, dash, plotly.
  - Functionality:
    - Data Retrieval and Processing: The code efficiently read two CSV files, 'movies.csv' and 'ratings.csv', which stored movie information and ratings, respectively. The 'movies.csv' file included a 'genres' column with pipe-separated genre lists for each movie. It combined movie genres to form super genres and extracted necessary information such as movie titles, genres, and ratings.
    - Genre Average Ratings Calculation: The code calculated the average ratings for each genre for each movie in the provided list. It iterated through the movies in the list, extracted the relevant genre ratings from the data, and computed the average rating for each genre.
    - Bar Chart Generation: The code generated a horizontal bar chart using Plotly for each movie in the list. It iterated through the genres and their corresponding average ratings for each movie and created a stacked bar chart with each genre represented by a colored bar.
    - Interactive Visualization: Interactivity was implemented by allowing users to select the genres that they want to see in the list of recommended movies. The code dynamically generated the bar chart based on the selected genres as well as displaying the average ratings for each genre.

- **Task3 : Distribution of Users for each Movie**
  - The provided solution utilizes the libraries pandas, dash, and plotly to analyze the distribution of users for each movie. It enables interactive exploration of user demographics such as gender, age group, and occupation across different movies.
  - Functionality:
    * Data Loading and Processing: The code efficiently loads three CSV files: 'movies.csv', 'ratings.csv', and 'users.csv', containing movie information, ratings, and user data, respectively. It extracts essential details such as movie titles, ratings, and user demographics.
    * Age Grouping: A function is implemented to group users into 10-year intervals based on their ages. This ensures a clear segmentation of users by age groups for analysis.
    * Dash App Initialization: The Dash framework is utilized to create an interactive web application for exploring the distribution of users across movies. The layout includes dropdowns for selecting a movie and three layers for analyzing user demographics.
    * Dropdown Interactivity: The dropdowns are dynamically populated based on user selections. For instance, selecting a layer in the first dropdown determines the options available in the subsequent dropdowns, ensuring coherent data exploration.
    * Sunburst Chart Generation: Upon selecting a movie and specifying demographic layers, the code generates a sunburst chart using Plotly. This chart visually represents the distribution of users across different demographic groups for the selected movie.
    * Interactive Visualization: Interactivity is a key feature of the solution, allowing users to dynamically explore the distribution of users for each movie based on various demographic factors. Users can select different movies and demographic layers to visualize the data interactively.
  - The solution provides a user-friendly interface for exploring the distribution of users across movies based on gender, age group, and occupation. Its interactive nature enables users to gain insights into user demographics and their distribution across the movie dataset.
- **Task4 : WordCloud for Movie Tags**
  - The provided solution employs the Dash framework along with pandas, WordCloud, and Plotly to generate wordclouds representing tags associated with each movie. It enables users to visualize the most relevant tags for a selected movie in an interactive manner.
  - Functionality:
    * Data Loading: The code efficiently loads four CSV files: 'movies.csv', 'tags.csv', 'genome-tags.csv', and 'genome-scores.csv', containing movie information, tags, genome tags, and relevance scores, respectively. These datasets provide necessary information for generating movie tag wordclouds.

    * Dash App Initialization: A Dash application is initialized to create an interactive web interface for exploring movie tag wordclouds. The layout includes a dropdown menu for selecting a movie.
    * Dropdown Interactivity:The dropdown menu dynamically populates movie options based on the available movies in the dataset. Users can select a movie of interest to visualize its associated tags.
    * Wordcloud Generation: Upon selecting a movie, the code filters tags associated with that movie and retrieves their relevance scores from the 'genome-scores.csv' file. It then generates a wordcloud using the WordCloud library, where the size of each tag is determined by its relevance score.
    * Plotly Integration: The wordcloud generated by WordCloud is converted into a Plotly figure for seamless integration with the Dash app. This enables interactive visualization and further customization.
    * Interactive Visualization: Users can interactively explore the wordcloud representing tags for a selected movie. The wordcloud dynamically adjusts based on the chosen movie, providing insights into the most relevant tags associated with it.
  - The solution offers a user-friendly interface for visualizing tags associated with each movie through wordclouds. By interactively selecting movies, users can gain insights into the most relevant tags and their relevance scores, facilitating better understanding of movie content and themes.

- **Task5 : User Genre Analysis**
  - The solution was designed to perform user genre analysis using the following dependencies pandas, dash and plotly.
  - Functionality:
    * Data Reading and Preprocessing: Read CSV files ('ratings.csv', 'tags.csv', 'movies.csv', 'links.csv') into Pandas DataFrames. Processed the data to compute metrics such as average rating and Bayesian average rating for movies.
    * Genre Handling: Merged similar genres into broader categories based on predefined mappings. Updated the dataset with the combined genres for each movie. Conducted genre-specific analysis by counting movies with specific genres that the user had rated previously. Generated a Sunburst chart to visualize the distribution of genres and sub-genres in the dataset.
    * Interactive Visualization: Developed an interactive Dash web application allowing users to compare their preferred genres. Implemented a dropdown menu for users to input their user id. Dynamically updated the Sunburst chart based on user selections to display the count of movies for the chosen genre.

