Python For DS

Final Assignment: AI Engineer Assignment

Group:

- Mai Phong Đăng: 22280008
- Trần Bá Đông: 22280011
- Trảo An Huy: 22280041
- Lê Trọng Nghĩa: 22280059

In [ ]:
```
!pip install langchain openai google-generativeai python-dotenv
!pip install langchain-community
```

In [ ]:
```
pip install faiss-gpu
```

test gemini API

In [3]:
```
import os
import google.generativeai as genai
from typing import List,Optional
genai.configure(api_key="AIzaSyAio6wKkYpRanuS8xsaGU5wfukwFirkwPo")
model = genai.GenerativeModel("gemini-1.5-flash")
response = model.generate_content("Explain how AI works")
print(response.text)
```

AI works by combining large amounts of data with fast, iterative processing and intelligent algorithms, allowing the software to learn automatically from patterns or features in the data. There's no single "how it works" because dif ferent AI approaches employ different techniques, but here's a breakdown of common principles:

1. Data Acquisition and Preparation:

* **Data Gathering:** AI systems need vast quantities of data – text, images, audio, sensor readings, etc. This dat a is often sourced from various places, including databases, APIs, sensors, and the internet.
* **Data Cleaning:** This crucial step involves handling missing values, removing inconsistencies, and transforming data into a usable format for the algorithms. This might include normalization, standardization, or feature enginee ring (creating new features from existing ones).
* **Data Labeling (for supervised learning):** Many AI systems require labeled data, meaning each data point is tagg ed with the correct answer or category. For example, in image recognition, images of cats would be labeled "cat."

2. Algorithm Selection and Training:

This stage involves choosing the appropriate algorithm and training it on the prepared data. Common approaches incl ude:

* **Supervised Learning:** The algorithm learns from labeled data to map inputs to outputs. Examples include linear regression, logistic regression, support vector machines (SVMs), and decision trees. The algorithm adjusts its inte rnal parameters to minimize the difference between its predictions and the actual labels.
* **Unsupervised Learning:** The algorithm learns patterns and structures from unlabeled data. Examples include clu stering (grouping similar data points) and dimensionality reduction (reducing the number of variables while preservi ng important information). This is used for tasks like anomaly detection and recommendation systems.
* **Reinforcement Learning:** The algorithm learns through trial and error by interacting with an environment. It r eceives rewards for desirable actions and penalties for undesirable ones. This is commonly used in robotics, game p laying, and control systems.
* **Deep Learning:** This is a subfield of machine learning that uses artificial neural networks with multiple layer s (hence "deep") to extract increasingly complex features from data. This approach has proven highly effective in a reas like image recognition, natural language processing, and speech recognition.

3. Model Evaluation and Refinement:

* **Testing:** Once trained, the model is tested on a separate dataset (not used during training) to evaluate its pe rformance. Metrics like accuracy, precision, recall, and F1-score are used to measure the model's effectiveness.
* **Refinement:** Based on the evaluation results, the model may be further refined. This could involve adjusting p arameters, using different algorithms, or gathering more data. This iterative process is crucial for building high-performing AI systems.

4. Deployment and Monitoring:

Once the model is deemed satisfactory, it's deployed to perform its intended task. This could involve integrating t into a software application, a website, or a physical system. The model's performance is continuously monitored, and adjustments may be made over time as new data becomes available or requirements change.

In short, AI works by learning patterns from data through sophisticated algorithms, allowing it to make predictions, decisions, or take actions without explicit programming for every scenario. The specific methods and complexities v ary widely depending on the application.

Question 1: LLM integration (Score: 30%)

The task involves building an AI capable of language translation.

1.1 Single Text Translation: (Score: 15%)

You are asked to write a Python code using the OpenAI API to translate a given text into Vietnamese (You should check the text if it's already the destination language . For example, translating "Hello" into Vietnamese should return "Xin chào", but "Xin chào" should return the same.

In [7]:
```
from google.colab import userdata
import os
os.environ['GEMINI_API_KEY'] = userdata.get('GEMINI_API_KEY')
os.environ['OPENGRAPH_APP_ID'] = userdata.get('OPENGRAPH_APP_ID')
```

In [8]:
```
def translate_to_vietnamese(text: str) -> Optional[str]:
    try:
        # Get API key from environment variable
        api_key= os.getenv('GEMINI_API_KEY')
        if not api_key:
            raise ValueError("Missing API key. Set GEMINI_API_KEY environment variable")

        # Configure the API
        genai.configure(api_key=api_key)
        model = genai.GenerativeModel("gemini-1.5-flash")

        # Create the prompt
        prompt = "Translate this text to Vietnamese. If it's already in Vietnamese, return it unchanged.."
```

```
        # Generate the translation
        response = model.generate_content(
        prompt + text,
        generation_config={
                "temperature": 0,
                "top_p": 0
            }
        )

        return response.text

    except Exception as e:
        print(f"Translation error: {str(e)}")
        return None

if __name__ == "__main__":

    text = input("Enter text to translate: ")
    result = translate_to_vietnamese(text)

    if result:
        print("Result:", result)
    else:
        print("Translation failed")
```

```
Enter text to translate: hello
Result: Xin chào
```

1.2 Multiple Texts Translation: (Score: 15%)

Similar to 2.1, but the input is a list of texts. The Python code should accept a list of strings and return their translations in the specified language. For instance, translating ["Hello", "I am John", "Tôi là sinh viên"] into Vietnamese should return ["Xin chào", "Tôi tên là John", "Tôi là sinh viên"].

