



DIGITAL ASSIGNMENT-6

Practice Questions

TRANSLATION USING GEMINI AND LANGCHAIN

January 05, 2025

SAI VARUN AITHA

saivarun.aitha18103@gmail.com

STEP-1

INSTALLING NECESSARY LIBRARIES

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\hello> pip install langchain google-cloud-aiplatform
Collecting langchain
  Downloading langchain-0.3.14-py3-none-any.whl.metadata (7.1 kB)
Collecting google-cloud-aiplatform
  Downloading google_cloud_aiplatform-1.75.0-py2.py3-none-any.whl.metadata (31 kB)
Requirement already satisfied: PyYAML>=5.3 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from langchain) (6.0.1)
Collecting SQLAlchemy<3,>=1.4 (from langchain)
  Downloading SQLAlchemy-2.0.36-cp312-cp312-win_amd64.whl.metadata (9.9 kB)
Collecting aiohttp<4.0.0,>=3.8.3 (from langchain)
  Downloading aiohttp-3.11.11-cp312-cp312-win_amd64.whl.metadata (8.0 kB)
Collecting langchain-core<0.4.0,>=0.3.29 (from langchain)
  Downloading langchain_core-0.3.29-py3-none-any.whl.metadata (6.3 kB)
Collecting langchain-text-splitters<0.4.0,>=0.3.3 (from langchain)
  Downloading langchain_text_splitters-0.3.4-py3-none-any.whl.metadata (2.3 kB)
Collecting langsmith<0.3,>=0.1.17 (from langchain)
  Downloading langsmith-0.2.10-py3-none-any.whl.metadata (14 kB)
Requirement already satisfied: numpy<3,>=1.26.2 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from langchain) (1.26.3)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from langchain) (2.10.4)
Requirement already satisfied: requests<3,>=2 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from langchain) (2.32.3)
Collecting tenacity!=8.4.0,<10,>=8.1.0 (from langchain)
  Downloading tenacity-9.0.0-py3-none-any.whl.metadata (1.2 kB)
Requirement already satisfied: google-api-core!=2.0.*,!=2.1.*,!=2.2.*,!=2.3.*,!=2.4.*,!=2.5.*,!=2.6.*,!=2.7.*,<3.0.0dev,>=1.34.1 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from google-api-core[grpc]==2.0.*,!=2.1.*,!=2.2.*,!=2.3.*,!=2.4.*,!=2.5.*,!=2.6.*,!=2.7.*,<3.0.0dev,>=1.34.1->google-cloud-aiplatform) (2.24.0)
Requirement already satisfied: google-auth<3.0.0dev,>=2.14.1 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from google-cloud-aiplatform) (2.37.0)
Requirement already satisfied: proto-plus<2.0.0dev,>=1.22.3 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from google-cloud-aiplatform) (1.25.0)
Requirement already satisfied: protobuf!=4.21.0,!4.21.1,!4.21.2,!4.21.3,!4.21.4,!4.21.5,<6.0.0dev,>=3.20.2 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from google-cloud-aiplatform) (5.29.2)
Requirement already satisfied: packaging>=14.3 in c:\users\hello\appdata\local\programs\python\python312\lib\site-packages (from google-cloud-aiplatform) (23.2)
Collecting google-cloud-storage<3.0.0dev,>=1.32.0 (from google-cloud-aiplatform)
  Downloading google_cloud_storage-2.19.0-py2.py3-none-any.whl.metadata (9.1 kB)
Collecting google-cloud-bigquery!=3.20.0,<4.0.0dev,>=1.15.0 (from google-cloud-aiplatform)
  Downloading google_cloud_bigquery-3.27.0-py2.py3-none-any.whl.metadata (8.6 kB)
```

STEP-2 IMPLEMENTING CODE-

```
File Edit Selection View Go Run ... my_portfolio
C:\Users\hello\Downloads> first.py
1 from langchain_google_genai.llms import GoogleGenerativeAI
2 from langchain.prompts import PromptTemplate
3 import os
4
5 # Load API key from environment variable
6 api_key = os.environ.get("GEMINI_API_KEY")
7
8 # Define prompt template
9 prompt_template = """
10 Translate the following text from {source_language} to {target_language}:
11
12 {text}
13 """
14 prompt = PromptTemplate(
15     template=prompt_template,
16     input_variables=["text", "source_language", "target_language"]
17 )
18
19 def translate(text, source_language, target_language):
20     """
21     Translates the given text using the specified languages.
22     """
23     llm = GoogleGenerativeAI(api_key=api_key)
24     prompt = prompt.format(text=text, source_language=source_language, target_language=target_language)
25     response = llm.invoke(prompt)
26     return response.text
```

```
from langchain_google_genai.llms import GoogleGenerativeAI
from langchain.prompts import PromptTemplate
import os

# Load API key from environment variable
api_key = os.environ.get("GEMINI_API_KEY")

# Define prompt template
prompt_template = """
Translate the following text from {source_language} to {target_language}:

"{text}"
"""

prompt = PromptTemplate(
    template=prompt_template,
    input_variables=["text", "source_language", "target_language"]
)

def translate(text, source_language, target_language):
    """
    Translates the given text using the specified languages.

    Args:
        text: The text to be translated.
        source_language: The source language of the text.
        target_language: The target language for translation.

    Returns:
```

The translated text.

```
"""
```

```
try:
```

```
    llm = GoogleGenerativeAI(model_name="Gemini-1.5 flash", type="translation",  
credentials={"api_key": api_key})
```

```
    final_prompt = prompt.format(  
        text=text, source_language=source_language, target_language=target_language  
    )
```

```
    translation = llm(final_prompt)
```

```
    return translation
```

```
except Exception as e:
```

```
    print(f"Translation failed: {e}")
```

```
    return None
```

Example Usage

```
source_text = "Hello, how are you?"
```

```
source_language = "English"
```

```
target_language = "Spanish"
```

```
translated_text = translate(source_text, source_language, target_language)
```

```
if translated_text:
```

```
    print(f"Translated text: {translated_text}")
```

OUTPUT-

Translated text: Hola, ¿cómo estás?