HUSH PROTO - Convolnsight

s

Design Document:

1. Introduction:

* Purpose: ConvoInsight aims to empower users by providing deep insights and facilitating easy interactions with their personal data from Instagram, LinkedIn, and Twitter. By utilizing state-of-the-art embeddings and a conversational interface, it serves as a personalized AI chatbot that decodes and presents vital information from the user's connections and messages across these platforms.
* Background: In today's digital age, managing information from platforms like Instagram, LinkedIn, and Twitter is daunting. ConvoInsight was conceived to simplify this, transforming users' vast data from these networks into intuitive, conversational insights, empowering individuals to understand and leverage their online interactions seamlessly.

---

2. Data Overview:

* Instagram Data:JSON format detailing personal connections, interactions, followers,following, messages etc.
* LinkedIn Data:CSV format detailing with connections, endorsements, messages and invitations etc.
* Twitter Data: JSON format detailing with followers , following and direct messages.

---

3. System Architecture & Flow:

- Data Ingestion:

* Parsing Data: Modules to parse JSON (Instagram, Twitter) and CSV (LinkedIn) data formats.

- Data Embedding:

* OpenAI Embedding: Use OpenAI's model to transform personal data into vector embeddings.
* Pinecone Integration: Store these embeddings into Pinecone, a vector database, for efficient retrieval.
* DocChunks: DockChunks are created using langchain text splitter which creates documents for which embeddings will be generated and stored in vector store.

- Query Mechanism:

* ConversationalRetrievalQAChain: ConversationalRetrievalQAChain mechanism to obtain information from the source documents which are stored as the vector embeddings in pinecone vector database and provide response based on the context
* LangChain: Ensure language translation and natural language querying capabilities are optimized.

- Response Generation:

* Matching & Retrieval: Use Pinecone's vector search capabilities to fetch relevant data vectors.
* Decoding: Convert these vectors back to human-readable information using OpenAI's mechanisms.

---

4. Chatbot Interface:

* User Input: Text box functionality.
* Display: Chat history with user and bot interactions.
* Error Handling: Responses for when the bot can't understand or find a match for the query.

---

5. Security & Privacy:

* Data Encryption: Ensure user data, both raw and in vector format, is securely stored. The vector database that is pinecone stored data in vector format which secures the data and the api keys that provide access are stored as environment variables which is secured.
* Access Control: Only authenticated users can query their personal data.

---

6. Performance & Scalability:

* Response Time: Aim for real-time or near-real-time responses for a seamless user experience.
* Pinecone Scalability: As Pinecone is designed for scalability, ensure integration methods allow for growth.

---

7. User Experience & Frontend:

* Design: A clean, intuitive chat interface resembling ChatGPT.
* Interactivity: Features such as suggesting next possible questions, or visualizing certain data insights when appropriate.

---

8. Milestones & Development Phases:

* Data Parsing & Integration: Complete data ingestion modules.
* Embedding & Storage: Integrate OpenAI and Pinecone systems.
* Chatbot Interface Development: Design and implement the frontend.
* Testing & Iteration: Run various testing phases and iterate based on feedback.

---