In [9]:
```
def translate_texts_to_vietnamese(texts: List[str], api_key: str) -> List[str]:
  translations = []
  for text in texts:
    try:
        # Get API key from environment variable
        api_key= os.getenv('GEMINI_API_KEY')
        if not api_key:
            raise ValueError("Missing API key. Set GEMINI_API_KEY environment variable")

        # Configure the API
        genai.configure(api_key=api_key)
        model = genai.GenerativeModel("gemini-1.5-flash")

        # Create the prompt
        prompt = "Translate this text to Vietnamese. If it's already in Vietnamese, return it unchanged: "

        # Generate the translation
        response = model.generate_content(prompt + text)
        translated_text = response.text.strip()

        # Add to results
        translations.append(translated_text)

    except Exception as e:
        print(f"Translation error: {str(e)}")
  return translations

if __name__ == "__main__":
    # Your API key
    api_key = os.getenv('GEMINI_API_KEY')
    # Example texts
    sample_texts = ["Hello", "I am John", "Tôi là sinh viên"]

    # Get translations
    translated_texts = translate_texts_to_vietnamese(sample_texts, api_key)

    # Print results
    print (translated_texts)
    for original, translated in zip(sample_texts, translated_texts):
        print(f"Original: {original}")
        print(f"Translated: {translated}")
        print("-" * 30)
```

```
['Xin chào', 'Tôi là John', 'Tôi là sinh viên']
Original: Hello
Translated: Xin chào
------------------------------
Original: I am John
Translated: Tôi là John
------------------------------
Original: Tôi là sinh viên
Translated: Tôi là sinh viên
------------------------------
```

Question 2: Chatbot Development (Score: 70%)

Assignment Test: Chatbot Development from Website Data. The data is at [https://www.presight.io/privacy-policy.html]

2.1 Data Access and Indexing (Score: 40%)

Tasked with creating a chatbot, begin by web crawling the specified website to gather relevant data, then preprocess and structure this data into a searchable index, ready for query retrieval. (Short version: crawling then embedding data, you can use selenium or requests)

2.2 Chatbot Development (Score: 30%)

Develop a chatbot that employs natural language processing to comprehend user questions, searches the indexed data from 2.1 for the best match, and delivers the most accurate response drawn from the website's information. (Use any distance/similarity metrics to get the best match paragraph then feed to LLM to get answer)

