

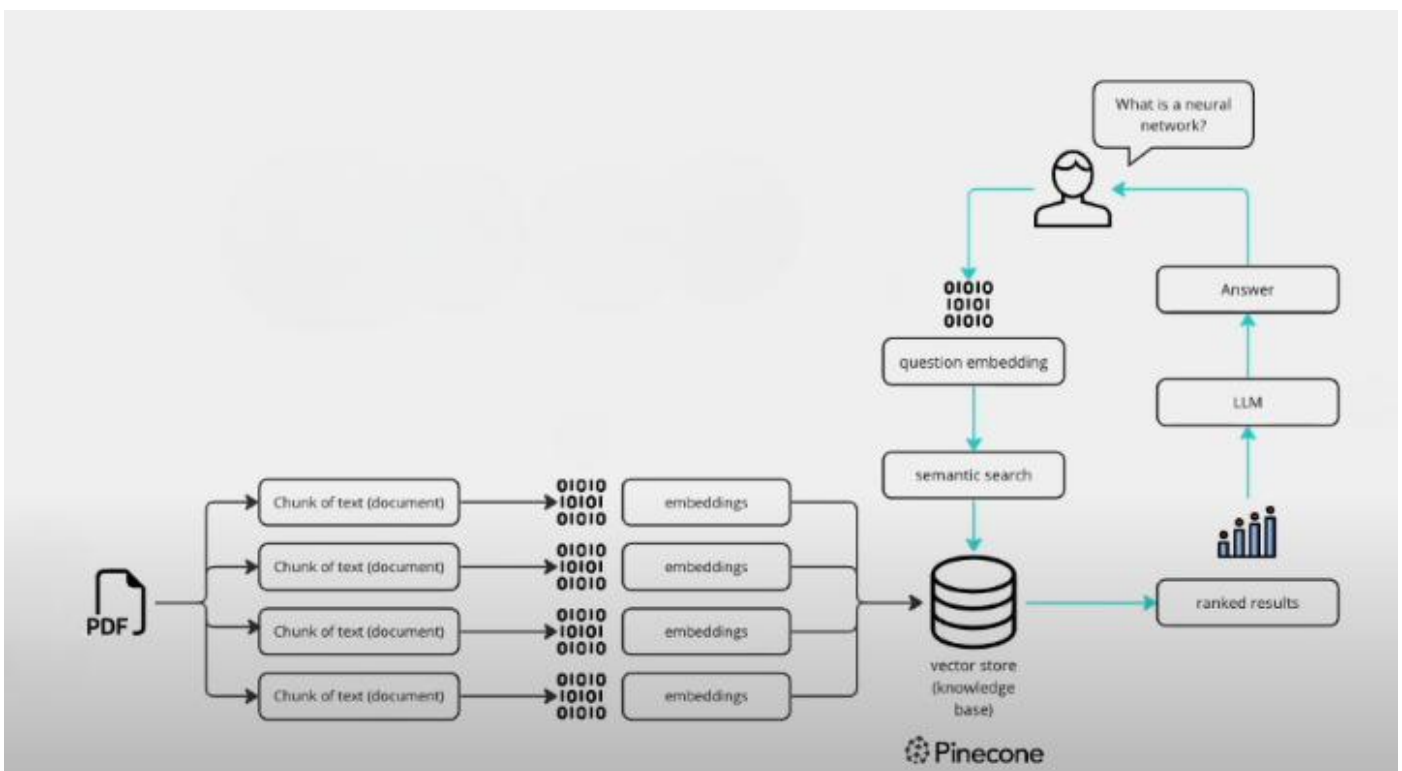
QA FROM PDF FILE USING LANGCHAIN AND PINECONE WITH HUGGINGFACE MODEL:

LangChain is a framework for building question answering systems. It provides a number of features that make it well-suited for this task, such as the ability to handle different types of documents, the ability to use different embeddings models, and the ability to query the index with multiple questions.

Pinecone is a vector database that can be used to store and query text embeddings. This makes it a good choice for storing the embeddings of the PDF documents that we will be using for QA.

HuggingFace Transformers is a library that provides a number of pre-trained language models that can be used for question answering. In this case, we will be using the bert-base-uncased model, which is a large language model that has been trained on a massive dataset of text and code

Architecture



Importing Python Required Libraries

```
from langchain.document_loaders import PyPDFLoader, OnlinePDFLoader
from langchain.text_splitter import RecursiveCharacterTextSplitter
from langchain.embeddings import HuggingFaceEmbeddings
from langchain.vectorstores import Pinecone
from sentence_transformers import SentenceTransformer
from langchain.chains.question_answering import load_qa_chain
import pinecone
import os
```

LangChain - allows you to build dynamic, data-responsive applications.

PyPDF2 - allows you to manipulate and extract data from PDF files.

sentence transformers - provides pre-trained models for generating numerical representations (embeddings) of sentences or text passages.

Pinecone - Pinecone is a vector database with broad functionality. It's a managed, cloud-native vector database with a simple API.

Recursive Character Text Splitter - This text Splitter is the recommended one for generic text. It is parameterized by a list of characters.

HuggingFace Embeddings - allows us to embed a data set Since the embeddings capture the semantic meaning of the questions.

Load_QA_Chain - load_qa_chain provides the most generic interface for answering questions.

```
# initialize pinecone
pinecone.init(
    api_key=PINECONE_API_KEY, # find at app.pinecone.io
    environment=PINECONE_API_ENV # next to api key in console
)
index_name = "langchainpinecone1"

#Create Embeddings for Each of the Text Chunk
docsearch=Pinecone.from_texts([t.page_content for t in docs], embeddings, index_name=index_name)
```

Created the embeddings and stored in docsearch. Then run streamlit application.

Ask Your PDF

Upload your pdf



Drag and drop file here

Limit 200MB per file • PDF

Browse files



farmerbook.pdf 4.3MB



154

714

Ask Question about your PDF:

about Sprinkler irrigation

- The scheme includes both drip and sprinkler irrigation. However, sprinkler irrigation will be applicable only for those crops where drip irrigation is uneconomical.
- There will be a strong HRD input for the farmers, field functionaries and other stakeholders at different levels.
- Moreover, there will be publicity campaigns, seminars/workshops at extensive locations to develop skills and improve awareness among farmers about importance of water conservation and management.

Ask Question about your PDF:

Mechanization

First step in mechanization • Get good hands on training • Read manufacturer information • Give attention to maintenance • Understand do's and don'ts with respect to equipments and machinery used • Take utmost care in following safety tips given in the manufacture information booklet. Selection of farm machinery • Select based on holding size • Economic feasibility • Availability of skilled labour to operate • Workout the feasibility of hiring v/s owning