- **Task6 : Genre VS Genre Analysis**
  - Solution was tailored to conduct genre-versus-genre analysis using the dependencies pandas, dash and plotly.

- Functionality:
  - Data Handling: The code efficiently read two CSV files, 'movies.csv' and 'ratings.csv', which stored movie information and ratings, respectively. The 'movies.csv' file included a 'genres' column with pipe-separated genre lists for each movie.
  - Metrics Calculation: It calculated essential metrics for each genre, including the count of movies, average rating, and total ratings.
  - Interactive Dashboard Creation: Using Dash, it constructed a user-friendly web application featuring three dropdown menus. These dropdowns enabled users to select two genres for comparison.
  - Dynamic Visualization: Based on the user's selections, the application dynamically updated three bar charts. These charts visually represented the count of movies, average rating, and total ratings for the chosen genres.

- **Task7 : Genre wise Movie releases over Time**
  - The approach utilizes the Dash framework and Plotly to craft an interactive visualization demonstrating movie releases across genres over time. Users can scrutinize trends in movie releases across various genres spanning from 1995 to 2012.
  - Functionality:
    - Data Preparation: Initially, the code extracts movie data from 'movies.csv' and categorizes movies into distinct genres. To enhance visualization clarity, genres are mapped to broader categories, and a dictionary structure is established to tally movie releases for each genre within the specified timeframe.
    - Dash App Initialization: An interactive web interface is instantiated using Dash to visualize genre-wise movie releases over time. The layout encompasses a line plot designed to showcase trends in movie releases for user-selected genres.
    - Plot Generation: Upon genre selection, the code dynamically generates line plots depicting movie release counts across the years for each chosen genre. Each genre is delineated by a distinct line on the plot, enabling users to juxtapose trends across genres effectively.
    - Interactive Visualization: The plot offers interactivity, allowing users to click on individual genre lines to scrutinize specific genres or compare multiple genres concurrently. This feature empowers users to pinpoint trends and gain a comprehensive understanding of each genre's movie release pattern over time.
  - The solution offers an intuitive interface for exploring genre-wise movie releases throughout the years. Users can get insights and analyze trends in movie releases, thereby attaining a deeper understanding of how movie genres have evolved over time.

- **Task8 : Movie Genre Distribution by Age Group**
  - The solution employs Pandas and Plotly Express to analyze the genre distribution among different age groups of users. By merging user and movie

datasets, the code generates pie charts illustrating the distribution of movie genres within specified age ranges.

- Functionality:
  - Data Preparation:The code reads user and movie data from CSV files and maps movie genres to broader categories according to predefined mappings. This preprocessing step ensures uniformity and clarity in genre representation. Additionally, age range options are defined for user selection.
  - Pie Chart Generation: Upon selecting age range parameters, the code filters user data accordingly and merges it with movie data to obtain genre information. It then calculates the frequency of each genre and generates a pie chart illustrating the distribution of genres within the specified age group.
  - Interactive Visualization: The resulting pie chart provides an interactive visualization of genre distribution for the selected age range. Users can dynamically explore the prevalence of different genres among users of varying ages, gaining insights into genre preferences across different age groups.
- The solution offers an intuitive interface for exploring genre distribution among different age groups of users. By interactively selecting age ranges, users can visualize the prevalence of various movie genres, facilitating analysis and understanding of genre preferences across different demographics.

- **Task9 : Temporal Analysis of Movies and Users**
  - Solution tailored to conduct temporal analysis using the dependencies pandas, dash and plotly.
  - Functionality:
    - Data handling: It involves loading movie data and ratings from CSV files using Pandas. After merging the datasets based on the movie ID, timestamps are converted to datetime format. Data is then processed to extract years from timestamps for temporal analysis. Finally, the processed data is used to generate various visualizations to explore movie rating trends.
    - Metrics Calculations: Metrics are calculated by grouping data by genre and year, then aggregating ratings. Average ratings per genre are computed by averaging ratings within each genre. Number of ratings per genre and year are calculated by counting the occurrences of ratings. These metrics are used to create visualizations illustrating rating trends over time and across genres.
    - Dynamic visualizations: We achieved it using Plotly and Dash libraries in Python. Dash components enable interactive elements like dropdowns to filter data. Callback functions update the visualizations based on user-selected options. Users can dynamically explore movie rating trends by selecting genres from dropdown menus.