In [10]:
```python
import requests
import json
from urllib.parse import quote

original_url = "https://www.presight.io/privacy-policy.html"
base_url = "https://opengraph.io/api/1.1/extract"
encoded_url = quote(original_url, safe='')
url = f"{base_url}/{encoded_url}"
app_id = os.getenv('OPENGRAPH_APP_ID')
# Api params
params = {
    "accept_lang": "auto",
    "html_elements": "h1,h2,p",
    "app_id": app_id
}

def fetch_data():
    try:
        response = requests.get(url, params=params)
        response.raise_for_status()
        data = response.json()

        # Print out formatted json
        formatted_json = json.dumps(data, indent=2, ensure_ascii=False)
        print(formatted_json)

    except requests.exceptions.RequestException as e:
        print(f"An error occurred: {e}")

# Function call
fetch_data()
```

```
{
  "tags": [
    {
      "tag": "p",
      "innerText": "By Role",
      "position": 0
    },
    {
      "tag": "p",
      "innerText": "By Team",
      "position": 1
    },
    {
      "tag": "h2",
      "innerText": "PRIVACY POLICY",
      "position": 2
    },
    {
      "tag": "h2",
      "innerText": "Last updated 15 Sep 2023",
      "position": 3
    },
    {
      "tag": "p",
      "innerText": "At Presight, we are committed to protecting the privacy of our customers and visitors to our web
site. This Privacy Policy explains how we collect, use, and disclose information about our customers and visitors.",
      "position": 4
    },
    {
      "tag": "h2",
      "innerText": "Information Collection and Use",
      "position": 5
    },
    {
      "tag": "p",
      "innerText": "We collect several different types of information for various purposes to provide and improve ou
r Service to you.",
      "position": 6
    },
    {
      "tag": "h2",
      "innerText": "Types of Data Collected",
      "position": 7
    },
    {
      "tag": "p",
      "innerText": "While using our Service, we may ask you to provide us with certain personally identifiable infor
mation that can be used to contact or identify you (\"Personal Data\"). Personally identifiable information may incl
ude, but is not limited to:",
      "position": 8
    },
    {
      "tag": "p",
      "innerText": "We may also collect information that your browser sends whenever you visit our Service or when y
ou access the Service by or through a mobile device (\"Usage Data\"). This Usage Data may include information such a
s your computer's Internet Protocol address (e.g. IP address), browser type, browser version, the pages of our Servi
ce that you visit, the time and date of your visit, the time spent on those pages, unique device identifiers, and ot
her diagnostic data.",
      "position": 9
    },
    {
      "tag": "h2",
      "innerText": "Use of Data",
      "position": 10
    },
    {
      "tag": "p",
      "innerText": "Presight uses the collected data for various purposes:",
      "position": 11
    },
    {
      "tag": "h2",
      "innerText": "Consent",
      "position": 12
    },
    {
      "tag": "p",
      "innerText": "As personal information is collected, you will be asked to confirm that your information is corr
ect prior to submitting it to Presight.",
      "position": 13
    },
    {
      "tag": "h2",
      "innerText": "Access to Personal Information",
      "position": 14
```

```
    },
    {
      "tag": "h2",
      "innerText": "Accessing Your Personal Information",
      "position": 15
    },
    {
      "tag": "p",
      "innerText": "You have the right to access all of your personal information that we hold. Through the applicat
ion, you can correct, amend, or append your personal information by logging into the application and navigating to y
our settings and profile.",
      "position": 16
    },
    {
      "tag": "h2",
      "innerText": "Automated Edit Checks",
      "position": 17
    },
    {
      "tag": "p",
      "innerText": "Presight employs automated edit checks to ensure that data entry fields are completed properly w
hen collecting personal information. These edit checks help maintain data integrity and accuracy. You are encouraged
to provide complete and valid information to ensure the smooth processing of their personal data.",
      "position": 18
    },
    {
      "tag": "h2",
      "innerText": "Disclosure of Information",
      "position": 19
    },
    {
      "tag": "p",
      "innerText": "We may disclose your application data to third-party service providers who help us provide our s
ervices such as Datadog, AWS, Google Cloud and Google Workspace. We may also disclose your information in response t
o a legal request, such as a subpoena or court order, or to protect our rights or the rights of others.",
      "position": 20
    },
    {
      "tag": "h2",
      "innerText": "Sharing of Personal Data",
      "position": 21
    },
    {
      "tag": "p",
      "innerText": "Your personal data will not be subject to sharing, transfer, rental or exchange for the benefit
of third parties, including AI models.",
      "position": 22
    },
    {
      "tag": "h2",
      "innerText": "Google User Data and Google Workspace APIs",
      "position": 23
    },
    {
      "tag": "p",
      "innerText": "In all cases when users authenticate the platform to Google Workspace, the following applies:",
      "position": 24
    },
    {
      "tag": "h2",
      "innerText": "Data Security",
      "position": 25
    },
    {
      "tag": "h2",
      "innerText": "Data Retention & Disposal",
      "position": 26
    },
    {
      "tag": "p",
      "innerText": "Customer data is retained for as long as the account is in active status. Data enters an “expire
d” state when the account is voluntarily closed. Expired account data will be retained for 60 days. After this perio
d, the account and related data will be removed.",
      "position": 27
    },
    {
      "tag": "h2",
      "innerText": "Quality, Including Data Subjects' Responsibilities for Quality",
      "position": 28
    },
    {
      "tag": "h2",
      "innerText": "Monitoring and Enforcement",
      "position": 29
    },
```

```
    {
      "tag": "h2",
      "innerText": "Cookies",
      "position": 30
    },
    {
      "tag": "p",
      "innerText": "We use cookies to enhance your experience on our website. You can control the use of cookies thr
ough your web browser settings.",
      "position": 31
    },
    {
      "tag": "h2",
      "innerText": "Third-Party Websites",
      "position": 32
    },
    {
      "tag": "p",
      "innerText": "Our website may contain links to third-party websites. We are not responsible for the privacy pr
actices or content of those websites.",
      "position": 33
    },
    {
      "tag": "h2",
      "innerText": "Changes to Privacy Policy",
      "position": 34
    },
    {
      "tag": "p",
      "innerText": "We may update this Privacy Policy from time to time. The updated Privacy Policy will be posted o
n our website.",
      "position": 35
    },
    {
      "tag": "h2",
      "innerText": "Contact Us",
      "position": 36
    },
    {
      "tag": "p",
      "innerText": "If you have any questions about this Privacy Policy, please contact us through the customer port
al or by email at .css-glrzqe{cursor:pointer;color:var(--chakra-colors-blue-500);}presight@presight.io.",
      "position": 37
    },
    {
      "tag": "h2",
      "innerText": "Purposeful Use Only",
      "position": 38
    },
    {
      "tag": "p",
      "innerText": "We commit to only use personal information for the purposes identified in the entity's privacy p
olicy.",
      "position": 39
    },
    {
      "tag": "p",
      "innerText": "Presight.io 2022 All Rights Reserved",
      "position": 40
    },
    {
      "tag": "p",
      "innerText": "Ho Chi Minh City, Vietnam",
      "position": 41
    },
    {
      "tag": "p",
      "innerText": "Singapore",
      "position": 42
    },
    {
      "tag": "p",
      "innerText": "Seattle, WA, USA",
      "position": 43
    }
  ],
  "concatenatedText": "By Role By Team PRIVACY POLICY Last updated 15 Sep 2023 At Presight, we are committed to prot
ecting the privacy of our customers and visitors to our website. This Privacy Policy explains how we collect, use, a
nd disclose information about our customers and visitors. Information Collection and Use We collect several differen
t types of information for various purposes to provide and improve our Service to you. Types of Data Collected While
using our Service, we may ask you to provide us with certain personally identifiable information that can be used to
contact or identify you (\"Personal Data\"). Personally identifiable information may include, but is not limited to:
We may also collect information that your browser sends whenever you visit our Service or when you access the Servic
e by or through a mobile device (\"Usage Data\"). This Usage Data may include information such as your computer's In
ternet Protocol address (e.g. IP address), browser type, browser version, the pages of our Service that you visit, t
```

