Movie Recommender System Based on Collaborative Filtering Using Apache Spark

Pranav Chalasani
University of North Texas
11712496

Aryanth Kondreddy
University of North Texas
11696947

Ganesh Gundekarla
University of North Texas
11700551

Puneet Puttu
University of North Texas
11691685

Aishwarya Karukonda
University of North Texas
11644805

Abstract*—In this modern digital era, the wide availability of content present within movie streaming platforms across the world will pose significant challenges for users who are seeking to uncover relevant and enjoyable movies according to the user's preferences amidst the vast spectrum of options available. Here, our project on a personalized movie recommendation system will emerge as an effective tool that will be able to guide the users through the vast content available across these services, by offering them effective tailored suggestions based on a much-sophisticated algorithm and backed by previous user data which is analyzed. Our project will primarily focus on the development and implementation of a movie recommendation system according to user preferences through leveraging machine learning algorithms and data analytics tools. This is done by studying the user's behavior and preferences which is unique for everyone and his/her past interactions with the platform. Our developed system will aim to generate individualized recommendations that will closely align with the respective user's taste and interests. This project aims to contribute to the growing research on personalized recommendation systems which will provide tailored suggestions to each individual automatically elevating user experience, and increasing overall platform competitiveness.

***Index Terms*—Big Data, PySpark, Apache, ML, Movie Recommendation System , Collaborative filtering , Streaming platforms , Recommendation accuracy , Algorithmic bias , Data analytics , User engagement , User statisfaction.**

I. Introduction

In today's digital age, the landscape of entertainment consumption, particularly in the realm of movies, has undergone a profound transformation. The streaming services now have content libraries that are massive to the point of being filled to the brim. It becomes a paradox of choice for users when they want to discover new and relevant movies, as they have to sieve through numerous options. It is an age of plenty, but it makes the users walk in a minefield trying to get away by finding gems through the navigation system in an ocean of content that best matches his or her unique preferences and tastes. In such a desperate need for directions in a sea of choices, it is not a surprise that personal movie recommendation systems have appeared as a friend who really helps the users get the best of their time and find the best way in the discovery of content. The latter taps into sophisticated algorithms and user data, which help give personalized recommendations, not in a bid to meet individual preferences but in a manner that helps deepen a relationship between the user and the wide world of cinematic offerings at his or her disposal. In an expandingly complex digital entertainment space, personalized movie recommendation systems have been identified as technology beacons to deliver end-users more directly to desired, interesting content across the vast expanse of choice in movies. Such recommendation systems make use of machine learning algorithms and data analytics to study the user behavior, preferences, past interaction with the films in order to be able to generate individual viewing habits. With that, the personalized recommendations system should be able to actually sift through the whole body of content in order to select a set of movies that would actually be tailor-made to be of interest to the user with some specific taste and set of interests. The systems facilitate the process of content discovery and contribute not only to its easing but also to increasing an enhanced feeling of connection and engagement between users and their movies, enhancing so their overall experience in viewing films and engendering greater satisfaction and loyalty of the users so as to survive the raised competition in the digital entertainment space.

II. Existing Solutions on Personalized Movie Recommendations

Personalized movie recommendation systems provide a pivotal role in the information overload as faced by users within the present day in the media environment. Traditional ways of film discovery—like the routine scanning of genre categories or generic recommendations - often fail to make tailored suggestions for individual preferences (Deldjoo et al., 2015). This may end up making users feel either overwhelmed or even unsatisfied with the overall experience - hence not interested in engaging and not retained. Personalized movie recommendation systems can draw through strong algorithms and massive user data on individual preference, viewing habits, and demographic data for recommending the best-matched movie for a particular user. These recommendations would assist not only the users in the discovery of new and relevant content but also increase their satisfaction, engagement, and loyalty. Personalized recommendations could also help create a very much differentiated and individual experience that would make such streaming platforms far more competitive.

