SUPERIOR UNIVERSITY

Programming For Artificial Intelligence (Lab)

Assignment - 12

Name:

Ali Maqsood.

Roll no:

SU92-BSAIM-F23-050.

Department:

Software Engineering Department.

Program:

Artificial Intelligence.

Section:

BSAI-4A

Question # 1:

Then create your own QnA Bot from the topic you selected in Lab 10, using the same pipeline

Note:

Please note that the file by the name of model.ipynb shows all the EDA on the dataset and flask uses the code form model.py to import out model.

model.py:

```
import pandas as pd
import numpy as np
from sentence_transformers import SentenceTransformer
import faiss

dataset=pd.read_csv("spacenews.csv")
news_dataset=dataset[["title","author","content","date"]]
news_dataset.dropna(inplace=True)
import re
def clean_text(text):
    text=text.lower()
    text=re.sub(r'[^a-zA-z0-9\s]',' ',text)
    text=re.sub(r'\s+',' ',text)
    return text
news_dataset["clean_content"]=news_dataset["content"].apply(clean_text)
model=SentenceTransformer('all-MiniLM-L6-v2')
# embeddings=model.encode(news_dataset["clean_content"].values)
# embeddings=np.array(embeddings)
# np.save("content_embeddings.npy",embeddings)
```

```
embeddings=np.load("content_embeddings.npy")
dimension=embeddings.shape[1]
faiss_index=faiss.IndexFlatL2(dimension)
faiss_index.add(embeddings)
# faiss.write_index(faiss_index,"content_faiss_index.index")
def get_similar_news(query,model=model,faiss_index=faiss_index,count=3):
    query_embedding=model.encode([query])
    distance,indices=faiss_index.search(query_embedding,count)
    results=[]
    for i in range(count):
        idx = indices[0][i]
        news_item = {
            "title": news_dataset["title"].iloc[idx],
            "author": news_dataset["author"].iloc[idx],
            "date": news_dataset["date"].iloc[idx],
            "content": news_dataset["content"].iloc[idx]
        }
        results.append(news_item)
    return results
```

app.py:

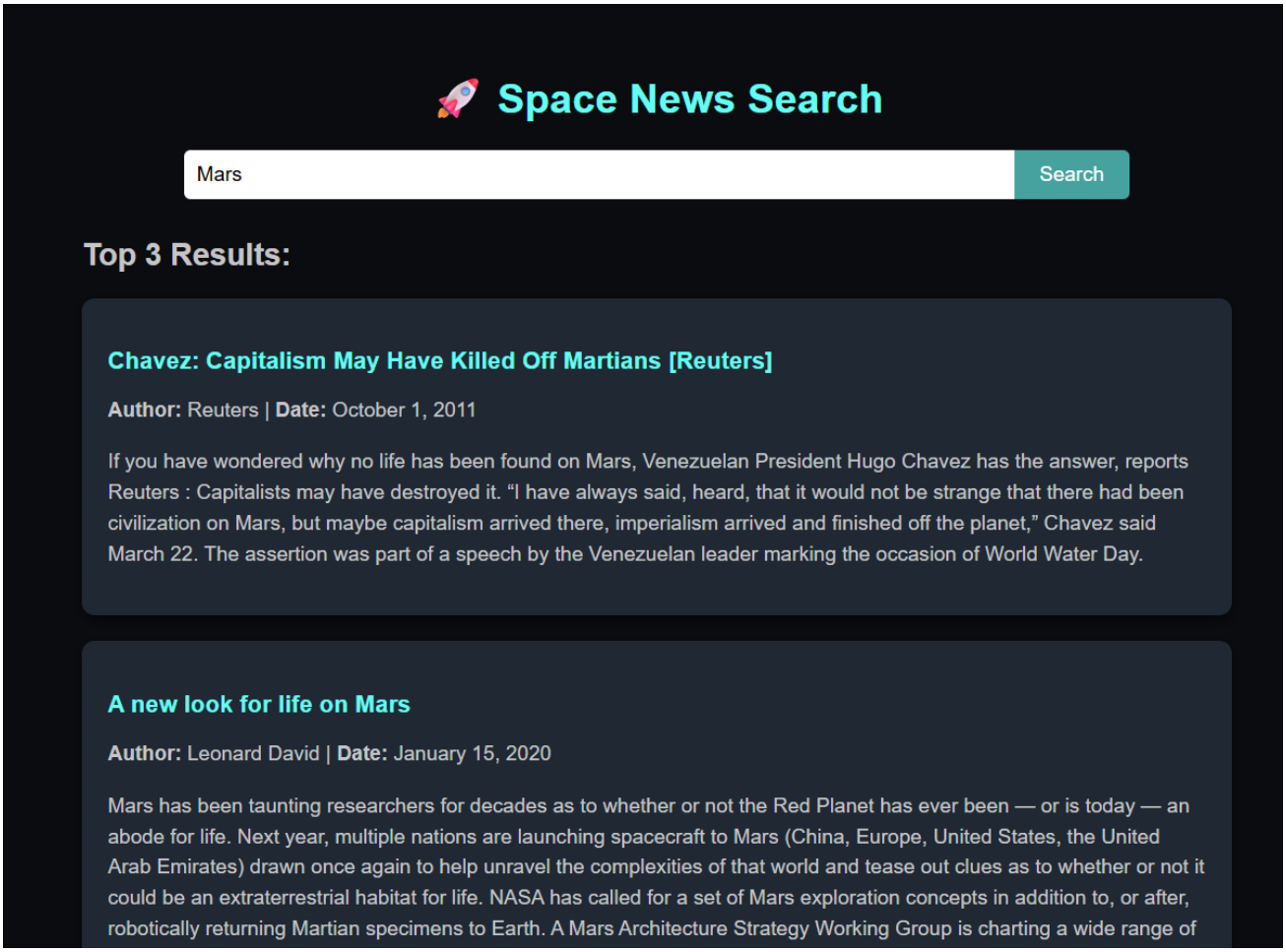
```
from flask import Flask, render_template, request
from model import get_similar_news

app = Flask(__name__)
```

```
@app.route("/", methods=["GET", "POST"])
def home():
    results=None
    query=""
    message=""
    if request.method=="POST":
        query=request.form.get("query")
        if query:
            results=get_similar_news(query)
            if not results:
                message="No news found for your query. Please try different keywords."
    return
render_template("index.html",results=results,query=query,message=message)


if __name__ == "__main__":
    app.run(debug=True)
```

Output:



Documentation:

This project is a Space News Search Web Application built using Flask for the backend and HTML/CSS for the frontend. It allows users to enter queries related to space topics, such as satellites, Mars, or space missions, and retrieves the most relevant news articles from a dataset using semantic search. The search functionality is powered by a SentenceTransformer model to generate embeddings and FAISS for efficient similarity search. The app displays the title, author, date, and content of the top 5 relevant news articles in a clean and user-friendly interface. All HTML, CSS used in this project were written by **ChatGPT**.