he time and date of your visit, the time spent on those pages, unique device identifiers, and other diagnostic data. Use of Data Presight uses the collected data for various purposes: Consent As personal information is collected, you will be asked to confirm that your information is correct prior to submitting it to Presight. Access to Personal Information Accessing Your Personal Information You have the right to access all of your personal information that we hold. Through the application, you can correct, amend, or append your personal information by logging into the application and navigating to your settings and profile. Automated Edit Checks Presight employs automated edit checks to ensure that data entry fields are completed properly when collecting personal information. These edit checks help maintain data integrity and accuracy. You are encouraged to provide complete and valid information to ensure the smooth processing of their personal data. Disclosure of Information We may disclose your application data to third-party service providers who help us provide our services such as Datadog, AWS, Google Cloud and Google Workspace. We may also disclose your information in response to a legal request, such as a subpoena or court order, or to protect our rights or the rights of others. Sharing of Personal Data Your personal data will not be subject to sharing, transfer, rental or exchange for the benefit of third parties, including AI models. Google User Data and Google Workspace APIs In all cases when users authenticate the platform to Google Workspace, the following applies: Data Security Data Retention & Disposal Customer data is retained for as long as the account is in active status. Data enters an “expired” state when the account is voluntarily closed. Expired account data will be retained for 60 days. After this period, the account and related data will be removed. Quality, Including Data Subjects' Responsibilities for Quality Monitoring and Enforcement Cookies We use cookies to enhance your experience on our website. You can control the use of cookies through your web browser settings. Third-Party Websites Our website may contain links to third-party websites. We are not responsible for the privacy practices or content of those websites. Changes to Privacy Policy We may update this Privacy Policy from time to time. The updated Privacy Policy will be posted on our website. Contact Us If you have any questions about this Privacy Policy, please contact us through the customer portal or by email at .css-glrzqe{cursor:pointer;color:var(--chakra-colors-blue-500);}presight@presight.io. Purposeful Use Only We commit to only use personal information for the purposes identified in the entity's privacy policy. Presight.io 2022 All Rights Reserved Ho Chi Minh City, Vietnam Singapore Seattle, WA, USA "
}

In [11]:
```
import google.generativeai as genai
import os
api_key = os.getenv('GEMINI_API_KEY')
genai.configure(api_key = api_key)
result = genai.embed_content(
        model="models/text-embedding-004",
        content="What is the meaning of life?")

print(str(result['embedding']))
```