III. Overview of the Problem Statement and Objectives of the Project

Whereas personalized movie recommendation bears many advantages, certain challenges stand in the development and deployment of effective algorithms. For example, the scalability and efficiency of the recommendation system, more so when working with big datasets involving millions of user interactions and movie ratings. A traditional recommender system would find it hard even to process and analyze this huge amount of data on time, thereby resulting in suboptimal performance in terms of recommendation quality and user experience (Beel et al., 2016). Based on these challenges, the main objective of this project is to come up with a developed movie recommendation system based on collaborative filtering techniques, which is implemented on the Hadoop framework. One of the most popular ways of implementing recommendation systems is collaborative filtering. In doing this, recommendations are personalized, and they are derived from user-item interaction data. We would increase the recommendation-system effectiveness without any bounds on the scalability by using the power of Hadoop, which is a computing framework targeted for large-scale data processing, capable of doing the parallel process.

IV. Collaborative Filtering and Its Relevance to the Project

Collaborative Filtering is a recommendation technique that is based on the combined judgment of the users or preference when selecting some of the items. While other methods are content-based in that they consider attributes (features of movies) of users and items, and profile attributes to make a recommendation, collaborative filtering uses the preferences and behaviors of other analogous users in making recommendations. This approach is based on the assumption that users whose preferences in the past were similar will very likely possess similar preferences in the future. In the context of our project, the relevance of collaborative filtering comes into play because it is one of the ways through which the large-scale user-item interaction data is analyzed to generate the personalized movie recommendations. Collaborative filtering tries to establish patterns from the viewing behaviors of the users in order to make appropriate suggestions of users' preference for optimized recommendations that would apply to all the interested parties of the whole movie-watching process. Moreover, by the practice of collaborative filtering on a Hadoop framework, the feature of scalability and parallel processing ability could be used to process systematically colossal data capacities. Subsequent sections of the report will elaborate on methodology, implementation, results, and implications of our personalized movie recommendation system using collaborative filtering on Hadoop. We rigorously analyze and evaluate to show the effectiveness and scalability of our proposed approach, which actually addresses the problem of improving users' satisfaction and activity in the context of movie recommendation.

V. Literature Review

Movie recommendation systems have recently attracted great attention in both academia and industry due to their major role in enhancing user satisfaction and engagement in the domain of digital entertainment. A number of works on research have been completed with accurate and effective recommendation system objectives. For instance, work by Shi et al. (2014) developed collaborative filtering techniques that aimed at working with the implementation of matrix factorization using user-item interaction data for the purposes of doing customized recommendations. On the other hand, work by Ekstrand et al. (2011) treated the use of content-based filtering techniques that analyze the attributes of movies and user profiles to give recommendations specifically designed for individual preferences. In the area of personalized movie recommendation systems, the strategies and algorithms used to fight the information overload problem and improve the quality of recommendations were quite different. Moreover, today, collaborative filtering and its user-based and item-based variants are amongst the most used recommendations techniques. With the advent of modern hybrid approaches in recent years, we are also witnessing the emergence of collaborative filtering combined with content-based filtering, in order to take advantage of the former's benefits while still avoiding some of the drawbacks that both have (Ortega et al., 2016). Progress in Big Data technologies, such as Hadoop and Spark, have therefore enabled the scalability and efficiency of recommendation systems. It has, therefore, allowed for easy processing and analysis of such big data containing millions of movie ratings and user interactions. Each of the personalized movie recommendation approaches has its strengths and weaknesses that should be gauged accordingly in the design and implementation of recommendations. They are pretty good at summarizing user preference and making very good recommendations. The drawback in this problem is the "cold start" issue: that new users and new items do not have enough data and therefore there are no data that could be used to make any meaningful recommendations. Contrarily, content-based filtering methods recommend items with great performance history based on their attributes, meaning there is no cold start problem in these methods (Loepp et al., 2019). However, it may give poor results in capturing the fine-grained user preference and may also result in a serendipity bias, wherein the user would like the items only if they are similar to those that the user has been liking in the past. Hybrid approaches try to take advantage of strong aspects of collaborative and content-based filtering, balancing their weaknesses at the same time, to provide a strategy of recommendations that would be able to enhance user satisfaction and interaction.

