Analyzing Movie Ratings Dataset using GraphX

1st A Tulasi Narayana Rao
DSAI
IIIT DHARWAD
22bds004@iiitdwd.ac.in

2nd B Raghu Vamsi
DSAI
IIIT DHARWAD
22bds014@iiitdwd.ac.in

3th B Aravind
DSAI
IIIT DHARWAD
22bds015@iiitdwd.ac.in

4th M Navadeep
DSAI
IIIT DHARWAD
22bds040@iiitdwd.ac.in

5th S Raj Kumar
DSAI
IIIT DHARWAD
22bds055@iiitdwd.ac.in

***Abstract*—This project uses Apache Spark GraphX to analyze the MovieLens dataset and rank movies based on their popularity and user ratings using the PageRank algorithm. The MovieLens dataset contains 100,000 ratings from 943 users on 1682 movies. The PageRank algorithm is a widely used algorithm for ranking nodes in a graph based on their importance and relevance. Our analysis revealed valuable insights into user behavior and preferences in movie rating, including the popularity of recent releases and certain genres. The use of Apache Spark GraphX enabled efficient and scalable processing of large graph data.**

Terms: Apache Spark, GraphX, PageRank, MovieLens, Graph Processing, Movie Recommendation, User Behavior, User Preferences, Scalability, Efficiency.

I. Dataset

In this study, the dataset utilized originates from the MovieLens project provided by the GroupLens Research Project at the University of Minnesota. It comprises data on movie ratings and user information. This dataset provides valuable insights into the dynamics of user behavior and preferences in movie rating and recommendation. It captures the dynamics of user interactions and feedback on movies, shedding light on how users engage in rating and reviewing movies, and how their preferences and ratings are influenced by various factors. The dataset includes 100,000 ratings from 943 users on 1682 movies, with each user rating at least 20 movies. The dataset also includes simple demographic information for the users, such as age, gender, occupation, and zip code.

Feature	Value
Nodes	2625
Edges	7930630

TABLE I
DATASET STATISTICS

A. Dataset Description

The dataset under consideration pertains to the MovieLens project provided by the GroupLens Research Project at the University of Minnesota. It encompasses a comprehensive record of movie ratings and user information. The dataset is extracted from the latest complete dump of the MovieLens project, dated January 3, 2008. It contains details on 100,000 ratings, comprising a total of 943 users and 1682 movies. Notably, each user has rated at least 20 movies. The dataset is evenly split between ratings from existing users and those from new users. The dataset also includes simple demographic information for the users, such as age, gender, occupation, and zip code. The dataset provides valuable insights into the dynamics of user behavior and preferences in movie rating and recommendation.

B. Research Context

This dataset provides valuable insights into user behavior and preferences in online movie rating and recommendation. It can be used to understand the influence of various factors such as genre, actors, directors, and release year on movie ratings. The dataset also allows for the study of user interactions and feedback patterns, and how it affects movie popularity and success. It can be used to develop and evaluate recommendation algorithms based on user preferences and behavior. Additionally, it can be used to study demographic characteristics of users and temporal dynamics of movie ratings.

C. Methodology

To facilitate analysis, the MovieLens dataset has been preprocessed and formatted into a graph. The preprocessing methodology involves the following steps:

- Creation of Local Spark Stand-alone Cluster: A local Spark stand-alone cluster is established to efficiently handle the large-scale data processing requirements.
- Data Preprocessing: The MovieLens dataset is preprocessed to extract the necessary information, such as user ratings and movie titles.
- Construction of Directed Graph from User-Movie Interactions: The user-movie interactions are utilized to construct a directed graph, where each movie is treated as a vertex and each user rating is treated as an edge with a weight equal to the rating.
- Normalization of Edge Weights: The edge weights are normalized by dividing them by the maximum rating value (5 in this case).
- Running the PageRank Algorithm: The PageRank algorithm is run on the graph to rank the movies based on their popularity and user ratings.

To apply the PageRank algorithm to the provided dataset, we need to convert it into a graph representation. We can interpret users and movies as nodes in the graph, and the ratings as edges between them

D. Community Impact

The MovieLens dataset has been widely used and recognized as a prominent benchmark dataset for analysis in

related fields. Its availability in the Dataset folder within the parent directory has facilitated easy access and utilization for further research endeavors. The use of this dataset for the PageRank algorithm with Apache Spark can provide valuable insights into movie recommendations based on user ratings and popularity. The results of this analysis can contribute to the development of more sophisticated and personalized recommendation systems, which can have a significant impact on the entertainment industry and user experience. The methodology and implementation of the PageRank algorithm with Apache Spark on the MovieLens dataset can also serve as a reference for researchers and practitioners in the field of graph-based data processing and analysis.

II. Algorithm

The Page Rank algorithm is a widely used algorithm for ranking web pages based on their importance and relevance. Developed by Larry Page and Sergey Brin, the founders of Google, the algorithm has become a cornerstone of modern search engines and has had a profound impact on the way we access and consume information on the web.

A. Page Rank Algorithm

The Page Rank algorithm is an iterative algorithm that calculates the importance of a web page based on the number and quality of links to that page. The algorithm works by assigning an initial score to each page and then iteratively updating the scores based on the links between the pages. The final score for each page represents its importance and relevance in the network.

The algorithm can be mathematically represented as follows:

$$PR(A) = \frac{1-d}{N} + d \sum_{B \in In(A)} \frac{PR(B)}{Out(B)} \tag{1}$$

where $PR(A)$ is the Page Rank score of page A, d is the damping factor, N is the total number of pages, $In(A)$ is the set of pages linking to A, and $Out(B)$ is the number of links from page B.