[-0.010632273, 0.019375853, 0.020965198, 0.0007706437, -0.061464068, 0.014739741, -0.0022759985, 0.013184195, 0.014464715, 0.022593116, 0.02184836, -0.059616957, 0.06032222, -0.047657482, 0.017848385, -0.10987464, -0.0598155, -0.00479664, -0.043298274, -0.05090505, 0.029398112, 0.011642447, 0.04183789, -0.017999396, 0.011026355, 0.049722955, 0.012025892, 0.007331535, 0.01967245, -0.0025621902, 0.028765293, 0.0068937168, 0.0029231338, -0.0002095079, 0.032031864, 0.02518659, -0.032855466, 0.00758291, -0.00011585959, -0.034515556, -0.066151336, 0.03191643, -0.026680378, 0.017334407, -0.025778342, -0.008119435, -0.002431255, -0.009850676, -0.030725427, 0.08225489, 0.036220998, -0.011677602, -0.048477963, 0.026030278, 0.0027632737, -0.036962725, -0.051528536, -0.027265795, 0.04703419, -0.03285586, -0.01514 0722, -0.003516825, -0.006665491, -0.024252947, -0.031371485, 0.056986455, -0.02846856, 0.009047717, -0.021733612, 0.01993043, -0.016926913, 0.051008012, -0.022356581, 0.05340387, -0.036262874, 0.038486782, 0.00097307086, 0.0058653215, -0.03454564, 0.038883448, -0.020346535, -0.0015010178, 0.050026324, 0.07690296, 0.04075089, 0.031162778, -0.048467305, -0.031640615, -0.050462708, -0.0020114628, 0.028352365, 0.016939064, -0.032321587, -0.017523259, 0.045018278, 0.005056391, -0.08844299, -0.039214693, 0.032369446, 0.013868324, 0.048252415, 0.012212794, -0.0059761675, -0.055453815, 0.059123088, 0.077673666, 0.012595949, -0.030664278, 0.0019445478, -0.04473188, 0.03904732, -0.045189187, 0.005711123, -0.024350755, 0.006020905, -0.056398984, -0.008473793, 0.026584638, -0.019225147, -0.003090118, 0.02925659, 0.037855238, -0.033372607, 0.027388284, 0.058645032, -0.0034353225, -0.00052528176, -0.061926123, -0.047651615, -0.020240242, 0.037042357, -0.101258375, -0.017224912, 0.031264607, -0.029515961, 0.04070285, 0.08155317, -0.026804389, 0.010762277, -0.068192326, 0.010339065, 0.001237995, 0.025081903, 0.025549553, 0.033473987, -0.011019555, 0.025582748, -0.044487614, -0.02351738, -0.019466395, -0.05739292, -0.023219999, 0.06383781, -0.0032941306, 0.0032277782, 0.014958662, 0.037334923, 0.010649138, 0.07434867, -0.024096856, -0.0051036896, -0.05779452, -0.087558694, 0.005070572, -0.059070442, -0.0075670946, 0.020864079, -0.059642896, -0.017373137, 0.010781379, 0.005737636, 0.01155112, -0.051110126, -0.00469127, 0.003082495, 0.021098692, -0.010646007, -0.0075031, 0.01649139, -0.010034379, 0.03665796, -0.02178521, -0.060414966, -0.0110657895, -0.018490821, -0.038217384, -0.008570785, 0.06764553, 0.045524262, 0.028033433, -0.049999256, 0.038643356, -0.001174409, 0.0071052625, -0.0071540354, -0.03563122, 0.040300176, -0.01187511, -0.020187229, 0.034496624, -0.018076168, -0.025241721, -0.03251734, -0.005546835, 0.01218167, 0.001308468, -0.01956061, -0.016109072, 0.033482637, -0.013107253, -0.04336891, 0.017510926, -0.059141196, -0.023261068, 0.025163788, 0.04890369, 0.076442, -0.0016504959, 0.10172619, -0.015871631, -0.023793343, -0.02358568, 0.036539588, -0.06184051, -0.026249573, 0.006468363, -0.031341415, -0.06234132, -0.049488295, -0.018885756, 0.03395302, -0.006009219, -0.031574816, -0.0054155374, -0.033587996, -0.015623983, 0.013743329, 0.06735172, 0.009166206, -0.027008668, 0.053747576, -0.019794546, -0.004977181, -0.0011775235, 0.055169225, 0.031791825, 0.025199965, 0.080965735, 0.0039748563, -0.08897454, -0.027933061, 0.00045005116, -0.013844743, -0.06260468, -0.046366338, -0.029402703, 0.023191761, -0.01076239, 0.0076124803, -0.020023048, 0.039004155, -0.070678934, -0.07069906, -0.02288811, -0.03803117, -0.05004868, -0.018108511, -0.024550572, 0.040691372, -0.05350907, 0.051243976, -0.0021085127, -0.009738572, -0.008890091, -0.015601183, 0.019753162, 0.0053467727, 0.031590473, 0.0041920035, -0.04371269, 0.067351475, -0.019107815, -0.014121782, 0.009763881, 0.031802285, -0.0069985087, 0.013498973, 0.023104675, 0.0006382101, -0.008508383, 0.03777484, 0.008196443, -0.0025804106, -0.033261176, -0.033644095, -0.0039042186, 0.049756225, 0.03194955, 0.018670397, -0.004185749, 0.01654144, 0.063628886, -0.08167434, 0.004465523, 0.0054312716, 0.00061390334, 0.02128485, 0.0031732921, -0.025789104, 0.006552007, -0.039853606, -0.009466623, -0.021836154, 0.08548205, -0.06237011, -0.035231795, -0.09519183, -0.02711923, 0.00037482058, 0.0036829626, 0.015760176, 0.015482902, 0.004761403, 0.025655402, 0.07554531, -0.043909427, -0.0041645113, 0.031294763, -0.0028018486, -0.011339259, -0.0031232522, -0.02227631, 0.004836296, -0.009918578, 0.029489264, 0.024922853, -0.028259983, 0.037678096, 0.022683982, 0.07546214, -0.0070300903, -0.023265228, -0.0025721574, 0.01389813, -0.010174201, -0.0040706755, -0.025229212, 0.008944433, -0.025699921, -0.060985804, 0.0058162743, 0.07175555, 0.032720394, -0.036219627, 0.011701761, -0.012563732, 0.06423104, 0.022426128, 0.008510076, 0.011255559, -0.048804004, -0.017703402, -0.007979923, -0.018820668, -0.0053055533, 0.009278715, 0.017546115, 0.055455945, 0.043846007, -0.022937374, 0.09124366, -0.0059768124, -0.016920665, 0.0077798367, -0.007818393, 0.0030480593, -0.05119679, 0.0072891167, 0.020954333, -0.08999456, -0.036280062, -0.058427356, -0.053980652, 