Cloud Environment Setup

The initial phase involved setting up the working environment in Azure. This required creating a Blob Storage Account, an Azure Databricks Workspace, and mounting the Blob Storage to the Databricks File System (DBFS). These steps ensured a seamless integration of cloud storage and computing resources, facilitating easy access to sample HR documents for the chatbot.

- **Blob Storage Account**: Created to store HR documents, enabling centralized data management and access.
- **Databricks Workspace**: Set up on the 14-Day Trial pricing tier, serving as the primary environment for developing and executing the chatbot application.
- **Mounting Blob Storage**: Utilizing **dbutils.fs.mount**, the Blob Storage container was mounted to DBFS, providing direct access to HR documents from the Databricks notebooks.

Data Preparation & Vector Database Setup

Data preparation involved extracting, cleaning, and chunking the HR documents, followed by embedding these chunks for efficient retrieval. Tools and libraries such as **langchain**, **chromadb**, and various document loaders (**TextLoader**, **PyPDFLoader**, **DirectoryLoader**, **Docx2txtLoader**) were employed for this process.

- **Document Extraction**: PDF and DOCX documents were loaded using respective loaders, ensuring a comprehensive extraction of HR data.
- **Cleaning and Chunking**: The extracted text was cleaned to remove excessive spaces and line breaks, then chunked into manageable sizes using **RecursiveCharacterTextSplitter**.
- **Vector Database Setup**: Each text chunk was embedded using OpenAI Embeddings and stored in a Chroma vector database, facilitating efficient semantic retrieval based on query relevance.

Chatbot Implementation

The chatbot implementation leveraged the RAG workflow, grounding the Large Language Model (LLM) on the vector database and utilizing prompt engineering for interactive responses.

- **LLM Initialization**: A **ChatOpenAI** object was initialized with specific parameters (e.g., model version, temperature) for conversational response generation.
- **Retrieval Augmented Generation (RAG)**: A custom RAG workflow was created, combining document retrieval with the LLM to answer queries based on HR documents. This involved developing custom prompt templates and utilizing the **RetrievalQA** class from **langchain** for question answering.
- **Interactive QA**: The chatbot was tested with various HR-related queries, showcasing its ability to retrieve and utilize information from the HR documents to provide accurate and helpful responses.

Chatbot Response Samples

This section provides examples of questions asked by users and the corresponding responses given by the chatbot, showcasing its capability to leverage HR documents and provide informative answers.

1. **Question: "Give list of federal holidays?"**
   Answer: "The list of federal holidays for 2023-2024 is as follows:
   - New Year's Day
   - Birthday of Martin Luther King, Jr.
   - Washington's Birthday
   - Memorial Day
   - Juneteenth National Independence Day
   - Independence Day
   - Labor Day
   - Columbus Day
   - Veterans Day
   - Thanksgiving Day
   - Christmas Day

   Thanks for asking!"

2. **Question: "What are the employee policies?"**
   - Answer: "The employee policies at {ORGANIZATION NAME} are outlined in the Employee Handbook. This includes guidelines on privileges and obligations of employment, the voluntary at-will employment policy, and the ability for policies to be modified, amended, or revoked by the organization at any time without advance notice. Thanks for asking!"

3. **Question: "Give details about the retirement plan?"**
   - Answer: "The retirement plan at {ORGANIZATION NAME} is available for eligible full-time and part-time employees who are 21 years of age or older. The organization contributes to the employee's retirement plan when the employee becomes vested after one year of employment. More information about the retirement plan will be provided to you at the time of employment. Thanks for asking!"

4. **Question: "What do I do if I find a glitch?"**
   - Answer: "Thanks for asking! If you find a glitch, the first thing you should do is to document the details of the glitch, including screenshots and steps to reproduce the issue. This will help the development team to investigate and isolate the problem. Once you have documented the glitch, you can create a bug report following the guidelines provided in the HR document 'Creating Bug Reports (COE-KG-XXX)'. Remember to prioritize the glitch based on its impact on core functionalities, and provide as much information as possible to help with the investigation."

5. **Question: "At what point do I get health benefits after joining?"**

- Answer: "Coverage will be effective the 1st of the month following 30 days of employment. For example, if you are hired on February 11th, your coverage will be effective on April 1st. Thanks for asking!"

6. **Question: "How serious does a bug need to be to get top priority?"**
- Answer: "Thanks for asking! According to our HR documents, bugs that affect the core functionalities of the product are considered high priority. These are the bugs that significantly impact the performance or usability of the product. Bugs that have workarounds but still degrade the core functionalities are considered medium priority, and bugs that do not affect critical/core functions of the product are low priority. I hope this helps clarify the seriousness of bug priorities."

7. **Question: "What's the remote work policy look like?"**
- Answer: "The remote work policy allows employees to work from home or another location outside of the office. It outlines the expectations and guidelines for remote work, including communication, work hours, and equipment usage. Thanks for asking!"

8. **Question: "What options do I have for dental care?"**
- Answer: "You have the option to receive dental services from any dental provider with our open access Dental PPO plan. This plan allows you to receive services from both in-network and out-of-network providers without the need for a Primary Dental Provider or referrals to specialists. The plan also covers a range of services, including preventative services such as exams, cleanings, and X-rays, as well as basic and major services. If you need to find an in-network dentist, you can search for a participating provider through our network."

9. **Question: "Need to update my address. Who do I talk to?"**
- Answer: "You will need to notify your supervisor in writing of any changes in your personal mailing address. Thanks for asking!"

10. **Question: "Can you show me what a good bug report is?"**
- Answer: "Thanks for asking! A good bug report should include proof that you observed the problem, a general location or starting point to investigate the issue, and steps to reproduce the bug. It should also prioritize the defect list into three main categories: high, medium, and low priority. Screenshots and error message capturing are also important in a bug report. If you need more detailed information, you can refer to the document 'Creating Bug Reports (COE-KG-XXX)' for a guide on how to handle bugs and create bug reports in user stories."

Key Technologies and Frameworks Used

- **Azure Databricks**: Served as the primary platform for development and execution, offering a robust environment for data science and AI projects.
- **Azure Blob Storage**: Used for storing HR documents, ensuring scalable and secure data management.
- **LangChain**: Played a crucial role in data preparation, embedding, and the RAG workflow, facilitating the integration of retrieval-augmented generation capabilities.
- **ChromaDB**: Selected for the vector database, enabling efficient semantic search and retrieval of document chunks.
- **OpenAI Embeddings**: Utilized for generating text embeddings, enhancing the semantic understanding and retrieval accuracy of the chatbot.

References

https://python.langchain.com/docs/use_cases/chatbots/

https://www.topcoder.com/thrive/articles/creating-a-spark-cluster-in-databricks

https://www.youtube.com/watch?v=LhnCsygAvzY

https://www.youtube.com/watch?v=3yPBVii7Ct0

https://www.anaconda.com/blog/how-to-build-a-retrieval-augmented-generation-chatbot

https://codebasics.io/resources/langchain-crash-course

https://medium.com/the-data-perspectives/custom-prompts-for-langchain-chains-a780b490c199