Movie Dialog QA Bot — Semantic Search System

Overview

This project implements a **semantic search application** over the Cornell Movie-Dialogs Corpus. Users can ask natural-language questions, and the system retrieves the most semantically relevant movie dialog lines as grounded evidence, without generative hallucination.

The focus of this project is **retrieval quality, system design, and explainability**, rather than end-to-end text generation.

Data Source

- **Dataset:** Cornell Movie-Dialogs Corpus
- **Access Method:** ConvoKit (`movie-corpus`)
- **Content:** ~304k utterances from 617 movies with rich metadata
- **Key Fields Used:**
    - Utterance text
    - Conversation ID
    - Speaker ID & character name
    - Movie title, year, genre, IMDb rating, votes

System Architecture

1. Data Processing

- Utterances were flattened into a tabular structure (one row per dialog line).
- Conversation-level metadata was joined via conversation IDs.
- Final dataset stored as a Parquet file for efficient access.

2. Embedding & Indexing

- **Embedding Model:** `sentence-transformers/all-MiniLM-L6-v2`
- Each utterance was embedded independently.
- Embeddings were **L2-normalized**.
- **Similarity Metric:** Cosine similarity
- **Index:** FAISS `IndexFlatIP`

3. Search Flow

1. User query is embedded using the same model.
2. FAISS retrieves top-K nearest neighbors.
3. Results are filtered by a similarity threshold.
4. Matching dialog lines are displayed with metadata as grounded evidence.

Deployment Design

To support deployment constraints:

- The FAISS index and metadata are prebuilt offline.
- Index artifacts are hosted on **Hugging Face Datasets**.
- On the first app run, the index is downloaded and cached locally.
- This avoids recomputation and reduces memory usage on Streamlit Cloud.

User Interface

- Built using **Streamlit**
- Chat-style query input
- Adjustable result count and confidence threshold

- Expandable evidence panels showing:
    - Character name
    - Movie title & year
    - Dialog line text

Known Limitations

- Retrieval is **embedding-only**, so semantic similarity can surface logically opposite statements (e.g., refusal vs reluctant apology).
- Indexing is done at the **single-utterance level**, limiting conversational context.

Future Improvements

- Hybrid scoring (semantic + keyword heuristics)
- Context-window indexing across multiple utterances
- Reranking using cross-encoder models
- Optional LLM-based reasoning over retrieved results

Conclusion

This project demonstrates a complete, explainable semantic search pipeline:

- Robust data handling
- Correct similarity modeling
- Scalable deployment strategy
- Clear identification of limitations and trade-offs

The system prioritizes **grounded retrieval and interpretability** over uncontrolled generation, making it suitable as a foundation for more advanced conversational AI systems.