0.02610353, -0.023728639, 0.032551993, -0.032998607, -0.010366301, -0.004644334, 0.0052025192, -0.036866736, 0.037116528, 0.01658842, 0.024684586, -0.024388108, -0.005666494, -0.03671624, 0.008723972, -0.01812843, 0.019828215, 0.010995856, -0.019123131, 0.10374082, -0.038003173, -0.025865225, -0.0029166006, 0.09824402, 0.006400806, 0.011756453, -0.057788208, -0.03922871, 0.029061263, 0.06839164, -0.014544535, -0.047662966, -0.059395976, -0.03727927, 0.014318371, 0.025973465, 0.042332895, 0.04511835, -0.039885864, 0.04445013, -0.00909842, -0.0022268177, -0.055778414, 0.044562876, -0.0029349416, 0.0045089596, 0.04649308, 0.05095703, 0.024818162, -0.01763042, -0.016380813, 0.03626134, 0.029747656, -0.018518452, 0.054535143, -0.03725233, 0.015218341, -0.035265, -0.008258693, 0.016336355, 0.003180061, 0.017113037, 0.013840924, 0.08571888, -0.016922096, -0.045846672, -0.026123295, -0.01862711, 0.00086665194, -0.02700387, -0.039896443, 0.025839228, -0.008957712, -0.045702096, 0.011689191, -0.02518643, 0.04189632, 0.024877924, -0.029749716, 0.07723543, 0.013161921, 0.035233274, 0.013950026, -0.026914261, -0.0012491347, 0.022125386, 0.06322952, 0.026747808, 0.016557682, 0.0026654843, 0.018403852, -0.002208754, -0.0043939324, 0.021411125, -0.0720841, -0.014162335, 0.009017187, 0.009589008, 0.013714266, -0.013205313, 0.055074606, 0.0135510815, -0.009647225, -0.0073859296, -0.015533789, 0.041406598, -0.029964559, -0.004557068, 0.042442538, 0.003949693, -0.060314845, -0.0485521, -0.008145191, -0.0008701478, 0.026269091, 0.064659014, -0.0014519938, 0.07755499, 0.012390666, 0.0009994709, 0.010512895, -0.0278039, -0.007720246, -0.017693883, -0.048093677, 0.048450127, -0.0084898835, 0.033827696, 0.012179157, 0.0439037, 0.019806726, -0.0033815033, 0.055004198, -0.010644163, 0.069836334, -0.0012867257, 0.116212435, 0.0018561919, -0.03540732, 0.018552277, -0.014596015, 0.007995569, 0.02062322, -0.013589375, 0.013323644, 0.058206026, 0.014310659, 0.009776701, 0.022025304, 0.043452848, 0.007224779, -0.005841782, 0.007922995, 0.029124206, 0.027332257, 0.011426645, 0.0610715, 0.033370003, -0.0032318854, 0.032962296, 0.044215627, -0.0019828111, -0.015901793, -0.00029608337, 0.013392526, -0.009583505, 0.101087496, 0.029640157, -0.04264001, 0.028663691, 0.0012885618, -0.00042037942, -0.05097693, 0.046501413, 0.034346417, -0.03722956, 0.030485353, -0.028618095, -0.014943351, 0.024176005, 0.0059531713, -0.035492424, 0.04719729, -0.022705767, -0.004888659, 0.013763481, -0.006877845, 0.039462008, -0.022432147, -0.024738846, -0.0030126623, 0.014878597, 0.047142185, -0.028536918, -0.0019756483, -0.024875728, -0.049604762, 0.0076611727, 0.0125418445, 0.06991834, 0.03057351, -0.0378383, -0.01601651, 0.023397712, -0.006465213, -0.016750913, -0.028563995, 0.013968368, 0.04284747, 0.013723971, -0.038290635, 0.0062841102, -0.016612995, 0.0060477066, 0.0071878443, 0.017012084, 0.026105886, -0.029898316, -0.0034338816, 0.022605129, -0.031070229, -0.014588787, -0.05051719, 0.011172559, -0.009865424, -0.0006022246, -0.050201006, 0.010974502, 0.068753, -0.064111741, 0.021827834, -0.079100326, 0.027182067, -0.0049233013, -0.00854883, 0.042056426, -0.041176684, -0.043345083, 0.007900265, 0.03339074, 0.009065677, -0.11376203, 0.026648033, -0.02173746, -0.056054536, -0.05019967, 0.02505995, -0.073714115, 0.00041753243, 0.046410866, -0.00787225, -0.04326591, 0.052950215, -0.020633917, 0.0053053875, 0.038686555, 0.0076096985, -0.044483498, 0.01734079, 0.050843734, 0.041709274, -0.032848667, 0.06583798, -0.0462481, -0.019906212, 0.062381882, 0.010934914, -0.053675517, -0.033560812, 0.027787214, 0.003391649, 0.019972153, 0.0442223, -0.06779605, -0.057355773, -0.00908375, -0.031183494, 0.07079641, -0.020006215, -0.024294054, -0.016699182, 0.0010443482, 0.018393427, 0.032058917, 0.04007311, -0.013608359, -0.02647255, -0.023104627, 0.07973177, 0.0143912, -0.00773088, -0.0105773965, 0.009673522, 0.030086972, 0.021788783, 0.0215211, -0.0021278693, 0.01382664, -0.05028589, 0.0037969938, -0.019241702, -0.055900373, 0.047401533, 0.047825735, -0.008378417, -0.021368338, -0.0029360335, -0.023776283, -0.030378696, 0.0042622155, -0.04370354, 0.046717755, 0.057218548, -0.07626953, 0.06840914, -0.013551472, -0.040814577, 0.0024602, -0.019596782, -0.034115944, 0.022949563, 0.08198656, 0.010917071, 0.012808682, -0.0024835565, -0.06742202, 0.035741765, -0.007581535, 0.01281636, 0.05919395, 0.019007294, -0.057466842, 0.031478077, 0.011478408, 0.0197156, 0.03522307, -0.0039083306, 0.009473974, -0.061164707, -0.010365365, 0.020075476, 0.025542602, -0.030813247, -0.050739173, -0.0037222796, -0.0025314046, 0.03607207, 0.085864864, 0.030587368, -0.011790973, 0.02897135, 0.009813614,

```
0.0036375853, 0.01939262, -0.012913535, 0.032164395, -0.012496243, 0.053962503, -0.0030092895, -0.013271072, -0.0691
50545, -0.014564991, 0.01531648, -0.0493518, -0.026759734, -0.030610656, -0.022655917, -0.09071128, -0.051921394, -
0.014159941, 0.086534575, 0.039204597, -0.018607471, -0.023076432, 0.016071219, 0.08200573, 0.036090653, -0.00292501
41, 0.032362826, -0.014467054, 0.013964356, -0.075049624, 0.047506943, -0.007153866, -0.028534686]
```

