Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

BIG DATA TECHNOLOGY MINI PROJECT: Recommendation System

Submitted to faculty: Prof. Sagar Apune

INDEX:

1. Introduction

A recommender system can be defined as “Any system that produces individualized recommendations as output or has the effect of guiding the user in a personalized way to interesting or useful objects in a large space of possible options” (Burke 2002).

Recommender systems form a specific type of information filtering (IF) technique that attempts to present information items (movies, music, books, news images, web pages, etc.) that are likely of interest to the user. Typically, a recommender system compares the user's profile to some reference characteristics, and seeks to predict the 'rating' that a user would give to an item they had not yet considered.

The myriad approaches to Recommender Systems can be broadly categorized as - Collaborative Filtering (CF): In CF systems a user is recommended items based on the past ratings of all users collectively. Content-based recommending: These approaches recommend items that are similar in content to items the user has liked in the past, or matched to attributes of the user. Hybrid approaches: These methods combine both collaborative and content based approaches.

2.Team Members:

1032210504 Riya Bambal
1032210519 Raina Basu
1032210794 Pranav Pisal
1032210755 Devanshu Surana
1032210826 Abhilash Kashid

3.Overview: Big Data Technology:

1.Hadoop

- Hadoop:

  - Cluster: 3 Nodes (1 NameNode and 2 Data Nodes)

  - Version: Hadoop 2.8.0

  - Components: Pig, Hive, Sqoop

Setting up a recommendation system using **Hadoop** can be a powerful way to process large amounts of data and generate personalized recommendations efficiently. Hadoop is a framework that enables the distributed processing of massive datasets across clusters of computers. Here's a high-level overview of how you could approach building a recommendation system using Hadoop:

Data Collection and Preparation: Gather the relevant data for your recommendation system. This could include user interactions, item data, and other relevant information. Prepare and preprocess the data to make it suitable for analysis.
Data Storage: Hadoop's HDFS (Hadoop Distributed File System) can store the large datasets required for recommendation systems. The data is stored in a distributed manner across the Hadoop cluster.
Data Processing: Utilize Hadoop's MapReduce programming model to process the data. This involves writing Map and Reduce functions that process and aggregate the data to generate user-item interaction matrices, similarity matrices, or any other relevant data structures.

2.Mongodb:

- MongoDb:

  - Version: 3.4.5

MongoDB, a leading NoSQL database, is a preferred choice for recommendation systems due to its flexible data model and seamless scalability. It efficiently handles real-time user interactions

and integrates with machine learning algorithms, enabling businesses to deliver personalized recommendations, enhancing user experience and engagement.

4. Workflow:

1. Data Collection and Ingestion:
   - Gather raw data from various sources, such as user interactions, item details, and contextual data.
   - Ingest data into Hadoop HDFS for initial storage and processing.

2. Data Preprocessing and Cleaning:
   - Cleanse the data to remove inconsistencies, handle missing values, and format it for analysis.
   - Use Hadoop MapReduce jobs to preprocess large datasets efficiently.

3. Data Transformation and Feature Engineering:
   - Transform raw data into suitable formats for recommendation algorithms.
   - Perform feature engineering to create relevant features, enhancing the recommendation model's accuracy.

4. Recommendation Model Development:
   - Utilize machine learning algorithms like collaborative filtering, content-based filtering, or hybrid methods.
   - Train recommendation models on Hadoop clusters, leveraging distributed computing power.

5. Real-time Data Processing and Analysis:
   - Implement real-time data processing using Pig.
   - Process real-time user interactions and update recommendations dynamically.

6. Data Storage and Retrieval using MongoDB:
   - Store preprocessed and transformed data in MongoDB collections.
   - Utilize MongoDB's flexible querying for quick data retrieval based on user preferences.

7. Integration of Hadoop and MongoDB:

- Use Hadoop MapReduce for batch processing tasks, transforming large datasets.
- Integrate Hadoop with MongoDB using connectors for seamless data exchange between systems.

8. Deployment and Scalability:
- Deploy the recommendation system on a scalable Hadoop cluster and MongoDB database.
- Monitor system performance and scale resources as data volume and user base grow.

5. Future Scope and Conclusion

Future Scope:-

The field of smart recommender systems using Hadoop and MongoDB is dynamic and ripe with opportunities for further research and development.
Some potential future scopes for this technology:
1.Machine Learning Enhancements: Incorporating advanced machine learning algorithms and deep learning techniques into the recommender system can improve recommendation quality and accuracy.
2.Enhanced User Experience: Focus on creating a seamless user experience by incorporating user feedback loops, real-time analytics, and user interface improvements. Developing mobile apps and web interfaces that integrate with the recommender system can enhance the overall user experience.
3.Big Data Integration: As data continues to grow, integrating more data sources, including unstructured data from social media and IoT devices, can further enrich the recommender system's understanding of user preferences and behaviors.
4.Adaptive Learning: Implementing adaptive learning and reinforcement learning techniques to fine-tune the recommendation system in real-time, adapting to user behavior changes and preferences.
5.Scalability and Optimization: With the ever-increasing data volume, it's essential to continue optimizing the system for scalability and performance. Investigating new technologies and frameworks that can handle even larger datasets efficiently is important.

6. Conclusion:-

In this report, we have explored the development and implementation of a smart recommender system using Hadoop and MongoDB. This technology stack offers a robust and scalable solution for building recommendation engines, leveraging the power of big data processing and NoSQL databases. Through our analysis,

we can draw several key conclusions:
1.Scalability and Performance: Hadoop's distributed computing framework allows for the efficient processing of large datasets, making it well-suited for recommendation systems that must handle massive amounts of data.
2.Personalization: With the ability to process vast amounts of user data, the recommender system can provide highly personalized recommendations, improving user engagement and satisfaction.
3.Real-time Processing: The combination of Hadoop and MongoDB enables real-time data processing and recommendations, ensuring that users receive up-to-date suggestions based on their interactions and preferences.
4.Flexibility: The use of MongoDB as the database offers flexibility in data modeling, making it easier to adapt to changing data structures and requirements.

7. References

https://www.nvidia.com/en-us/glossary/data-science/recommendation-system/#:~:text=A%20recommendation%20system%20is%20an,recommend%20additional%20products%20to%20consumers.

https://unogeeks.com/recommendation-system-using-hadoop/