Project Report: Movie Recommendation Chatbot

Prepared by:

Name: Sohel Rana

Position: Digital Currency Investor & Technical Lead

Nationality: Bangladeshi

1. Introduction

The Movie Recommendation Chatbot Project is designed to provide users with personalized movie suggestions through a conversational interface. Built with Streamlit, the chatbot leverages the IMDb open datasets to deliver recommendations without requiring external API keys. The project emphasizes local deployment, ensuring accessibility and independence from cloud services.

2. Objectives

- Develop a chatbot capable of understanding natural language queries about movies.
- Automate the download and integration of IMDb datasets for up-to-date information.
- Provide recommendations based on title, genre, and rating.
- Deploy the chatbot locally on a PC/server using Streamlit.
- Create a modular and scalable codebase for future enhancements.

3. Project Code Structure (tree diagram)

```
movie_chatbot_project/
|
|-- app.py                # Main Streamlit app entry point
|-- requirements.txt      # Python dependencies
|
|-- chatbot/              # Chatbot logic package
|   |-- __init__.py       # Marks folder as a package
|   |-- logic.py          # Handles user queries
|   |-- recommender.py    # Recommendation engine (IMDb dataset)
|   |-- utils.py          # Helper functions
|
|-- services/            # Service integrations
|   |-- __init__.py       # Marks folder as a package
|
|-- data/                # IMDb datasets downloaded automatically
```

4. Workflow

1. Dataset Download

- IMDb datasets (title.basics.tsv.gz, title.ratings.tsv.gz) are automatically downloaded and stored in the data/ folder.

2. Data Processing

- Merging of title.basics and title.ratings ensures movies are enriched with genres, release years, and ratings.

3. Recommendation Engine

- Uses TF-IDF vectorization and cosine similarity to match user queries with relevant movies.

4. Chatbot Logic

- Processes user input, queries the recommender, and formats responses.

5. Streamlit UI

- Provides a chat-like interface (st.chat_input, st.chat_message) for seamless user interaction.

6. Local Deployment

- Run via streamlit run app.py, accessible at http://localhost:8501.

5. Features

- Movie Recommendations by genre, title, or keyword.
- IMDb Ratings Integration for credibility.
- Automated Dataset Handling (download + merge).
- Local Deployment for offline-first environments.
- Modular Codebase for scalability and future improvements.

6. Use Cases

- Entertainment Platforms: Provide personalized movie suggestions.

- Educational Projects: Demonstrate NLP and recommendation systems.
- Local Deployments: Useful in environments with limited internet access.
- Data Science Training: Showcase integration of open datasets with ML techniques.

7. Future Enhancements

- Add poster images and trailers for richer recommendations.
- Support regional filters (e.g., Asia-specific cinema).
- Integrate advanced NLP models for better query understanding.
- Extend deployment to mobile-friendly interfaces.

8. Conclusion

The Movie Recommendation Chatbot Project successfully demonstrates how open datasets, machine learning techniques, and Streamlit can be combined to create a practical, user-friendly application. With its modular design and automated dataset handling, the project is well-positioned for future enhancements and broader adoption.