4: Results

- **Task1 : Movie Recommendation System**

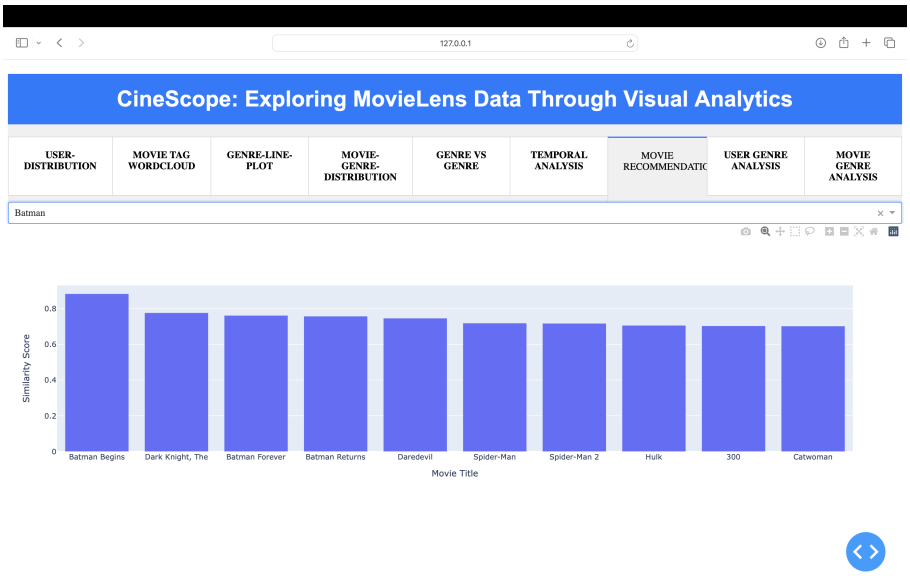

Figure 1: Recommended Movies for Batman

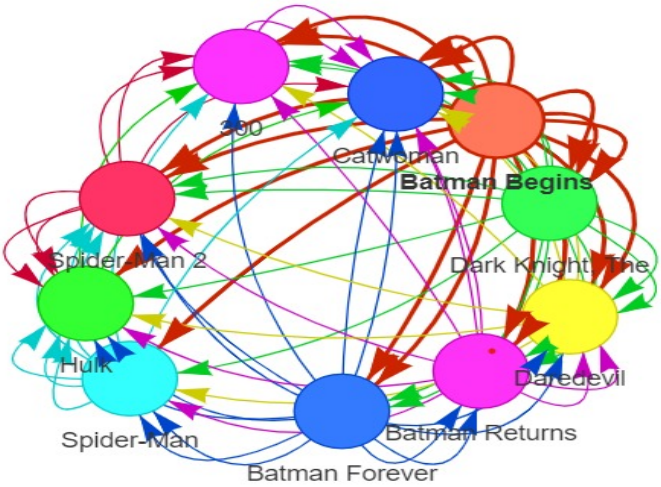

Figure 2: Network Graph of Recommended Movies for Batman

- **Task2 : Genre Analysis of Recommended Movies**

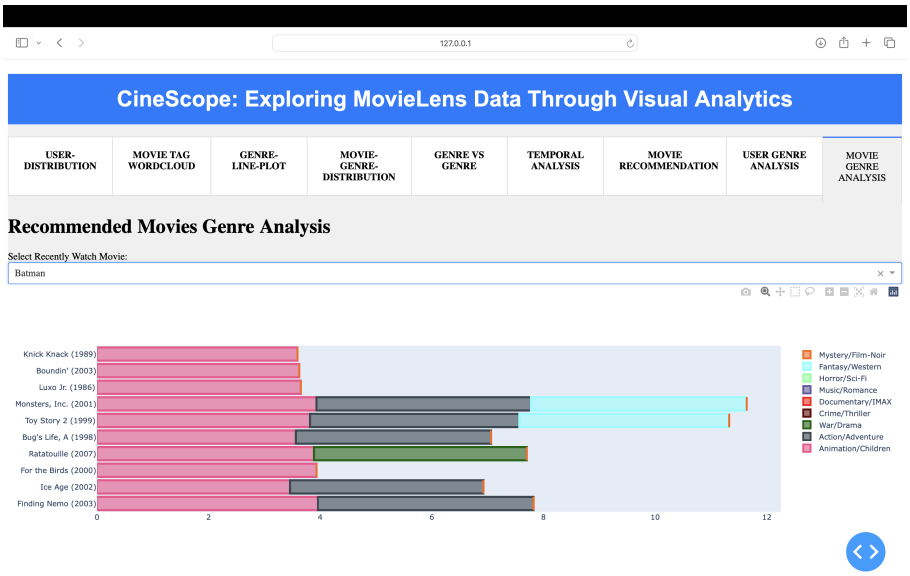

Figure 1: Genre analysis of recommended movies for Batman

- **Task3 : Distribution of Users for each Movie**

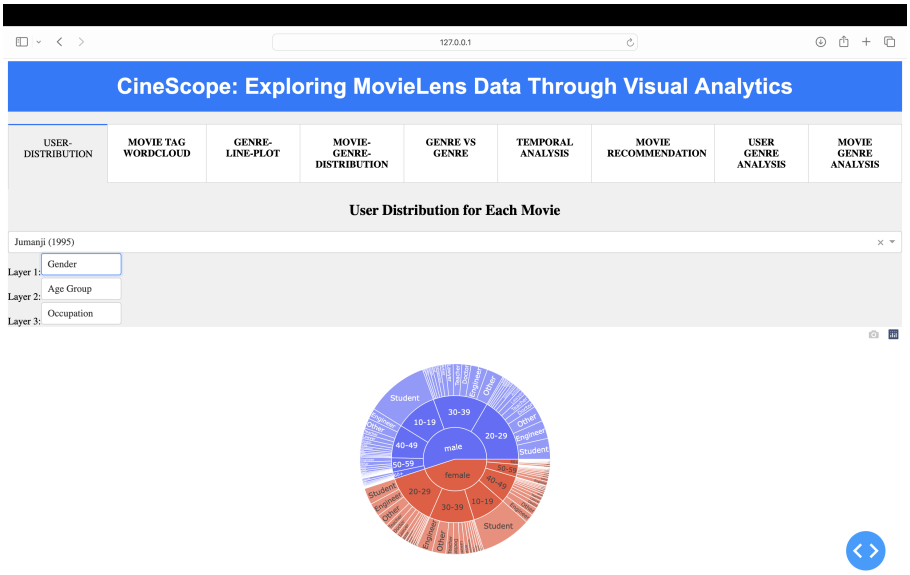

Figure 1: Distribution of Users for Jumanji Movie

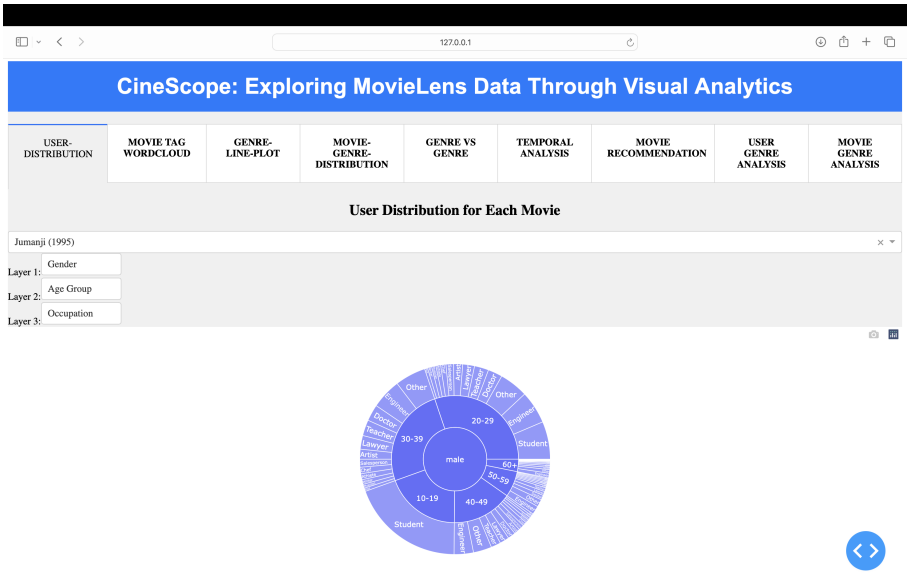

