Course : Deep Learning and Reinforcement Learning

Project Title : AI Based Movie Recommendation

Mohith Gowda H K	1BG23CS082
Mohith S Kumar	1BG23CS083
Prajwal Kumar T	1BG23CS106

Project Objective

The objective of the AI-based movie recommendation system is to provide users with personalized movie suggestions using machine learning and data analysis techniques. It aims to analyze user preferences, behavior, and ratings to enhance the viewing experience. The system adapts to individual tastes over time for more accurate recommendations. This ensures efficient content discovery and higher user engagement.

- Personalized Recommendations: Suggests movies based on user history and preferences.
- Machine Learning Integration: Uses algorithms like collaborative and content-based filtering.
- Real-Time Adaptation: Continuously updates recommendations as user behavior changes.
- Scalable System Design: Capable of handling large user and movie datasets efficiently.

Methodology and Workflow:

The AI-based movie recommendation system follows a structured methodology combining data collection, preprocessing, model training, and evaluation. It starts by gathering user and movie data, then cleans and processes it for use in training algorithms.

- Dataset loaded from directory with labeled gesture classes.
- Data Preprocessing: Cleans, formats, and normalizes data for model input.
- Collaborative Filtering: Predicts preferences based on similar user behavior.
- Content-Based Filtering: Recommends movies using item features like genre or actors.
- Model Training and Testing: Builds and evaluates ML models for accuracy.
- Feedback Loop: Incorporates user feedback to refine and improve recommendations.

Flowchart

Key Assumptions

The AI-based movie recommendation system assumes that users' past behaviors and preferences accurately reflect their future choices. It also assumes the availability of sufficient and relevant data for effective learning. The system relies on the assumption that user feedback is honest and consistent.

- User Behavior Consistency: Assumes past preferences predict future interests.
- Data Availability: Relies on rich datasets for model accuracy.
- Honest Feedback: Trusts user ratings and interactions reflect true opinions.
- Feature Relevance: Believes movie attributes impact recommendation success.

Model Evaluation and Analysis

The AI-based movie recommendation system undergoes evaluation to ensure accuracy, relevance, and user satisfaction. Performance is measured using metrics like precision, recall, and RMSE. It also involves analyzing user engagement and recommendation diversity.

- Performance Metrics: Uses precision, recall, F1-score, and RMSE for accuracy checks.
- User Engagement Analysis: Evaluates how users interact with recommended content.
- Diversity and Novelty Checks: Ensures varied and fresh recommendations.
- Continuous Improvement: Regularly updates models based on evaluation feedback.

Project Summary and Outcomes

This project builds a Content-Based Movie Recommendation System using movie metadata such as genres, cast, crew, keywords, and overview. It processes and combines these textual features into a unified format and transforms them into numerical vectors. Cosine similarity is then used to compute how closely movies are related. The system ultimately recommends the top similar movies based on user input.

- Extracts and cleans movie data from multiple sources (movies and credits datasets).
- Combines essential features (genres, cast, overview, director) into one unified tag.
- Applies NLP preprocessing including stemming and vectorization using CountVectorizer.
- Uses cosine similarity to identify and recommend the most similar movies.

Future Improvements and Extensions

Future versions of the model can benefit from additional training data, augmentation techniques, and deployment into mobile platforms.

- Integrate Collaborative Filtering:Add user-based or item-based collaborative filtering using user ratings to complement content-based recommendations.
- Incorporate Deep Learning Models:Use transformer-based models (e.g., BERT or Sentence-BERT) to capture deeper contextual meaning in movie descriptions and tags.
- Enhance Frontend Integration:Build a full-featured web or mobile app with personalized user profiles and real-time recommendation updates.
- Add Advanced Metadata:Include additional features like release year, language, runtime, and streaming availability to refine and filter recommendations.

Reflections and Learning Outcomes

Through this project, we gained practical experience in applying Natural Language Processing (NLP) techniques to real-world data. We explored the use of feature engineering, text vectorization, and similarity metrics to build a functional recommendation engine. The process enhanced our understanding of data preprocessing, model deployment, and performance evaluation.

- Understood and implemented content-based filtering using NLP and cosine similarity.
- Learned data cleaning and feature extraction from complex, multi-source datasets.
- Applied stemming, vectorization, and dimensionality reduction techniques.
- Practiced deploying models and exporting data for frontend application use.

Appendix

The appendix includes visual and technical details supporting the model's performance and implementation. These can aid in better understanding and presentation of results.

- Used Python libraries like pandas, nltk, scikit-learn, and pickle for data processing and model development.
- Processed TMDB 5000 Movies and Credits datasets to extract relevant features such as genres, cast, and overview.
- Implemented custom functions for JSON parsing, text cleaning, stemming, and similarity calculation.
- Exported essential artifacts (movies.pkl, similarity.pkl, movies.json) for frontend integration and deployment.

Thank you