RAG_Rothman: Chapter 2 (b) Embedding the Data and Populating Deeplake vector store

Sunday 29th December, 2024 at 11:28

#Embedding-Based Retrieval with Activeloop and OpenAI

This second component of the RAG pipeline transforms the prepared data by the first component of the pipeline into embeddings and stores the vectors obtained in the vector store.

1 Installing the environment

[2]:

```
import deeplake
```

Mount a drive or implement the method that best fits your project to retrieve API tokens.

[15]:

```
#The OpenAI Key
import os
from dotenv import load_dotenv
import openai

# Load API Key
dotenv_path = 'D:/AdvancedR/knowbankedu/openai/.env'
load_dotenv(dotenv_path)
# OpenAI API Key
openai.api_key = os.getenv("OPENAI_API_KEY")
ACTIVELOOP_TOKEN = os.getenv('ACTIVELOOP_TOKEN')
```

2 Embedding and Storage: populating the vector store

2.1 Downloading and preparing the data

[16]:

```
# Define the path to the local file
file_path = r"D:\RAG_Rothman\Chapter02\llm.txt"

# Read the content of the file
try:
    with open(file_path, 'r', encoding='utf-8') as file:
        content = file.readlines()
except FileNotFoundError:
        print(f"Error: File not found at {file_path}")
        content = []

# Display the first 20 lines of the file for verification
if content:
    print("First 20 lines of the file:")
    for line in content[:20]:
        print(line.strip())
else:
    print("The file is empty or could not be read.")

source_text = "D:/RAG_Rothman/Chapter02/llm.txt"
```

```
First 20 lines of the file: which is actually 200 pages so not sure what is going on here
Exploration of space, planets, and moons "Space Exploration" redirects here. For
the company, see SpaceX . For broader coverage of this topic, see Exploration .
Buzz Aldrin taking a core sample of the Moon during the Apollo 11 mission Self-
portrait of Curiosity rover on Mars 's surface Part of a series on Spaceflight
History History of spaceflight Space Race Timeline of spaceflight Space probes
Lunar missions Mars missions Applications Communications Earth observation
Exploration Espionage Military Navigation Settlement Telescopes Tourism
Spacecraft Robotic spacecraft Satellite Space probe Cargo spacecraft Crewed
spacecraft Apollo Lunar Module Space capsules Space Shuttle Space stations
Spaceplanes Vostok Space launch Spaceport Launch pad Expendable and reusable
launch vehicles Escape velocity Non-rocket spacelaunch Spaceflight types Sub-
orbital Orbital Interplanetary Interstellar Intergalactic List of space
organizations Space agencies Space forces Companies Spaceflight portal v t e
Space exploration is the use of astronomy and space technology to explore outer
space . [ 1 ] While the exploration of space is currently carried out mainly by
astronomers with telescopes , its physical exploration is conducted both by
uncrewed robotic space probes and human spaceflight . Space exploration, like
its classical form astronomy , is one of the main sources for space science .
( January 2020 ) ( Learn how and when to remove this
message ) By the early 1970s, astronomers began to consider the possibility of
placing an infrared telescope above the obscuring effects of Earth's atmosphere.
```

Chunking the data

[17]:

```
with open(source_text, 'r') as f:
    text = f.read()

CHUNK_SIZE = 1000
chunked_text = [text[i:i+CHUNK_SIZE] for i in range(0,len(text), CHUNK_SIZE)]
```

3 Verify if vector store exists in Deeplake or create it

Here we define an embedding function, then create the dataset in deeplake with needed tensors and populate with embeddings.

We set the chunk size at 1000 in this example. Can set the chunk size depending on token limits (see ChatGPT help)

[39]:

```
import deeplake
from openai import OpenAI
from dotenv import load_dotenv

# Load API key from .env
load_dotenv()
client = OpenAI()

# Path to the dataset in Deep Lake
vector_store_path = "hub://zagamog/space_exploration_v1"

# Embedding function using OpenAI client
def embedding_function(texts, model="text-embedding-3-small"):
    if isinstance(texts, str):
        texts = [texts]
    texts = [t.replace("\n", " ") for t in texts]  # Replace newlines with spaces
    response = client.embeddings.create(input=texts, model=model)
    return [item.embedding for item in response.data]

# Create and populate dataset
def create_and_populate_dataset(path, source_text_path, chunk_size=1000):
    print(f"Creating dataset at {path}...")
    dataset = deeplake.empty(path, overwrite=True)
    dataset.create_tensor("embedding_tensor", htype="embedding")
    dataset.create_tensor("text", htype="text")
    dataset.create_tensor("metadata", htype="json")
    dataset.commit("Initialized dataset.")

    # Read and chunk the text
    print("Reading and chunking source text...")
    with open(source_text_path, 'r', encoding='utf-8') as f:
        text = f.read()
        chunks = [text[i:i + chunk_size] for i in range(0, len(text),␣
 ↪chunk_size)]
    print(f"Split text into {len(chunks)} chunks.")

    # Generate embeddings
    print("Generating embeddings...")
    embeddings = embedding_function(chunks)
    print("Embeddings generated.")
    # Populate the dataset
    print("Populating the dataset...")
    dataset["embedding_tensor"].extend(embeddings)
    dataset["text"].extend(chunks)
    dataset["metadata"].extend([{"source": source_text_path}] * len(chunks))
    dataset.commit("Dataset populated.")
    return dataset

# Source text
source_text_path = "D:/RAG_Rothman/Chapter02/llm.txt"

# Run
try:
    dataset = create_and_populate_dataset(vector_store_path, source_text_path)
    print("Dataset created and populated successfully.")
except Exception as e:
    print(f"An error occurred: {e}")
```

```
Creating dataset at hub://zagamog/space_exploration_v1...
Your Deep Lake dataset has been successfully created!


This dataset can be visualized in Jupyter Notebook by ds.visualize() or at
https://app.activeloop.ai/zagamog/space_exploration_v1
hub://zagamog/space_exploration_v1 loaded successfully.


Reading and chunking source text...
Split text into 1635 chunks.
Generating embeddings...
Embeddings generated.
Populating the dataset...


Dataset created and populated successfully.
```

Visualize

Online: https://app.activeloop.ai/datasets/mydatasets/

[40]:
```
# Print the summary of the Vector Store
print(vector_store.summary())
```

```
Dataset(path='hub://zagamog/space_exploration_v1', tensors=['embedding_tensor',
'id'])

   tensor       htype     shape    dtype  compression
  -------      -------   -------  -------  -------
```

```
 embedding_tensor  embedding   (0,)    float32   None
        id           text      (0,)      str     None
None
```

[41]:

```python
ds = deeplake.load(vector_store_path)
```

```
|51

This dataset can be visualized in Jupyter Notebook by ds.visualize() or at
https://app.activeloop.ai/zagamog/space_exploration_v1


-

hub://zagamog/space_exploration_v1 loaded successfully.
```

Dataset size

[42]:

```python
#Estimates the size in bytes of the dataset.
ds_size=ds.size_approx()
```

[43]:

```python
# Convert bytes to megabytes and limit to 5 decimal places
ds_size_mb = ds_size / 1048576
print(f"Dataset size in megabytes: {ds_size_mb:.5f} MB")

# Convert bytes to gigabytes and limit to 5 decimal places
ds_size_gb = ds_size / 1073741824
print(f"Dataset size in gigabytes: {ds_size_gb:.5f} GB")
```

```
Dataset size in megabytes: 55.31311 MB
Dataset size in gigabytes: 0.05402 GB
```