VI. Methodology

A. Dataset

The right selection of datasets is relevant to the consideration of validity and robustness in regard to our personalized movie recommendation system. Our criterion for the selection

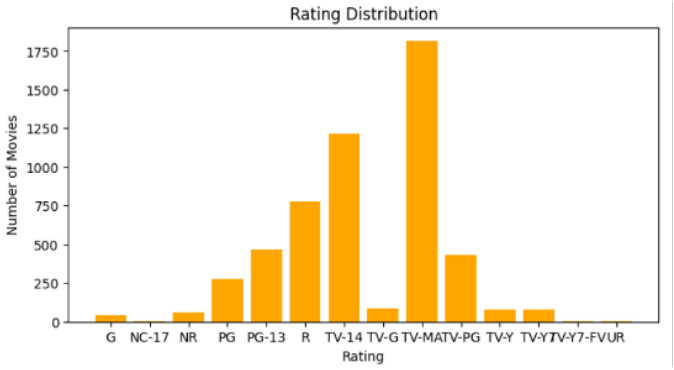

Fig. 1. Rating distribution

show_id_index	type_index	title_index	director_index	cast_index	country_index	rating_index	duration_index	listed_in_index
4616.0	0.0	3673.0	1822.0	2748.0	558.0	0.0	55.0	3.0
5248.0	1.0	4386.0	955.0	3275.0	2.0	1.0	149.0	258.0
0.0	0.0	4718.0	3679.0	3281.0	0.0	3.0	15.0	38.0
175.0	0.0	2146.0	1314.0	3013.0	281.0	0.0	42.0	0.0
955.0	0.0	2148.0	3341.0	3856.0	1.0	1.0	120.0	8.0
1151.0	0.0	1740.0	92.0	163.0	0.0	1.0	13.0	15.0
1216.0	0.0	1129.0	653.0	2683.0	0.0	3.0	2.0	147.0
1280.0	0.0	3269.0	37.0	2892.0	564.0	3.0	16.0	23.0
1755.0	0.0	686.0	1753.0	758.0	81.0	3.0	8.0	34.0
1945.0	0.0	2141.0	7.0	4171.0	0.0	5.0	39.0	80.0
1999.0	0.0	2142.0	2051.0	4170.0	0.0	5.0	24.0	309.0
2059.0	0.0	2143.0	2119.0	1269.0	0.0	5.0	9.0	180.0
2130.0	0.0	2144.0	2223.0	2951.0	0.0	5.0	5.0	180.0
2258.0	0.0	3645.0	1394.0	1280.0	410.0	2.0	34.0	11.0
2388.0	0.0	4929.0	82.0	1275.0	0.0	2.0	47.0	18.0
2582.0	0.0	2071.0	25.0	2562.0	7.0	1.0	10.0	35.0
2633.0	0.0	2072.0	25.0	54.0	7.0	1.0	10.0	35.0
2697.0	0.0	2073.0	25.0	2559.0	7.0	4.0	14.0	35.0
2759.0	0.0	2074.0	25.0	2558.0	7.0	4.0	18.0	35.0
2911.0	0.0	3021.0	108.0	53.0	7.0	1.0	6.0	35.0

only showing top 20 rows

Fig. 2. Top 20 rows

of a dataset is aimed at the use of datasets that will contain a reasonable number of user interactions and movie ratings, ensuring that the data drawn will be reasonable for any kind of analysis. We have done our best to consider a dataset with records over 8,000 to meet this criterion. We also focus on datasets which cover their range and, at least fully, represent either user taste or movie features with different genres, languages, and years of release. The process of selection included research and tracking down of the publicly available, citable dataset information from reputable sources, such as movie databases or academic repositories. We are going to critically appraise each dataset with an open mind to their magnitude, quality, and relevance to our research objectives before deciding on one to be adopted for the project