Figure 2: Results when user clicks on male in above figure

- **Task4 : Word Cloud for Movie Tags**

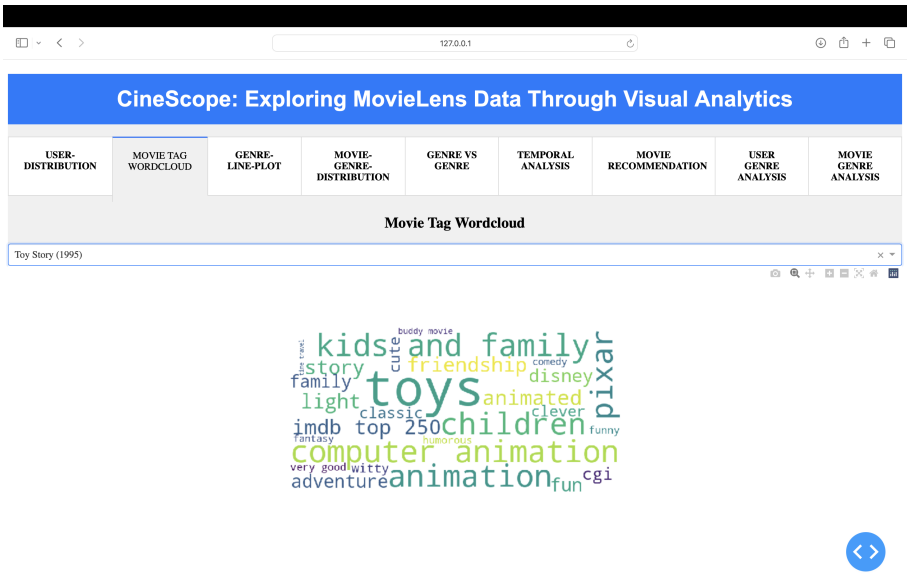

Figure 1: Word Cloud for Toy Story

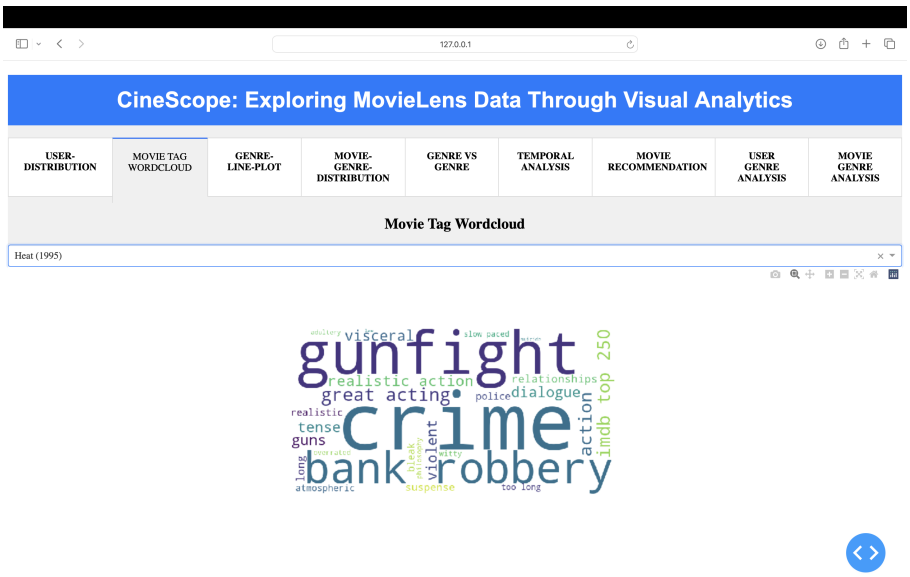

Figure 2: Word Cloud for Heat

- **Task5 : User Genre Analysis**

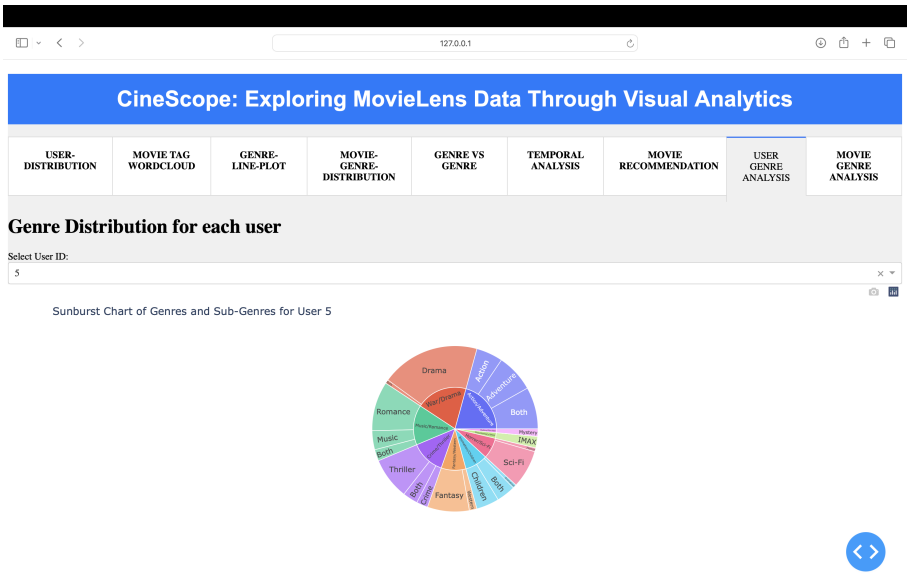

Figure 1: User Genre analysis for user 5

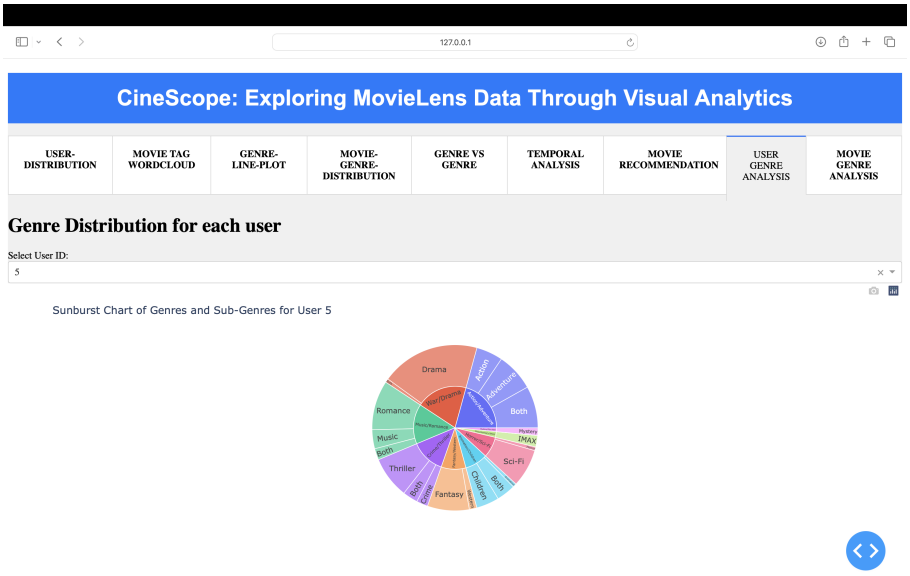

Figure 2: Result when user clicks on Action/Adventure in above figure

- **Task6 : Genre Vs Genre Analysis**

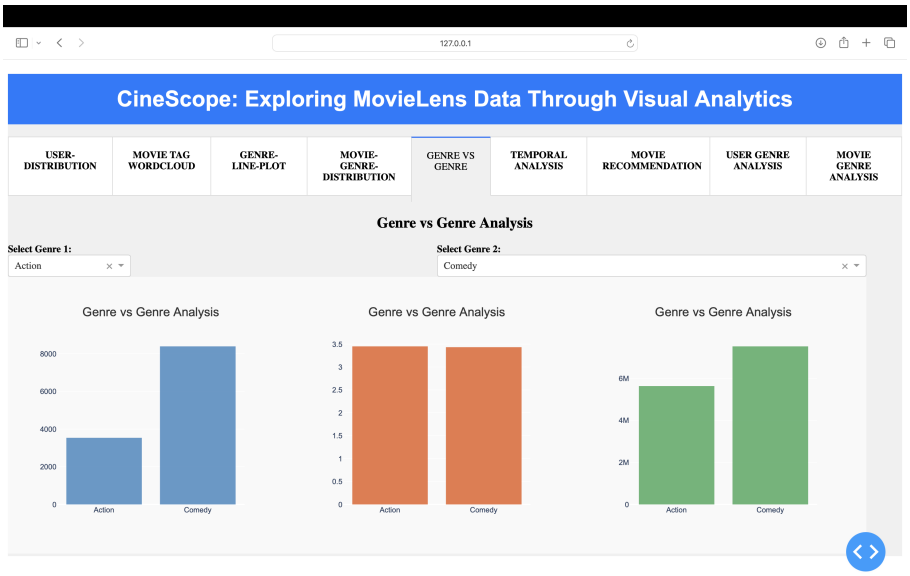

Figure 1: Action VS Comdey Analysis

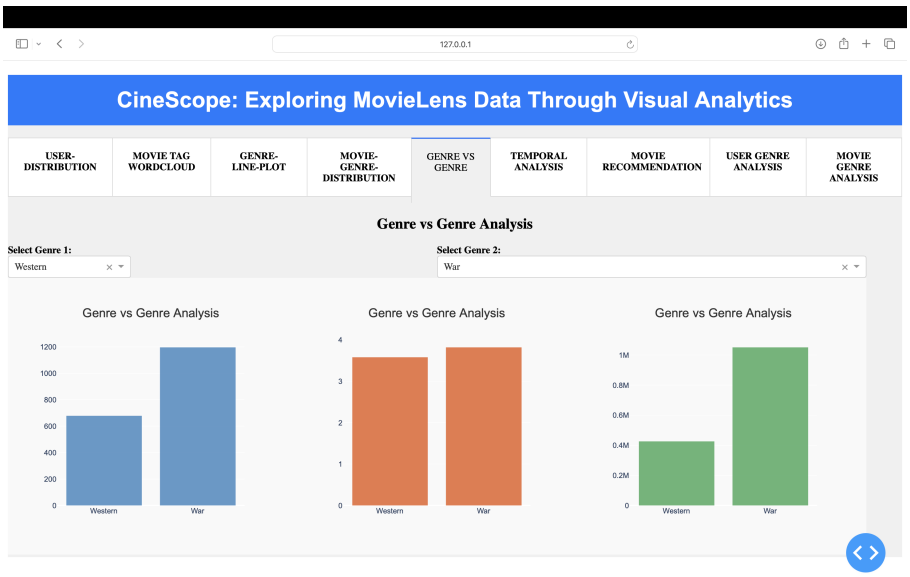

