```
!pip install --upgrade google-generativeai
!pip install sentence-transformers faiss-cpu streamlit pyngrok
```

```
Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lil
Collecting nvidia-nvjitlink-cu12==12.4.127 (from torch>=1.11.0->sentence-t
  Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64
Requirement already satisfied: triton==3.2.0 in /usr/local/lib/python3.11/
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3
Requirement already satisfied: tokenizers<0.22,>=0.21 in /usr/local/lib/py
Requirement already satisfied: safetensors>=0.4.3 in /usr/local/lib/python
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.11/
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/pyth
Requirement already satisfied: smmap<6,>=3.0.1 in /usr/local/lib/python3.1
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.1
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.11/
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /us
Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/pytho
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.11
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-
Downloading faiss_cpu-1.11.0.post1-cp311-cp311-manylinux_2_27_x86_64.manyl
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 31.3/31.3 MB 57.3 MB/s eta 0:0
Downloading streamlit-1.46.1-py3-none-any.whl (10.1 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 10.1/10.1 MB 106.3 MB/s eta 0:
Downloading pyngrok-7.2.12-py3-none-any.whl (26 kB)
Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.9/6.9 MB 101.7 MB/s eta 0:00
Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 363.4/363.4 MB 1.3 MB/s eta 0:
Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 13.8/13.8 MB 72.3 MB/s eta 0:0
Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 24.6/24.6 MB 55.4 MB/s eta 0:0
Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_6
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 883.7/883.7 kB 45.4 MB/s eta 0
Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl (
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 664.8/664.8 MB 2.9 MB/s eta 0:
Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl (
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 211.5/211.5 MB 5.6 MB/s eta 0:
Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.wh
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 56.3/56.3 MB 11.6 MB/s eta 0:0
Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.wh
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 127.9/127.9 MB 7.5 MB/s eta 0:
Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 207.5/207.5 MB 5.9 MB/s eta 0:
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.wl
                                                               eta 0:0
Do                                                            )
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 79.1/79.1 kB 6.9 MB/s eta 0:00
Installing collected packages: watchdog, pyngrok, nvidia-nvjitlink-cu12, n
  Attempting uninstall: nvidia-nvjitlink-cu12
```

What can I help you build?

```
    Found existing installation: nvidia-nvjitlink-cu12 12.5.82
    Uninstalling nvidia-nvjitlink-cu12-12.5.82:
      Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82
  Attempting uninstall: nvidia-curand-cu12
    Found existing installation: nvidia-curand-cu12 10.3.6.82
    Uninstalling nvidia-curand-cu12-10.3.6.82:
      Successfully uninstalled nvidia-curand-cu12-10.3.6.82
  Attempting uninstall: nvidia-cufft-cu12
    Found existing installation: nvidia-cufft-cu12 11.2.3.61
    Uninstalling nvidia-cufft-cu12-11.2.3.61:
```

```
import os
os.environ["GOOGLE_API_KEY"] = "AIzaSyB1fG6M9oFP1ndTg-76AN-iW4U_t2JWuW8"
```

```
code = '''
import os
import google.generativeai as genai
from sentence_transformers import SentenceTransformer
import faiss
import streamlit as st

# Configure Gemini
genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))
model = genai.GenerativeModel("gemini-pro")

# Sample documents
docs = [
    "Python is a versatile programming language.",
    "FAISS enables efficient similarity search over vectors.",
    "Streamlit is used for building web apps in Python quickly.",
    "Gemini is a large language model from Google for smart text generation."
]

# Embedding and indexing
embed_model = SentenceTransformer("all-MiniLM-L6-v2")
doc_embeddings = embed_model.encode(docs)

dimension = doc_embeddings.shape[1]
index = faiss.IndexFlatL2(dimension)
index.add(doc_embeddings)

# Retriever
def retrieve_top_k(query, k=2):
    query_vec = embed_model.encode([query])
    distances, indices = index.search(query_vec, k)
    return [docs[i] for i in indices[0]]
```

```
# Generator
def generate_answer(query, context):
    prompt = f"Context:\\n{context}\\n\\nQuestion: {query}\\nAnswer:"
    response = model.generate_content(prompt)
    return response.text

# UI
st.title("RAG Q&A Chatbot (Gemini)")
query = st.text_input("Ask your question:")

if query:
    context = "\\n".join(retrieve_top_k(query))
    answer = generate_answer(query, context)

    st.subheader("Answer")
    st.write(answer)

    st.subheader("Context")
    st.write(context)
'''

with open("rag_gemini_chatbot.py", "w") as f:
    f.write(code)

print("Chatbot code file created: rag_gemini_chatbot.py")
```

```
✅ Chatbot code file created: rag_gemini_chatbot.py
```

```
!ngrok config add-authtoken "2zvNQ6yvtZxkJivtLC5e3vbxbEd_3J5wMRFaxQJL6KWZZPJYz"
```

```
Authtoken saved to configuration file: /root/.config/ngrok/ngrok.yml
```

```
from pyngrok import ngrok
```

```
public_url = ngrok.connect("http://localhost:8501")
print("🔗 Your chatbot is live at:", public_url)

!streamlit run rag_gemini_chatbot.py --server.enableCORS false --server.enableXsr
```

```
...
/usr/local/lib/python3.11/dist-packages/google/ai/generativelanguage_v1b
/generative_service/client.py:835 in generate_content

   832 │   │   self._validate_universe_domain()
```

```
  │   │
  │   │   # Send the request.
❱ │   │   response = rpc(
  │   │   │   request,
  │   │   │   retry=retry,
  │   │   │   timeout=timeout,

/usr/local/lib/python3.11/dist-packages/google/api_core/gapic_v1/method.
__call__

  │   │   if self._compression is not None:
  │   │   │   kwargs["compression"] = compression
  │   │
❱ │   │   return wrapped_func(*args, **kwargs)


  def wrap_method(

/usr/local/lib/python3.11/dist-packages/google/api_core/retry/retry_unar
retry_wrapped_func

  │   │   │   sleep_generator = exponential_sleep_generator(
  │   │   │   │   self._initial, self._maximum, multiplier=self._mul
  │   │   │   )
❱ │   │   │   return retry_target(
  │   │   │   │   target,
  │   │   │   │   self._predicate,
  │   │   │   │   sleep_generator,

/usr/local/lib/python3.11/dist-packages/google/api_core/retry/retry_unar
retry_target

  │   │   # This function explicitly must deal with broad exceptions
  │   │   except Exception as exc:
  │   │   │   # defer to shared logic for handling errors
❱ │   │   │   next_sleep = _retry_error_helper(
  │   │   │   │   exc,
  │   │   │   │   deadline,
  │   │   │   │   sleep_iter,

/usr/local/lib/python3.11/dist-packages/google/api_core/retry/retry_base
_retry_error_helper

  │   │   │   RetryFailureReason.NON_RETRYABLE_ERROR,
  │   │   │   original_timeout,
  │   │   )
❱ │   │   raise final_exc from source_exc
  │   if on_error_fn is not None:
  │   │   on_error_fn(exc)
  │   # next_sleep is fetched after the on_error callback, to allow

/usr/local/lib/python3.11/dist-packages/google/api_core/retry/retry_unar
retry_target
```

Start coding or generate with AI.