B. Collaborative Filtering Algorithm and Its Implementation

In recommendation systems, collaborative filtering is the most widely used method, particularly to try and exploit the collective knowledge of the system users, giving customized recommendations. The algorithm looks in the user-item interaction data, e.g., movie ratings data, for similarities of either users or items and hence predicts the preferences of users. Generally, there are two forms of collaborative filtering: user-based and item-based. While user-based collaborative filtering seeks similarities between users in terms of their past interactions with items, item-based collaborative filtering seeks to compute similarities between items based on users who have interacted with those items. For the implementation of collaborative filtering in our personalized movie

```
from pyspark.sql import SparkSession
from pyspark.sql.types import StructType, StructField, StringType, IntegerType, DateType
from pyspark.sql.functions import col, explode, count, split, isnan, when, desc, avg,expr
import matplotlib.pyplot as plt
from pyspark.ml.recommendation import ALS
from pyspark.ml.evaluation import RegressionEvaluator,MulticlassClassificationEvaluator
from pyspark.ml.feature import StringIndexer
from pyspark.ml.tuning import ParamGridBuilder, CrossValidator

import matplotlib.pyplot as plt
```

```
[ ] # Create a SparkSession
    spark = SparkSession.builder.appName("MovieRecommendation").getOrCreate()
```

Fig. 3. Pyspark implementation output

recommendation system, we will adopt a user-based approach due to its simplicity and effectiveness. The implementation involves several steps, including: • Preprocessing the dataset to handle missing values, normalize ratings, and remove outliers. • Calculating similarity scores between users based on their movie ratings using techniques such as Pearson correlation coefficient or cosine similarity. • Generating predictions for unseen movies for a given user by aggregating ratings from similar users weighted by their similarity scores. • Evaluating the performance of the recommendation system using metrics such as precision, recall, and mean absolute error. Big Data Algorithm Components Used In this project and in regard to our selected reference paper Aljunid and H (2018), appropriate Big Data technologies will be Apache Spark to address the large dataset with millions of user interactions and movie ratings. For such huge datasets, suppose millions of user interactions with movie ratings, some Big Data tools will be used. Apache Spark is extremely potent, giving superb dealing with large data and fast processing of enormous datasets in a distributed computing framework.

We used the abstractions provided by the Spark Resilient Distributed Dataset (RDD) to process data in parallel over a cluster of nodes. The code would be used for a highly scalable and optimized performance. Further, the rich ability of Spark's advanced analytics, like its machine learning library (MLlib), is very instrumental in affording us the ability to put the collaborative filtering algorithm in place effectively. Here, Apache Spark is used to enhance the processing speed of our personalized movie recommendation system and to obtain analytics at runtime of the system so that the recommendations can be delivered to the user at their required time.

VII. Results and Comparison

Our simulation for the personalized movie recommendation system yields a Root Mean Squared Error (RMSE) value of 3.7725. It's a metric of average value, showing at what rating, on average, prediction from our collaborative filtering-based algorithm deviates from the actual rating assigned by movie viewers. Even if our system shows reasonable performance, it is very important to compare these findings with the findings shown by the comparison paper in order to estimate more thoroughly how effective our suggested idea was for the tasks.

In the comparative paper, the recommendation system reports a very smaller value of RMSE at 1.0742. Besides this comparison, the paper also shows results that relate to

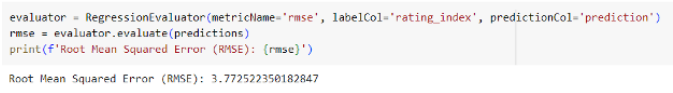

Fig. 4. Results after building the evaluator

the effect of parameter change on system performance. For example, regularization parameter (lambda) equals 0.2, and if the number of iterations is 15, then with this configuration, the running time is 1.4637, and a rank for the item being analyzed comes as 12. In these conditions, it clearly represents one of the lowest RMSE achieved, attaining 0.9167, which takes the minimum value of RMSE across all compared parameter configurations.