Figure 2: Western VS War Analysis

- **Task7 : Genre Wise Movie releases over Time**

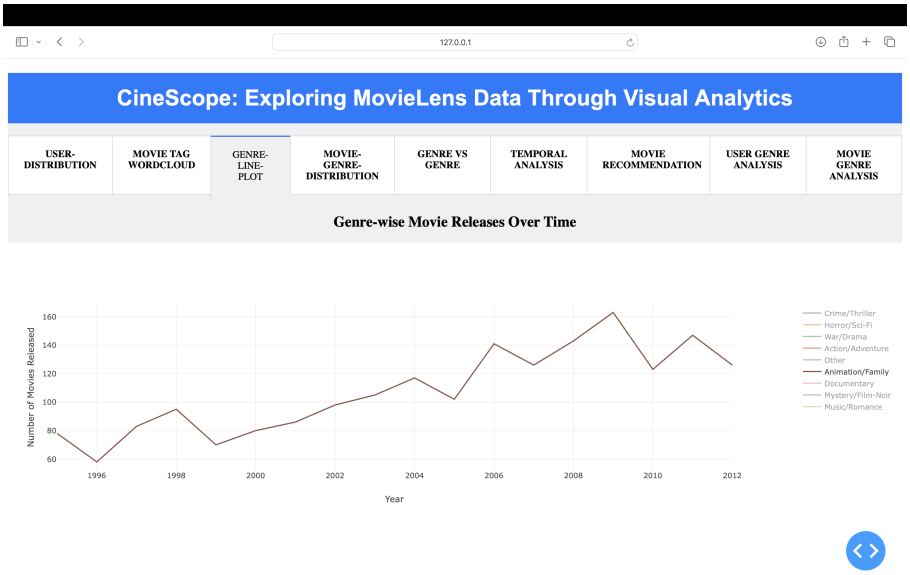

Figure 1: Release of Animation/Family Movies over time

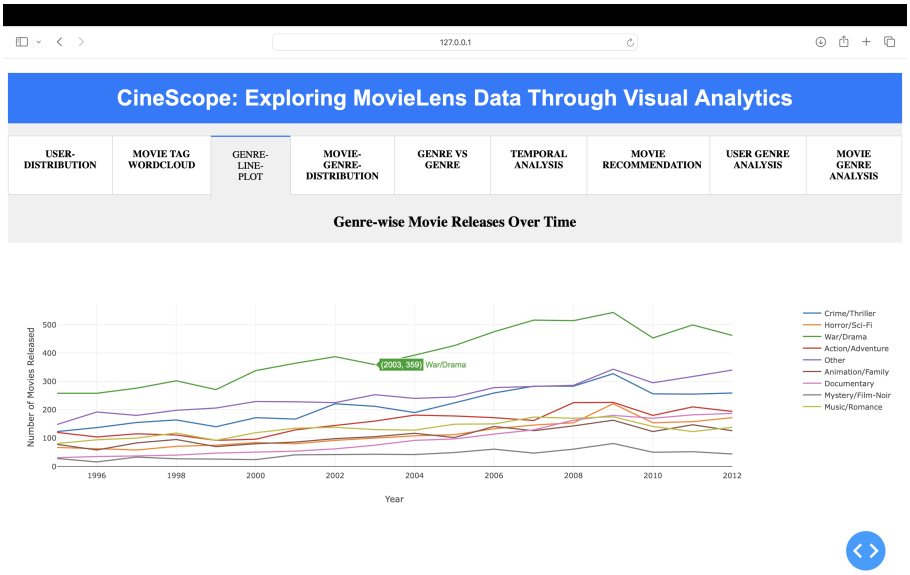

Figure 2: Release of Movies in all genre over time

- **Task8 : Movie Genre Distribution by Age Group**

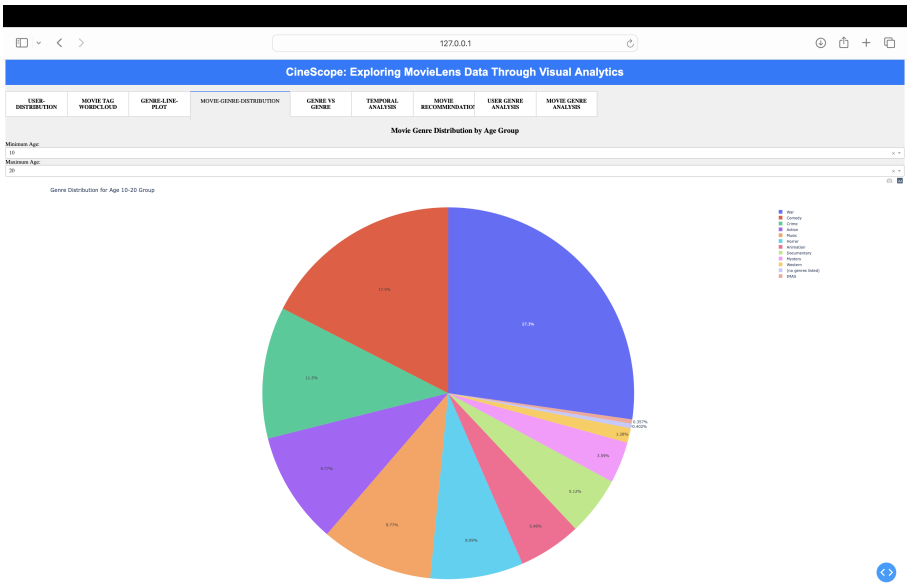

Figure 1: Movie Genre Distribution of Age Group 10-20

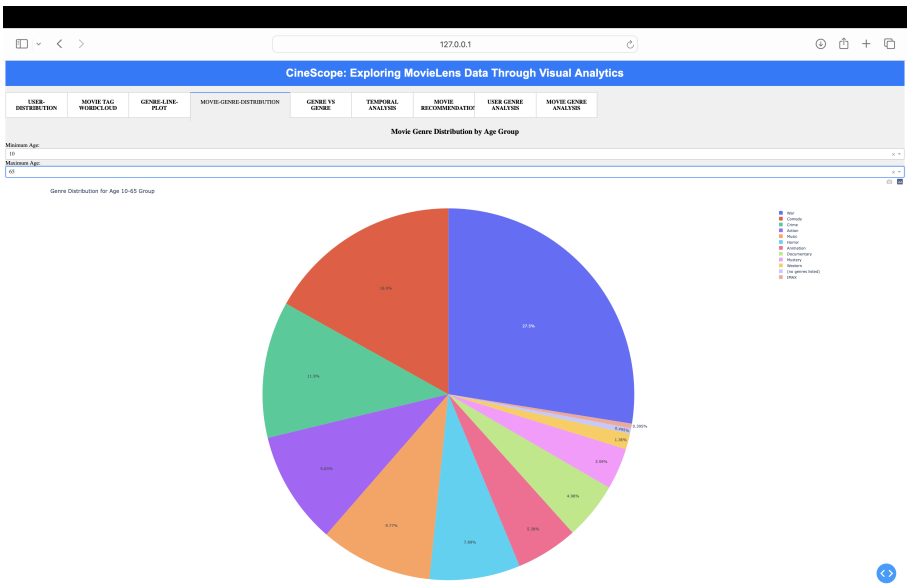

Figure 2: Movie Genre Distribution for Age group 10-65

- **Task9 : Temporal Analysis of Movies and Users**

