**Deployment code**-

import streamlit as st

import json

import numpy as np

import faiss

import spacy

import google.generativeai as genai

# ------------------- Configuration -------------------

PDF\_PROCESSED\_TEXT\_FILE = r"E:\\AI-EDUCATION\\processed\_science\_data.json"

EMBEDDINGS\_FILE = r"E:\\AI-EDUCATION\\spaCy\_embeddings.json"

VECTOR\_DB\_PATH = r"E:\\AI-EDUCATION\\vector\_database.index"

LOGO\_PATH = "E:/AI-EDUCATION/AiSPRY logo.jpeg"

# ------------------- UI: Logo and Title -------------------

st.image(LOGO\_PATH, width=250)

st.title("📘 Class 10 Science Chatbot (NCERT-based)")

# ------------------- Load Data -------------------

@st.cache\_resource

def load\_resources():

nlp\_model = spacy.load("en\_core\_web\_lg")

with open(PDF\_PROCESSED\_TEXT\_FILE, "r", encoding="utf-8") as f:

text\_data = json.load(f)

with open(EMBEDDINGS\_FILE, "r") as f:

embeddings = json.load(f)

faiss\_index = faiss.read\_index(VECTOR\_DB\_PATH)

chapter\_names = list(embeddings.keys())

return nlp\_model, text\_data, embeddings, faiss\_index, chapter\_names

nlp, processed\_text, embeddings, index, chapter\_names = load\_resources()

# ------------------- Gemini Setup -------------------

genai.configure(api\_key="MY API KEY ")

model = genai.GenerativeModel('models/gemini-1.5-flash')

# ------------------- Embedding & Search -------------------

def get\_embedding(text):

return nlp(text).vector

def search\_faiss(query, top\_k=3):

query\_vector = np.array([get\_embedding(query)], dtype='float32')

distances, indices = index.search(query\_vector, top\_k)

return [chapter\_names[i] for i in indices[0]]

def query\_gemini\_with\_context(query, top\_chapters):

context = "\n\n".join([f"{ch}:\n{processed\_text[ch]}" for ch in top\_chapters])

prompt = f"""You are a helpful AI tutor for Class 10 Science students.

Use the following NCERT content to answer the question accurately and simply.

Context:

{context}

Question: {query}

Answer in simple terms as per NCERT syllabus:"""

response = model.generate\_content(prompt)

return response.text

# ------------------- Streamlit UI -------------------

user\_query = st.text\_input("❓ Ask your NCERT Science question:", placeholder="e.g., What is photosynthesis?")

if user\_query:

with st.spinner("Searching and generating answer..."):

top\_chaps = search\_faiss(user\_query)

answer = query\_gemini\_with\_context(user\_query, top\_chaps)

st.markdown("---")

st.subheader("💡 Answer")

st.write(answer)