

Movie Recommendation System: Personalized and Scalable

This project showcases a personalized movie recommendation system built using Python, machine learning algorithms, and advanced techniques such as collaborative filtering, recency boosting, and clustering. The system is designed to recommend movies based on user behavior and preferences. It can handle large datasets and offers scalability using FAISS for efficient similarity searches.

Key Features:

1. Data Preprocessing:

- The dataset includes movie ratings, timestamps, and user information. Data is cleaned and preprocessed, including encoding user IDs and movie IDs for machine learning models.
- Temporal features such as recency (time since last interaction) and frequency (how often the user has rated movies) are calculated and normalized to boost recommendation relevance.

2. Collaborative Filtering with SVD:

- The system uses **Singular Value Decomposition (SVD)** from the surprise library to predict ratings and recommend movies. It achieves an RMSE score to evaluate prediction accuracy.
- This technique ensures that recommendations are made based on similar user behavior.

3. Personalized Recommendation Boost:

- A recency boost is applied to recommendations, prioritizing more recent interactions. This approach ensures that users are recommended movies that align with their recent viewing activity.

4. Clustering for User Segmentation:

- Users are grouped into clusters based on their recency and frequency of interaction with movies. **KMeans** clustering is used to identify user segments, enabling more targeted recommendations.
- Each cluster is analyzed to recommend movies that are popular among users with similar preferences.

5. Genre-Based Recommendations:

- Genres are encoded and vectorized for each movie, allowing the system to recommend movies with similar genre preferences.
- This genre-based filtering is integrated into the clustering model, ensuring that users get recommendations that match both their activity patterns and genre interests.

6. Scalability with FAISS:

- FAISS (Facebook AI Similarity Search) is used to perform efficient similarity searches across a large dataset, enabling real-time recommendations even with extensive movie libraries.

User Interaction:

Users can input their **userId**, and the system will recommend movies based on their preferences, recent activity, and genre interests. The recommendation considers both individual preferences and group behavior patterns derived from clustering.

Technologies Used:

- **Machine Learning:** SVD, KMeans, Scikit-Learn
- **Libraries:** Surprise, FAISS, SHAP, Pandas, NumPy, Matplotlib
- **Scalability:** FAISS for fast, large-scale similarity search

Unique Aspects:

- **Personalization:** Dynamic recommendations based on recency and frequency boost.
- **User Segmentation:** Clustering users for more tailored recommendations.
- **Scalability:** Utilization of FAISS for handling large datasets and quick similarity searches.
- **Explainability:** SHAP values are incorporated for model transparency and interpretation.