RAG_Rothman: Chapter 2 (c) Augmented Generation using Deeplake vector store

Sunday 29th December, 2024 at 22:21

#Embedding-Based Retrieval with Deep Lake and OpenAI

Copyright 2024 Denis Rothman

1 1. Installing the environment

First run the following cells and restart Google Colab session if prompted. Then run the notebook again cell by cell to explore the code.

[30]:

```
import deeplake
```

[31]:

```
#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 Retrieval Augmented Generation

2.0.1 Initiating the query process by indicating location of vector store

[32]:

```
vector_store_path = "hub://zagamog/space_exploration_v1"
```

[33]:

```
from deeplake.core.vectorstore.deeplake_vectorstore import VectorStore
import deeplake.util
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.
```

[34]:

```
vector_store = VectorStore(path=vector_store_path)
```

```
Deep Lake Dataset in hub://zagamog/space_exploration_v1 already exists, loading
from the storage
```

2.1 Input and Query Retrieval

2.2 Input

2.2.1 Retrieval query

[35]:

```
# Print all tensors in the dataset
print("Tensors in dataset:", list(ds.tensors.keys()))
```

```
Tensors in dataset: ['embedding_tensor', 'id', 'metadata', 'text']
```

[36]:

```
def embedding_function(texts, model="text-embedding-3-small"):
   if isinstance(texts, str):
       texts = [texts]
   texts = [t.replace("\n", " ") for t in texts]
   return [data.embedding for data in openai.embeddings.create(input = texts,␣
 ↪model=model).data]
```

[37]:

```
def get_user_prompt():
    # Request user input for the search prompt
    return input("Enter your search query: ")

def search_query(prompt):
    # Assuming `vector_store` and `embedding_function` are already defined
    search_results = vector_store.search(embedding_data=prompt,␣
 ↪embedding_function=embedding_function, embedding_tensor="embedding_tensor")
    return search_results

# Get the user's search query
#user_prompt = get_user_prompt()
# or enter prompt if it is in a queue
user_prompt="Tell me about space exploration on the Moon and Mars."

# Perform the search
search_results = search_query(user_prompt)

# Print the search results
print(search_results)
```

```
{'score': [0.5716322660446167, 0.5714114904403687, 0.5654380321502686,
0.561565637588501], 'id': ['chunk_79', 'chunk_121', 'chunk_119', 'chunk_42'],
'metadata': [{'source': 'Source URL:
https://en.wikipedia.org/wiki/Exploration_of_Mars'}, {'source': ''}, {'source':
''}, {'source': 'Source URL:
https://en.wikipedia.org/wiki/Exploration_of_Mars'}], 'text': ['udy of orbits to
land on Mars and return to Earth (High School level) Planetary Society Mars page
v t e Space exploration Benefits Future Topics Astronomy Deep space exploration
Space colonization Space research Spaceflight Human Uncrewed Exploration targets
Asteroids Comets Earth Moon Jupiter Mars Human mission Phobos Mercury Neptune
Pluto Saturn Uranus Venus History List of spaceflight records Timeline of Solar
System exploration Timeline of space exploration Space agencies CNSA CSA ESA
ISRO JAXA NASA Roscosmos UAESA Category Outline v t e Mars Outline of Mars
Geography Atmosphere Circulation Climate Dust devil tracks Methane Regions
Arabia Terra Cerberus (Mars) Cydonia Eridania Lake Iani Chaos Olympia Undae
Planum Australe Planum Boreum Quadrangles Sinus Meridiani Tempe Terra Terra
Cimmeria Terra Sabaea Tharsis Undae Ultimi Scopuli Vastitas Borealis Physical
features "Canals" ( list ) Canyons Catenae Chaos terrain Craters Fossae Gullies
Mensae Labyrinthi Mountains by height Observed', 'e choose to go to the Moon "
Apollo 8 Genesis reading Apollo 15 postal covers incident Space Mirror Memorial
The Astronaut Monument Lunar sample displays Moon rocks stolen or missing U.S.
Astronaut Hall of Fame Space program on U.S. stamps Apollo 17 Moon mice Moon
tree Other primates in space NASA International Space Apps Challenge Astronauts
Day National Astronaut Day Nikon NASA F4 Category Commons v t e Spacecraft
missions to Mars List of missions to Mars List of Mars orbiters List of
artificial objects on Mars Active Flybys Psyche ‡ (2023, flyby in 2026) Hera ‡
(2024, flyby in 2025) Europa Clipper ‡ (2024, flyby in 2025) Orbiters 2001 Mars
Odyssey Mars Express Mars Reconnaissance Orbiter timeline MAVEN ExoMars Trace
Gas Orbiter Hope Tianwen-1 orbiter Rovers Curiosity Mars Science Laboratory
timeline Perseverance Mars 2020 timeline Past Flybys Mars 1 † Mariner 4 Zond 2 †
Mariner 6 and 7 Mars 6 Mars 7 Rosetta ‡ Dawn ‡ Mars Cube One Orbiters Mars 2
Mars 3 Mariner 9 Mars 4 † Mars 5 Viki', 'Mars Exploration Mars Exploration Rover
Current Living With a Star Lunar Precursor Robotic Program Earth Observing
System Great Observatories program Explorers Voyager Discovery New Frontiers
Solar Terrestrial Probes Commercial Lunar Payload Services SIMPLEx Individual
featured missions (human and robotic) Past Apollo 11 COBE Mercury 3 Mercury-
Atlas 6 Magellan Pioneer 10 Pioneer 11 Galileo timeline GALEX GRAIL WMAP Space
Shuttle Spitzer Space Telescope Sojourner rover Spirit rover LADEE MESSENGER
Aquarius Cassini Dawn Kepler space telescope Opportunity rover timeline observed
RHESSI InSight Ingenuity helicopter flights Currently operating Mars
Reconnaissance Orbiter 2001 Mars Odyssey New Horizons International Space
Station Hubble Space Telescope Swift THEMIS Mars Exploration Rover Curiosity
rover timeline GOES 14 Lunar Reconnaissance Orbiter GOES 15 Van Allen Probes
Solar Dynamics Observatory Juno Mars Science Laboratory timeline NuSTAR Voyager
1 Voyager 2 WISE MAVEN MMS OSIRIS-REx Tr', 'ome of which are listed below. NASA
[ edit ] Human exploration by the United States was identified as a long-term
goal in the Vision for Space Exploration announced in 2004 by then US President
George W. Bush . [ 145 ] The planned Orion spacecraft would be used to send a
human expedition to Earth\'s moon by 2020 as a stepping stone to a Mars
expedition. On September 28, 2007, NASA administrator Michael D. Griffin stated
that NASA aims to put a person on Mars by 2037. [ 146 ] On December 2, 2014,
NASA\'s Advanced Human Exploration Systems and Operations Mission Director Jason
Crusan and Deputy Associate Administrator for Programs James Reuthner announced
tentative support for the Boeing "Affordable Mars Mission Design" including
radiation shielding, centrifugal artificial gravity, in-transit consumable
resupply, and a lander which can return. [ 147 ] [ 148 ] Reuthner suggested that
if adequate funding was forthcoming, the proposed mission would be expected in
the early 2030s. [ 149 ] Jour']}
```

[38]:

```python
print(user_prompt)
```

```
Tell me about space exploration on the Moon and Mars.
```

[39]:

```python
# Function to wrap text to a specified width
def wrap_text(text, width=80):
    lines = []
    while len(text) > width:
        split_index = text.rfind(' ', 0, width)
        if split_index == -1:
            split_index = width
        lines.append(text[:split_index])
        text = text[split_index:].strip()
    lines.append(text)
    return '\n'.join(lines)
```

[40]:

```python
import textwrap

# Assuming the search results are ordered with the top result first
top_score = search_results['score'][0]
top_text = search_results['text'][0].strip()
top_metadata = search_results['metadata'][0]['source']

# Print the top search result
print("Top Search Result:")
print(f"Score: {top_score}")
print(f"Source: {top_metadata}")
print("Text:")
print(wrap_text(top_text))
```

```
Top Search Result:
Score: 0.5716322660446167
Source: Source URL: https://en.wikipedia.org/wiki/Exploration_of_Mars
Text:
udy of orbits to land on Mars and return to Earth (High School level) Planetary
Society Mars page v t e Space exploration Benefits Future Topics Astronomy Deep
space exploration Space colonization Space research Spaceflight Human Uncrewed
Exploration targets Asteroids Comets Earth Moon Jupiter Mars Human mission
```

```
Phobos Mercury Neptune Pluto Saturn Uranus Venus History List of spaceflight
records Timeline of Solar System exploration Timeline of space exploration
Space agencies CNSA CSA ESA ISRO JAXA NASA Roscosmos UAESA Category Outline v t
e Mars Outline of Mars Geography Atmosphere Circulation Climate Dust devil
tracks Methane Regions Arabia Terra Cerberus (Mars) Cydonia Eridania Lake Iani
Chaos Olympia Undae Planum Australe Planum Boreum Quadrangles Sinus Meridiani
Tempe Terra Terra Cimmeria Terra Sabaea Tharsis Undae Ultimi Scopuli Vastitas
Borealis Physical features "Canals" ( list ) Canyons Catenae Chaos terrain
Craters Fossae Gullies Mensae Labyrinthi Mountains by height Observed
```

2.3 Augmented Input

[41]:

```python
augmented_input=user_prompt+" "+top_text
```

[42]:

```python
print(augmented_input)
```

```
Tell me about space exploration on the Moon and Mars. udy of orbits to land on
Mars and return to Earth (High School level) Planetary Society Mars page v t e
Space exploration Benefits Future Topics Astronomy Deep space exploration Space
colonization Space research Spaceflight Human Uncrewed Exploration targets
Asteroids Comets Earth Moon Jupiter Mars Human mission Phobos Mercury Neptune
Pluto Saturn Uranus Venus History List of spaceflight records Timeline of Solar
System exploration Timeline of space exploration Space agencies CNSA CSA ESA
ISRO JAXA NASA Roscosmos UAESA Category Outline v t e Mars Outline of Mars
Geography Atmosphere Circulation Climate Dust devil tracks Methane Regions
Arabia Terra Cerberus (Mars) Cydonia Eridania Lake Iani Chaos Olympia Undae
Planum Australe Planum Boreum Quadrangles Sinus Meridiani Tempe Terra Terra
Cimmeria Terra Sabaea Tharsis Undae Ultimi Scopuli Vastitas Borealis Physical
features "Canals" ( list ) Canyons Catenae Chaos terrain Craters Fossae Gullies
Mensae Labyrinthi Mountains by height Observed
```

3 Generation and output

[43]:

```python
import openai
from openai import OpenAI
import time

client = OpenAI()
gpt_model="gpt-4o"
start_time = time.time()  # Start timing before the request

def call_gpt4_with_full_text(itext):
    # Join all lines to form a single string
    text_input = '\n'.join(itext)
    prompt = f"Please summarize or elaborate on the following content:
 ↪\n{text_input}"
```

```
    try:
        response = client.chat.completions.create(
            model=gpt_model,
            messages=[
                {"role": "system", "content": "You are a space exploration␣
↪expert."},
                {"role": "assistant", "content": "You can read the input and␣
↪answer in detail."},
                {"role": "user", "content": prompt}
            ],
            temperature=0.1  # Fine-tune parameters as needed
        )
        return response.choices[0].message.content
    except Exception as e:
        return str(e)

gpt4_response = call_gpt4_with_full_text(augmented_input)

response_time = time.time() - start_time  # Measure response time
print(f"Response Time: {response_time:.2f} seconds")  # Print response time

print(gpt_model, "Response:", gpt4_response)
```

```
Response Time: 8.85 seconds
gpt-4o Response: Space exploration on the Moon and Mars has been a significant
focus of scientific and technological efforts, driven by the desire to
understand more about our solar system and the potential for human colonization
beyond Earth.

### Moon Exploration
The Moon has been a primary target for space exploration due to its proximity to
Earth. Key milestones include:
- **Apollo Missions**: NASA's Apollo program successfully landed humans on the
Moon between 1969 and 1972, providing valuable data on lunar geology and the
potential for future human habitation.
- **Robotic Missions**: Various countries, including China, India, and Russia,
have sent robotic missions to the Moon to study its surface, search for water
ice, and test technologies for future human missions.

### Mars Exploration
Mars has been a focal point for exploration due to its similarities to Earth and
the potential for past or present life. Key aspects include:
- **Robotic Missions**: NASA's rovers, such as Spirit, Opportunity, Curiosity,
and Perseverance, have explored Mars' surface, analyzing rocks, soil, and the
atmosphere to understand the planet's history and habitability.
- **Orbiters**: Satellites like the Mars Reconnaissance Orbiter and the European
Space Agency's Mars Express have provided detailed maps of the Martian surface
```

and climate data.
- **Human Missions**: Plans for human missions to Mars are being developed by NASA and private companies like SpaceX, focusing on the challenges of long-duration space travel and sustainable living on Mars.

Study of Orbits
Understanding the orbits necessary for landing on Mars and returning to Earth is crucial for mission planning. This involves:
- **Hohmann Transfer Orbits**: A common method for traveling between planets, minimizing fuel use by taking advantage of the relative positions of Earth and Mars.
- **Aerobraking**: Using a planet's atmosphere to slow down a spacecraft, reducing the need for fuel and allowing for more efficient landings.

Benefits of Space Exploration
Space exploration offers numerous benefits, including:
- **Scientific Knowledge**: Expanding our understanding of the solar system and the potential for life beyond Earth.
- **Technological Advancements**: Innovations developed for space missions often have applications on Earth, such as improvements in materials, communications, and robotics.
- **Inspiration and Education**: Space exploration inspires future generations and promotes interest in STEM fields.

Future Topics and Targets
Future exploration efforts will continue to focus on:
- **Deep Space Exploration**: Missions to asteroids, comets, and other planets like Jupiter, Saturn, and beyond.
- **Space Colonization**: Developing the technology and infrastructure needed for human settlements on the Moon and Mars.
- **Space Agencies**: Collaboration among international space agencies, including NASA, ESA, CNSA, ISRO, JAXA, and others, to achieve common goals in space exploration.

Mars Geography and Features
Mars is characterized by diverse geographical features, including:
- **Regions**: Areas like Arabia Terra, Cerberus, and Cydonia, each with unique geological characteristics.
- **Physical Features**: Canyons, craters, and mountains, such as Olympus Mons, the tallest volcano in the solar system.
- **Atmosphere and Climate**: Mars has a thin atmosphere, with dust storms and evidence of past water flow, influencing its climate and potential habitability.

Overall, space exploration of the Moon and Mars continues to be a dynamic and evolving field, with ongoing missions and future plans aimed at unlocking the mysteries of our neighboring celestial bodies.

3.0.1 Formatted response

[46]:

```python
import textwrap
import re
from IPython.display import display, Markdown, HTML
import markdown

def print_formatted_response(response):
    # Check for markdown by looking for patterns like headers, bold, lists, etc.
    markdown_patterns = [
        r"^#+\s",           # Headers
        r"^\*+",            # Bullet points
        r"\*\*",            # Bold
        r"_",               # Italics
        r"\[.+\]\(.+\)",    # Links
        r"-\s",             # Dashes used for lists
        r"\`\`\`"           # Code blocks
    ]

    # If any pattern matches, assume the response is in markdown
    if any(re.search(pattern, response, re.MULTILINE) for pattern in␣
 ↪markdown_patterns):
        # Markdown detected, convert to HTML for nicer display
        html_output = markdown.markdown(response)
        display(HTML(html_output))  # Use display(HTML()) to render HTML in Colab
    else:
        # No markdown detected, wrap and print as plain text
        wrapper = textwrap.TextWrapper(width=80)
        wrapped_text = wrapper.fill(text=response)

        print("Text Response:")
        print("--------------------")
        print(wrapped_text)
        print("--------------------\n")

print_formatted_response(gpt4_response)
```

```
<IPython.core.display.HTML object>
```

4 Evaluating the output with Cosine Similarity

with initial user prompt

[47]:

```python
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

def calculate_cosine_similarity(text1, text2):
    vectorizer = TfidfVectorizer()
    tfidf = vectorizer.fit_transform([text1, text2])
    similarity = cosine_similarity(tfidf[0:1], tfidf[1:2])
    return similarity[0][0]

similarity_score = calculate_cosine_similarity(user_prompt, gpt4_response)

print(f"Cosine Similarity Score: {similarity_score:.3f}")
```

```
Cosine Similarity Score: 0.470
```

with augmented user prompt

[48]:

```
similarity_score = calculate_cosine_similarity(augmented_input, gpt4_response)

print(f"Cosine Similarity Score: {similarity_score:.3f}")
```

```
Cosine Similarity Score: 0.418
```

[50]:

```
from sentence_transformers import SentenceTransformer
model = SentenceTransformer('all-MiniLM-L6-v2')
```

```
C:\ProgramData\anaconda3\envs\rothmanwa\Lib\site-packages\tqdm\auto.py:21:
TqdmWarning: IProgress not found. Please update jupyter and ipywidgets. See
https://ipywidgets.readthedocs.io/en/stable/user_install.html
  from .autonotebook import tqdm as notebook_tqdm
modules.json: 100%|| 349/349 [00:00<?, ?B/s]
config_sentence_transformers.json: 100%|| 116/116 [00:00<?, ?B/s]
README.md: 100%|| 10.7k/10.7k [00:00<?, ?B/s]
sentence_bert_config.json: 100%|| 53.0/53.0 [00:00<?, ?B/s]
config.json: 100%|| 612/612 [00:00<?, ?B/s]
model.safetensors: 100%|| 90.9M/90.9M [00:01<00:00, 55.1MB/s]
tokenizer_config.json: 100%|| 350/350 [00:00<00:00, 348kB/s]
vocab.txt: 100%|| 232k/232k [00:00<00:00, 3.93MB/s]
tokenizer.json: 100%|| 466k/466k [00:00<00:00, 12.4MB/s]
special_tokens_map.json: 100%|| 112/112 [00:00<?, ?B/s]
1_Pooling/config.json: 100%|| 190/190 [00:00<?, ?B/s]
```

[51]:

```
def calculate_cosine_similarity_with_embeddings(text1, text2):
    embeddings1 = model.encode(text1)
    embeddings2 = model.encode(text2)
    similarity = cosine_similarity([embeddings1], [embeddings2])
    return similarity[0][0]


similarity_score = calculate_cosine_similarity_with_embeddings(augmented_input,␣
 ↪gpt4_response)
print(f"Cosine Similarity Score: {similarity_score:.3f}")
```

```
Cosine Similarity Score: 0.637
```

```
[ ]:
```