Course Info: Big Data Analysis
Date: June 5, 2023
Name: Rabiya Fatima

Use Case Name: Movie Recommendations Analysis

Introduction:
This paper presents an analysis of movie recommendations using a collaborative filtering approach. The goal of this use case is to develop a movie recommender system that provides personalized movie suggestions to users based on their preferences and ratings. By leveraging the power of collaborative filtering and user feedback, we aim to enhance the movie-watching experience for users.

Dataset Used:
The dataset used for this analysis is the MovieLens dataset. The dataset can be obtained from the MovieLens website at https://grouplens.org/datasets/movielens/. It provides a collection of movie ratings and user information, allowing us to analyze user preferences and generate accurate movie recommendations. The dataset includes the following fields:
- User ID
- Movie ID
- Rating
- Timestamp
- Additional movie metadata (e.g., title, genre)

Technical Details:
This analysis leverages Apache Spark for processing and analysis tasks. Spark's collaborative filtering algorithm is applied to generate movie recommendations based on user ratings and preferences. The implementation utilizes Spark's MLlib library, which provides efficient and scalable machine learning functionalities.

Debugging Details:

During the implementation of the movie recommendations analysis, several challenges were encountered and overcome. One significant challenge was related to the large dataset size, which resulted in low disk space errors during model training. To address this issue, a modification was made to the AWS instance used for running the project.
Initially, the project was executed on an AWS t2.medium instance, which had limited disk space. However, due to the size of the dataset and the computational requirements of the algorithms, additional disk space was necessary. To overcome this challenge, the AWS instance was updated to a t2.large instance, which provided more disk space and improved performance.
By upgrading the AWS instance, the project was able to run smoothly and efficiently, allowing for the successful execution of the movie recommendations analysis.

An achievement in this project was the successful implementation of the collaborative filtering algorithm and the generation of personalized movie recommendations. The system was able to accurately identify movies that align with users' tastes and preferences, enhancing the movie-watching experience.

Results:

The Spark-based movie recommender system was evaluated using various test sets and scenarios. The system successfully generated personalized movie recommendations for users based on their ratings and preferences.

The movie recommendations analysis yielded insightful results based on various test sets and scenarios. Here is a brief summary of the key findings:

1. Test Set 1: New User Recommendations with Limited Ratings
   - In this scenario, the recommender system successfully provided movie recommendations for a new user with limited ratings.
   - By leveraging collaborative filtering techniques, the system was able to predict the user's preferences and suggest movies tailored to their taste.
   - The recommendations were generated based on the similarity between the user's ratings and those of other users in the dataset.

2. Test Set 2: New User Recommendations with Diverse Ratings
   - This scenario involved a new user (user ID 2) with a wide range of ratings, indicating a diverse set of preferences.
   - The recommender system effectively analyzed the user's ratings and identified movies that aligned with their varied interests.
   - By considering the ratings of similar users, the system provided recommendations that encompassed different genres and styles.

3. Test Set 3: New User Recommendations with More Than 25 Reviews
   - In this scenario, the recommender system handled Users 1 and 2 with a substantial number of reviews.
   - Despite the increased complexity due to the larger amount of user data, the system successfully generated accurate recommendations.
   - The recommendations were based on the user's specific preferences, as well as the ratings and patterns observed in the broader dataset.

4. Test Set 4: Top Recommended Movies (with more than 100 reviews)
   - This scenario focused on identifying the top recommended movies based on popularity and positive user reviews.
   - The recommender system selected movies that had received a significant number of reviews and maintained a high average rating.
   - These recommendations provided users with a curated list of highly regarded movies with broad appeal.

Overall, the movie recommendations analysis demonstrated the effectiveness of the collaborative filtering approach in generating personalized movie recommendations. The system successfully handled different test sets and scenarios, accommodating users with limited or diverse ratings and providing accurate recommendations based on their preferences.

Insight:
The analysis of movie recommendations using collaborative filtering provided valuable insights and actionable items. By analyzing user preferences and ratings, we discovered that users tend to prefer movies with higher average ratings and higher review counts. This insight can be utilized by movie platforms to highlight popular and highly-rated movies to attract user attention.

Furthermore, the collaborative filtering approach allows for the identification of movies that users might have missed but align with their tastes. This can enhance user satisfaction and engagement with the movie platform, leading to increased user retention and loyalty.

The business implications of this analysis are significant. Movie platforms can leverage collaborative filtering algorithms to provide personalized recommendations, increasing user engagement and satisfaction. Additionally, the insights gained from user preferences can inform marketing and content strategies, enabling platforms to curate and promote movies that resonate with their user base.

References:

1. GroupLens. "MovieLens Dataset." MovieLens, GroupLens Research, 2021, https://grouplens.org/datasets/movielens/.

2. "Programming with RDDs in Apache Spark | RDD in Spark | Intellipaat." Intellipaat, 10 Feb. 2021, intellipaat.com/blog/tutorial/spark-tutorial/programming-with-rdds/?US.

3. "Using Collaborative Filtering in Recommender Systems." Recommender Systems Handbook, edited by Francesco Ricci et al., Springer, 2011, pp. 39-71, https://link.springer.com/chapter/10.1007/978-1-4899-7637-6_3

Appendix:

Dataset Fields and Descriptions
The movie recommendations analysis utilized a dataset consisting of movies and user ratings. Here are the fields included in the dataset along with their descriptions:

1. Movie ID: A unique identifier for each movie.
2. Title: The title of the movie.
3. Genres: The genres or categories associated with the movie.
4. User ID: A unique identifier for each user.
5. Rating: The rating given by the user for a particular movie.
6. Timestamp: The timestamp indicating when the user rated the movie.