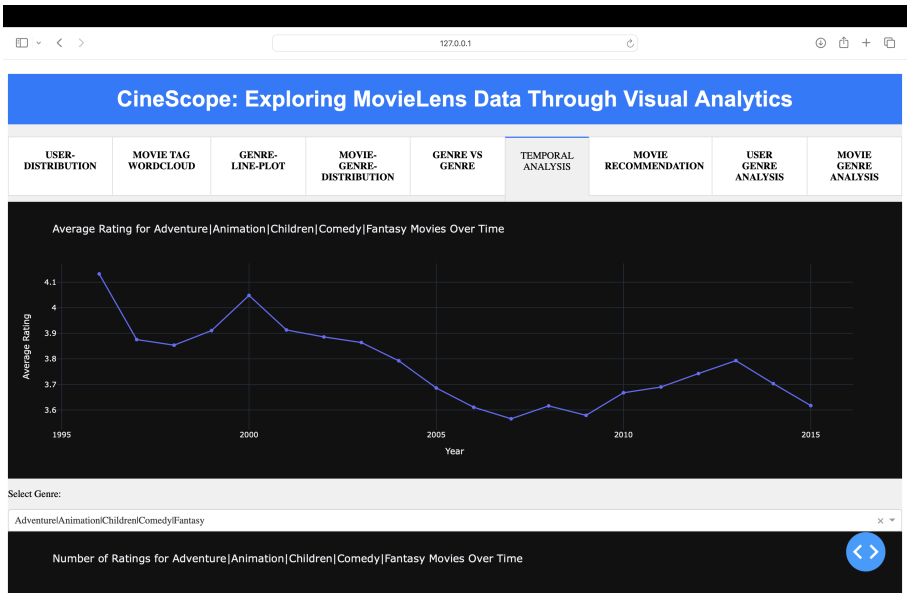

Figure 1: Temporal Analysis of Movies and Users

5. Conclusion: In conclusion, our project, "CineScope: Exploring MovieLens Data Through Visual Analytics," has provided valuable insights into the world of movies through the lens of Big Data and visual analytics. By leveraging the rich MovieLens dataset and employing advanced analytics techniques, we have successfully developed a suite of functionalities tailored to exploring various characteristics of movie-related data.

Throughout our project, we have demonstrated the power of visual analytics in uncovering valuable insights from movie data. From building recommendation systems to conducting genre analysis, exploring user preferences, and examining temporal trends, our project has offered a comprehensive view of the cinematic landscape. Through interactive visualizations and user-friendly interfaces, we have created a platform that enables movie enthusiasts, researchers, and industry professionals to explore and understand the multifaceted world of cinema in an engaging and insightful manner.

Moving forward, there are several avenues for further exploration and enhancement of our project. This includes refining recommendation algorithms, incorporating additional datasets for a more comprehensive analysis, and expanding the scope of visualizations to encompass a broader range of movie-related metrics and trends. In summary, our project has not only showcased the capabilities of visual analytics in the domain of movies but has also contributed to the ongoing dialogue surrounding movie analytics.

6. Link to source code:
https://github.com/JeswaanthGogula/CS661_PROJECT

References

https://plotly.com/python/
https://grouplens.org/datasets/movielens/20m/
https://pandas.pydata.org/docs/