In [14]:

```python
import os
import requests
import json
from urllib.parse import quote

# LangChain imports
from langchain.embeddings.base import Embeddings
from langchain.vectorstores import FAISS
from langchain.text_splitter import RecursiveCharacterTextSplitter
from langchain.chains import RetrievalQA
from langchain.prompts import PromptTemplate
from langchain.llms.base import LLM

# Gemini imports
import google.generativeai as genai

# Configure Gemini API
GEMINI_API_KEY = os.getenv('GEMINI_API_KEY')
OPENGRAPH_APP_ID = os.getenv('OPENGRAPH_APP_ID')
if not GEMINI_API_KEY or not OPENGRAPH_APP_ID:
    raise ValueError("Please set GEMINI_API_KEY and OPENGRAPH_APP_ID environment variables")

genai.configure(api_key=GEMINI_API_KEY)

# Gemini embeddings class
class GeminiEmbeddings(Embeddings):
    def embed_documents(self, texts):
        embeddings = []
        for text in texts:
            result = genai.embed_content(
                model="models/text-embedding-004",
                content=text
            )
            embeddings.append(result["embedding"])
        return embeddings

    def embed_query(self, text):
        result = genai.embed_content(
            model="models/text-embedding-004",
            content=text
        )
        return result["embedding"]

# Retrieve and process URL content
def get_url_content(url):
    base_url = "https://opengraph.io/api/1.1/extract"
    encoded_url = quote(url, safe='')
    api_url = f"{base_url}/{encoded_url}"

    params = {
        "accept_lang": "auto",
        "html_elements": "h1,h2,p",
        "app_id": OPENGRAPH_APP_ID
    }
    try:
        response = requests.get(api_url, params=params)
        response.raise_for_status()
        return response.json()
    except requests.exceptions.RequestException as e:
        print(f"Error accessing the URL: {e}")
        return None

def process_extracted_data(data):
    text_contents = []
    for tag in data.get('tags', []):
        if tag.get('innerText'):
            text_contents.append(tag['innerText'])

    text_splitter = RecursiveCharacterTextSplitter(
        chunk_size=2000,
        chunk_overlap=200,
        separators=["\n\n", "\n", " ", ""]
    )
    chunks = text_splitter.split_text('\n'.join(text_contents))

    embeddings = GeminiEmbeddings()
    vectorstore = FAISS.from_texts(chunks, embeddings)
    return vectorstore

# Custom LLM wrapper for Gemini
```

```python
class GeminiLLM(LLM):
    def _call(self, prompt, stop=None):
        model = genai.GenerativeModel("gemini-1.5-flash")
        response = model.generate_content(prompt)
        return response.text

    @property
    def _identifying_params(self):
        return {"name_of_model": "gemini-1.5-flash"}

    @property
    def _llm_type(self):
        return "custom-gemini"

# Create a chain with a PromptTemplate
def create_retrieval_qa_chain(vectorstore):
    prompt_template = PromptTemplate(
        input_variables=["context", "question"],
        template=(
            "You are a helpful assistant. Use the following context:\n"
            "{context}\n\n"
            "Answer the question below using only the provided context, If you don't know the answer, just say that
            "Question: {question}\n"
            "Answer:"
        ),
    )

    chain = RetrievalQA.from_chain_type(
        llm=GeminiLLM(),
        chain_type="stuff",
        retriever=vectorstore.as_retriever(),
        return_source_documents=True
    )

    return chain, prompt_template

# Main function to run the application
def main():
    url = "https://www.presight.io/privacy-policy.html"
    vectorstore_path = "vectorstore_privacy_policy"

    try:
        if os.path.exists(vectorstore_path):
            print("Loading existing vector store...")
            embeddings = GeminiEmbeddings()
            vectorstore = FAISS.load_local(vectorstore_path, embeddings, allow_dangerous_deserialization=True)
        else:
            print("Downloading and processing content...")
            data = get_url_content(url)
            if data:
                vectorstore = process_extracted_data(data)
                vectorstore.save_local(vectorstore_path)
            else:
                print("Failed to retrieve content.")
                return

        chain, prompt_template = create_retrieval_qa_chain(vectorstore)
        print("Chain created successfully!")

        while True:
            user_query = input("\nEnter your question (or 'quit' to exit): ").strip()
            if user_query.lower() == "quit":
                print("Thank you for using the QA system!")
                break
            if not user_query:
                print("Please enter a valid question.")
                continue

            docs = chain.retriever.get_relevant_documents(user_query)
            context = "\n".join([d.page_content for d in docs])

            prompt = prompt_template.format(context=context, question=user_query)

            answer_results = chain({"query": user_query})
            answer = answer_results["result"]
            print("\n---------------------------Context Used---------------------------------------------------:
            print("-------------------------------------------------------------------------------------------
            print("\n----------------------Answer from Chatbot--------------------------:\n", answer)

    except Exception as e:
        print(f"An error occurred: {e}")

if __name__ == "__main__":
    main()
```

