Week 7-Hands-On: Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs)

Report on AI-Powered Job Search Chatbot

1. Overview

This project focuses on developing an **AI-powered job search chatbot** that enables users to find job postings by querying an intelligent system. The chatbot uses **Retrieval-Augmented Generation (RAG)** principles to fetch relevant job postings based on user input and provides a seamless conversational experience.
The system is implemented using:

- **FAISS** for fast document retrieval.
- **SentenceTransformers** for generating text embeddings.
- **Streamlit** for an interactive user interface.
- **Cloudflare Tunnel** for deployment in Google Colab.

2. Project Goals and Scope

Use Case:

The chatbot helps users efficiently find job postings by querying a structured job dataset. It retrieves relevant job descriptions based on semantic similarity and presents results in a user-friendly manner.

Main Objectives:

- Develop a Retrieval-Augmented Chatbot for job searches.
- Implement FAISS indexing for fast job retrieval.
- Use SentenceTransformers (all-MiniLM-L6-v2) for embeddings.
- Create an interactive Streamlit-based UI.
- Deploy via Cloudflare Tunnel for external access.

3. Implementation Process

Step 1: Data Preparation

Dataset Used: job_postings.csv

- The dataset was preprocessed to extract relevant fields like job title, company, location, and job links.
- Embeddings **were generated** for each job posting using SentenceTransformers.

Step 2: Job Retrieval Using FAISS

- FAISS (Facebook AI Similarity Search) was used for efficient vector-based retrieval.

- Each job posting was **converted into an embedding** and stored in a FAISS index.
- The chatbot fetches **top-k matching job descriptions** based on the **semantic similarity** of user queries.

Step 3: Streamlit-Based UI

- Streamlit **was used** to build an interactive chatbot UI.
- Users can enter **job-related queries** (e.g., "Show me machine learning engineer jobs").
- The chatbot returns the **most relevant job postings** with job details and an "Apply Here" link.

Step 4: Deployment via Cloudflare Tunnel

- Google Colab was used for hosting the app.
- The chatbot was made **publicly accessible** via a generated Cloudflare link.

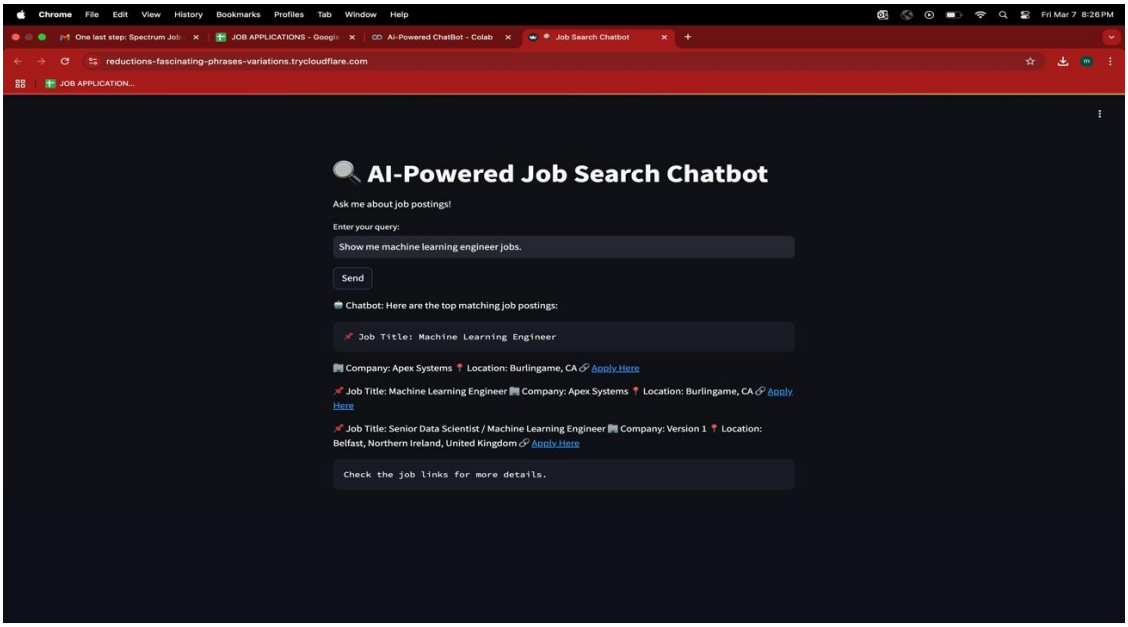
4. Challenges and Solutions

- FAISS Indexing Issues → Fixed by ensuring correct dimensions and structure for FAISS storage.
- Import Errors in Colab → Used %run retriever.py instead of direct import retriever.
- LocalTunnel Asking for Password → Switched to Cloudflare Tunnel for easy external access.

5. Features of the Chatbot

- Fast Job Search: Uses FAISS for quick retrieval.
- Smart Query Matching: Uses SentenceTransformers for semantic understanding.
- Interactive UI: Built with Streamlit.
- Accessible Anywhere: Hosted via Cloudflare Tunnel.
- Job Application Links: Provides direct apply links for users.

6. Outputs


Chrome File Edit View History Bookmarks Profiles Tab Window Help
Fri Mar 7 8:26 PM
One last step: Spectrum Job
JOB APPLICATIONS - Google
AI-Powered ChatBot - Colab
Job Search Chatbot
reductions-fascinating-phrases-variations.trycloudflare.com
JOB APPLICATION...
AI-Powered Job Search Chatbot
Ask me about job postings!
Enter your query:
Show me machine learning engineer jobs.
Send
Chatbot: Here are the top matching job postings:
Job Title: Machine Learning Engineer
Company: Apex Systems Location: Burlingame, CA Apply Here
Job Title: Machine Learning Engineer Company: Apex Systems Location: Burlingame, CA Apply Here
Job Title: Senior Data Scientist / Machine Learning Engineer Company: Version 1 Location: Belfast, Northern Ireland, United Kingdom Apply Here
Check the job links for more details.