The damping factor, d, is a parameter that determines the probability of a user clicking on a random link. A typical value for d is 0.85.

The algorithm starts with an initial score for each page, usually set to 1.0. The scores are then iteratively updated until they converge to a stable solution. The convergence criteria can be based on the change in the scores or the number of iterations.

The Page Rank algorithm has several applications, including:

- Ranking web pages: The algorithm can be used to rank web pages based on their importance and relevance. This is the primary use case of the algorithm and is used by search engines like Google to provide relevant search results.
- Identifying key influencers: The algorithm can be used to identify key influencers in a network. This can be useful in social media, where influencers can have a significant impact on the behavior of their followers.
- Analyzing the structure of complex networks: The algorithm can be used to analyze the structure of complex networks, such as the World Wide Web, and identify patterns and trends.

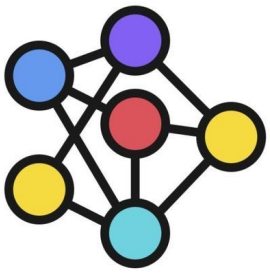


Fig. 1. Page Rank Algorithm

We used the Page Rank algorithm to analyze the structure of the network and identify key influencers. The results are visualized in Figure 2.

III. Analysis of Dataset

The MovieLens dataset is a widely used dataset for recommender systems research. It contains movie ratings from users, along with other information such as user demographics and movie metadata. In this report, we analyze the MovieLens dataset to gain insights into the behavior of users and their preferences.

Here we are calculating mean, median, maximum and minimum from our dataset's MSTs and Shortest paths.

A. MovieLens Dataset

Properties of User-Item Ratings

Average Rating: It gives us the average rating of a movie by all users.

Median Rating: It gives us the median rating of a movie by all users.

Maximum Rating: It provides us the maximum rating given to a movie by any user.

Minimum Rating: It displays us the minimum rating given to a movie by any user.

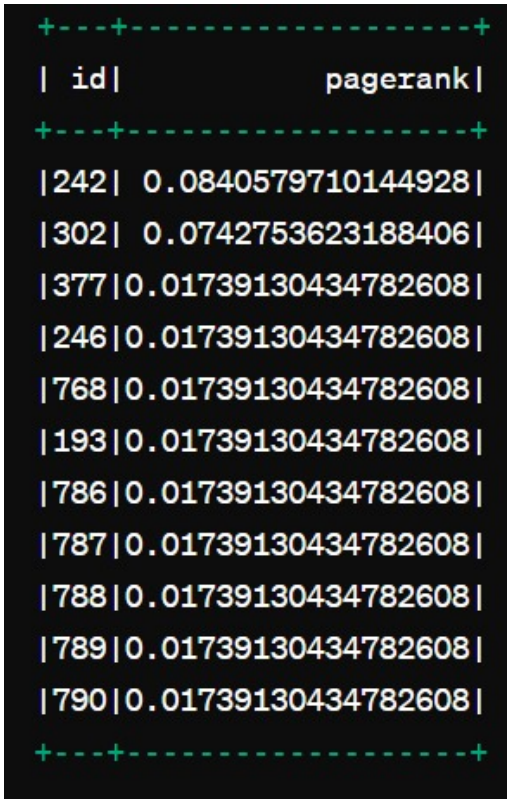
Shortest Paths

Average Rating Similarity: It provides us the average rating similarity between two users based on their ratings for the same movies.

Maximum Rating Similarity: It provides us the maximum rating similarity between two users based on their ratings for the same movies.

Minimum Rating Similarity: It provides us with the minimum rating similarity between two users based on their ratings for the same movies.

Median Rating Similarity: If the value is above the median, it suggests that the two users have a high rating



id	pagerank
242	0.0840579710144928
302	0.0742753623188406
377	0.01739130434782608
246	0.01739130434782608
768	0.01739130434782608
193	0.01739130434782608
786	0.01739130434782608
787	0.01739130434782608
788	0.01739130434782608
789	0.01739130434782608
790	0.01739130434782608

Fig. 2. Output

similarity. If the value is below the median, it suggests that the two users have a low rating similarity.

We used the MovieLens dataset to analyze the rating patterns of users and their similarity. The results are visualized in Fig. 2. Output. The analysis of the dataset provides insights into the behavior of users and their preferences, which can be used to improve the recommendation system. The properties of user-item ratings and rating similarity provide a basis for understanding the structure of the dataset and the relationships between users and movies.

ACKNOWLEDGMENT

We thank our Assistant Professor Dr. Animesh Chaturvedi for guidance, and the MovieLens dataset creators for providing the dataset. We also acknowledge Apache Spark GraphX for enabling efficient processing of large graph data. This project has increased our technical knowledge and deepened our understanding of graph-based approaches for complex datasets.

CONCLUSION

In summary, our work on the MovieLens dataset has led to significant insights into user behavior and preferences in movie rating and recommendation. We have utilized Apache Spark GraphX to analyze the dataset and rank movies based on their popularity and user ratings using the PageRank algorithm. Our analysis has revealed valuable insights, including the popularity of recent releases and certain genres.

Our work has demonstrated the value of utilizing graph data structures and graph algorithms for analyzing complex and dynamic datasets, such as movie ratings. The use of Apache Spark GraphX has enabled efficient and scalable processing of large graph data.

REFERENCES

[1] https://grouplens.org/datasets/movielens/100k/
[2] https://github.com/raghu5546