■ Movie Recommender Bot — Project Report

Introduction & Motivation

The Movie Recommender Bot is designed to help users interact with a movie knowledge base in a conversational manner. It supports Q&A; about movies, summarization of reviews, and recommendation of similar titles. The motivation stems from the need for personalized discovery and intelligent assistance in navigating the vast movie landscape.

Architecture (LangChain Components Used)

The application integrates LangChain and Streamlit, structured as follows:

- Frontend: Streamlit with sidebar configuration and tabs for Q&A;, Summarization, and Recommendations.
- VectorStore: FAISS used to store embeddings of movie plots and metadata.
- Embeddings: SentenceTransformers (all-MiniLM-L6-v2) for semantic similarity.
- LangChain Chains: RetrievalQA for Q&A; and Summarization Chain (map-reduce) for reviews.
- LLMs: OpenAI (gpt-4o-mini) or Ollama (llama3.1) as configurable backends.
- Data: Seed CSV (sample_movies.csv) and reviews (reviews.jsonl), with support for user-uploaded CSVs.

Challenges Faced & Solutions

During development, several challenges were encountered and resolved:

- Embedding Search Error: FAISS required NumPy arrays; solved by converting vectors or using similarity_search_with_score.
- Streamlit Compatibility: Adjusted progress display for version differences.
- CSV Upload Variability: Normalized column names and validated schema.
- Review Summarization: Improved results using map-reduce summarization chain.

Sample Inputs & Outputs

Examples of how the system responds to queries are shown below:

- Q&A;: Input: 'Explain Parasite in one line' → Output: 'Parasite is a darkly comic thriller exposing class divides.'
- Summarization: Input: 3 reviews for Inception → Output: Concise pros/cons summary highlighting concept, visuals, and complexity.
- Recommendations: Input: 'By title → Interstellar' → Output: Inception, The Martian, Gravity, Arrival (with rationale).

Potential Improvements

Future work could enhance the system with additional features and optimizations:

- Integration with real-time APIs (TMDb/OMDb) for live ratings and data.
- User profiles for personalized recommendations.
- Hybrid recommendation system combining semantic and collaborative filtering.
- Sentiment-aware review summarization.
- Conversation memory for multi-turn contextual queries.
- Multimodal inputs (trailers/posters) for richer insights.