POC for Internal LLM

Overview

This document presents a straightforward Proof of Concept (PoC) for a question-answering (QA) system designed to work with sensitive NDA Documentation. The goal is to set up a system that can answer questions directly from these documents without sending data outside our local network.

System Details

Language Model and Text Processing:

We're using a language model called "fastchat-t5-3b-v1.0" from Hugging Face, a popular choice for understanding and generating text.

A tokenizer, which is like a prep tool for our text, gets everything ready for the model to process.

Creating the Pipeline:

We've built a special pipeline that uses both the language model and the tokenizer. This is where the magic happens – turning text into something our model can work with.

The key here is that everything happens locally, keeping our data secure.

Handling the Document:

We start by loading a sample PDF using something called PyPDFLoader.

Then, we break this document into smaller parts, making it easier to manage and process.

Generating Embeddings:

Next, we use the instructor embedding model from Hugging Face to transform these text chunks into embeddings, which are like numerical representations of our text.

Think of embeddings as a way to turn text into a form that our system can easily understand and compare.

Storing and Managing Data:

All these embeddings are stored in a Chroma Vector store – it's like a library of our processed text.

We keep all this information on our local disk to ensure it stays private.

Bringing It All Together:

We use the Langchain framework to connect our text processing pipeline with our vector store.

The end result is a QA chain that lets us ask questions about the NDA Documentation and get answers based on the document's content, without relying on external help.