VIII. Potential Applications, Pros, and Cons

The prospective applications of our collaborative filtering personalized movie recommendation system over Apache Spark are sky-rocketing and diverse. Leading applications are within the digital entertainment platforms, whereby our system can be integrated for advanced user engagement and satisfaction through per-movie recommendation delivery. We have developed an algorithm that has advanced analyticals of user behaviors and preferences that will help in easy content discovery and to keep the user retained on the platform. The other area of application for our recommendation system is an e-commerce platform whose personalization of the recommendations can actually lead to a boosting in sales and improving the customer experience. For example, movies that are up for sale. However, the proposed approach may have some drawbacks, but still, it is very scalable, effective, and accurate for producing personalized recommendations. Drawbacks could include algorithmic bias, where the viewer of the information gets to see only a narrow scope of content based on what they had previously interacted with and thus doesn't obtain a diversified recommendation. Furthermore, privacy concerns may arise due to the collection and analysis of user data for recommendation purposes. However, the barriers we face in increasing user satisfaction and engagement with our methodology are far outweighed by the benefits it accrues for the benefit of personalized content recommendation across domains.

IX. conclusion

In conclusion, The collaborative filtering done on the personalized movie recommendation system that is being built on Apache Spark will offer us a wide array of how prospective applications work, particularly when it comes to digital entertainment platforms and e-commerce sites. Leveraging this kind of advanced algorithm which will allow us to analyze the user's behaviors and preferences, will enable us to facilitate seamless content discovery and this will eventually result in enhancing the user's retention on the platform targeting more users' time on our platforms. Our system's integration into the e-commerce platforms will help us boost sales and can lead to an overall customer experience which will be done by providing personalized movie recommendations for each user according to his / her interests. The approach we followed will significantly boost scalability, improving effectiveness and boosting accuracy in delivering recommendations. However, this algorithmic bias factor could limit in allowing the diversity of recommendations which potentially constrains users to a slightly narrow scope of content that is based on previous interactions only. Here, another factor we could really consider is privacy constraint which could arise due to the collection of user data that is being used for recommendation purposes. Even with these challenges, the benefits could significantly improve user satisfaction and engagement towards our platforms which will mostly outweigh the challenges we face. Moving forward, this continued refinement in our algorithm and the privacy measures must be essential in enhancing our performance and trustworthiness in our filtering process. Overall, our system stands mainly to revolutionize the user experiences according to their interests.

X. Acknowledgment

Our team members are very grateful for our project supervisor for their continued guidance, and support which had been throughout each and every step of our research process. Their expertise in the subject and direction had been significantly important in shaping our output of the work . We want to express out gratitude to each of the member present in our team for their unmatched dedication, cooperation and their input during the process. Finally, we wish to recognize the writers of the research papers which were crucial to our study.Their input in this field has been extremely valuable in guiding our project and discussions.

...

References

[1] Y. Deldjoo, M. Elahi, M. Quadrana, P. Cremonesi, and F. Garzotto, "Toward effective movie recommendations based on mise-en-scène film styles," in *Proceedings of the 11th Biannual Conference of the Italian SIGCHI Chapter*, Sep. 2015, pp. 162–165.

[2] J. Beel, C. Breitinger, S. Langer, A. Lommatzsch, and B. Gipp, "Towards reproducibility in recommender-systems research," *User Modeling and User-Adapted Interaction*, vol. 26, pp. 69–101, 2016.

[3] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Computing Surveys*, vol. 47, no. 1, pp. 1–45, 2014.

[4] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Foundations and Trends in Human–Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2011.

[5] B. Loepp, T. Donkers, T. Kleemann, and J. Ziegler, "Interactive recommending with Tag-Enhanced Matrix Factorization (TagMF)," *International Journal of Human-Computer Studies*, vol. 121, pp. 21–41, 2019, doi: 10.1016/j.ijhcs.2018.05.002.

[6] F. Ortega, A. Hernando, J. Bobadilla, and J. H. Kang, "Recommending items to group of users using Matrix Factorization based Collaborative Filtering," *Information Sciences*, vol. 345, pp. 313–324, 2016, doi: 10.1016/j.ins.2016.01.083.

[7] M.F. Aljunid and M. D. H, "Movie Recommender system based on collaborative filtering using Apache Spark," ResearchGate, 2018. [Online].