Movie Recommendation System

📌 **Overview**

The **Movie Recommendation System** is built using **machine learning techniques** to suggest movies based on user preferences and past interactions. The project is implemented in **Jupyter Notebook** and leverages **data science and AI algorithms** to provide accurate recommendations.

🚀 **Features**

- **Data Preprocessing & Cleaning**: Handles missing values, duplicates, and normalizes data.

- **Recommendation Algorithms**:

    - **Content-Based Filtering**: Suggests movies based on movie attributes (genre, director, etc.).

    - **Collaborative Filtering**: Uses user interactions to find similar preferences.

    - **Hybrid Approach**: Combines multiple recommendation techniques for better accuracy.

- **Visualization of Movie Trends**: Provides charts and graphs to analyze movie popularity and trends.

- **User-Friendly Interface**: Simple workflow for generating recommendations within Jupyter Notebook.

🛠 **Installation**

1. Clone the repository:

2. git clone https://github.com/sakethvemula8/movie-recommendation-system.git

3. Navigate to the project directory:

4. cd movie-recommendation-system

5. Install dependencies:

6. pip install -r requirements.txt

📖 **Usage**

1. Open the Jupyter Notebook:

2. jupyter notebook

3. Load the dataset in the appropriate directory.

4. Run the notebook to train and generate movie recommendations.

📊 Dataset

- The project uses a **publicly available movie dataset** (e.g., **MovieLens, IMDB, TMDB**).

- Ensure that the dataset is in the correct directory before running the notebook.

🎯 How It Works

1. **Data Preparation**: Cleans and preprocesses movie data.

2. **Feature Engineering**: Extracts meaningful insights from the dataset.

3. **Model Training**: Uses recommendation algorithms to find patterns.

4. **Generate Recommendations**: Suggests personalized movie lists for users.

5. **Evaluation**: Measures of the performance of the recommendation system.

🤝 Contributing

Contributions are welcome! Feel free to fork the repository and submit a pull request. Here's how you can contribute:

- Improve existing recommendation models.

- Optimize performance and accuracy.

- Add new datasets or visualization features.