Loading existing vector store...
Chain created successfully!

Enter your question (or 'quit' to exit): what is data use for?

----------Context Used----------:
Disclosure of Information
We may disclose your application data to third-party service providers who help us provide our services such as Data dog, AWS, Google Cloud and Google Workspace. We may also disclose your information in response to a legal request, such as a subpoena or court order, or to protect our rights or the rights of others.
Sharing of Personal Data
Your personal data will not be subject to sharing, transfer, rental or exchange for the benefit of third parties, including AI models.
Google User Data and Google Workspace APIs
In all cases when users authenticate the platform to Google Workspace, the following applies:
Data Security
Data Retention & Disposal
Customer data is retained for as long as the account is in active status. Data enters an “expired” state when the account is voluntarily closed. Expired account data will be retained for 60 days. After this period, the account and related data will be removed.
Quality, Including Data Subjects' Responsibilities for Quality
Monitoring and Enforcement
Cookies
We use cookies to enhance your experience on our website. You can control the use of cookies through your web browser settings.
Third-Party Websites
Our website may contain links to third-party websites. We are not responsible for the privacy practices or content of those websites.
Changes to Privacy Policy
We may update this Privacy Policy from time to time. The updated Privacy Policy will be posted on our website.
Contact Us
If you have any questions about this Privacy Policy, please contact us through the customer portal or by email at .css-glrzqe{cursor:pointer;color:var(--chakra-colors-blue-500);}presight@presight.io.
Purposeful Use Only
We commit to only use personal information for the purposes identified in the entity's privacy policy.
Presight.io 2022 All Rights Reserved
Ho Chi Minh City, Vietnam
Singapore
Seattle, WA, USA
By Role
By Team
PRIVACY POLICY
Last updated 15 Sep 2023
At Presight, we are committed to protecting the privacy of our customers and visitors to our website. This Privacy Policy explains how we collect, use, and disclose information about our customers and visitors.
Information Collection and Use
We collect several different types of information for various purposes to provide and improve our Service to you.
Types of Data Collected
While using our Service, we may ask you to provide us with certain personally identifiable information that can be used to contact or identify you ("Personal Data"). Personally identifiable information may include, but is not limited to:
We may also collect information that your browser sends whenever you visit our Service or when you access the Service by or through a mobile device ("Usage Data"). This Usage Data may include information such as your computer's Internet Protocol address (e.g. IP address), browser type, browser version, the pages of our Service that you visit, the time and date of your visit, the time spent on those pages, unique device identifiers, and other diagnostic data.
Use of Data
Presight uses the collected data for various purposes:
Consent
As personal information is collected, you will be asked to confirm that your information is correct prior to submitting it to Presight.
Access to Personal Information
Accessing Your Personal Information
You have the right to access all of your personal information that we hold. Through the application, you can correct, amend, or append your personal information by logging into the application and navigating to your settings and profile.
Automated Edit Checks
Presight employs automated edit checks to ensure that data entry fields are completed properly when collecting personal information. These edit checks help maintain data integrity and accuracy. You are encouraged to provide complete and valid information to ensure the smooth processing of their personal data.

----------Answer from Chatbot----------:
Presight uses collected data to provide and improve its service.

Enter your question (or 'quit' to exit): what is policy?

----------Context Used----------:
Disclosure of Information
We may disclose your application data to third-party service providers who help us provide our services such as Data dog, AWS, Google Cloud and Google Workspace. We may also disclose your information in response to a legal request, such as a subpoena or court order, or to protect our rights or the rights of others.
Sharing of Personal Data
Your personal data will not be subject to sharing, transfer, rental or exchange for the benefit of third parties, including AI models.

Google User Data and Google Workspace APIs
In all cases when users authenticate the platform to Google Workspace, the following applies:
Data Security
Data Retention & Disposal
Customer data is retained for as long as the account is in active status. Data enters an “expired” state when the ac
count is voluntarily closed. Expired account data will be retained for 60 days. After this period, the account and r
elated data will be removed.
Quality, Including Data Subjects' Responsibilities for Quality
Monitoring and Enforcement
Cookies
We use cookies to enhance your experience on our website. You can control the use of cookies through your web browse
r settings.
Third-Party Websites
Our website may contain links to third-party websites. We are not responsible for the privacy practices or content o
f those websites.
Changes to Privacy Policy
We may update this Privacy Policy from time to time. The updated Privacy Policy will be posted on our website.
Contact Us
If you have any questions about this Privacy Policy, please contact us through the customer portal or by email at .c
ss-glrzqe{cursor:pointer;color:var(--chakra-colors-blue-500);}presight@presight.io.
Purposeful Use Only
We commit to only use personal information for the purposes identified in the entity's privacy policy.
Presight.io 2022 All Rights Reserved
Ho Chi Minh City, Vietnam
Singapore
Seattle, WA, USA
By Role
By Team
PRIVACY POLICY
Last updated 15 Sep 2023
At Presight, we are committed to protecting the privacy of our customers and visitors to our website. This Privacy P
olicy explains how we collect, use, and disclose information about our customers and visitors.
Information Collection and Use
We collect several different types of information for various purposes to provide and improve our Service to you.
Types of Data Collected
While using our Service, we may ask you to provide us with certain personally identifiable information that can be u
sed to contact or identify you ("Personal Data"). Personally identifiable information may include, but is not limite
d to:
We may also collect information that your browser sends whenever you visit our Service or when you access the Servic
e by or through a mobile device ("Usage Data"). This Usage Data may include information such as your computer's Inte
rnet Protocol address (e.g. IP address), browser type, browser version, the pages of our Service that you visit, the
time and date of your visit, the time spent on those pages, unique device identifiers, and other diagnostic data.
Use of Data
Presight uses the collected data for various purposes:
Consent
As personal information is collected, you will be asked to confirm that your information is correct prior to submitt
ing it to Presight.
Access to Personal Information
Accessing Your Personal Information
You have the right to access all of your personal information that we hold. Through the application, you can correc
t, amend, or append your personal information by logging into the application and navigating to your settings and pr
ofile.
Automated Edit Checks
Presight employs automated edit checks to ensure that data entry fields are completed properly when collecting perso
nal information. These edit checks help maintain data integrity and accuracy. You are encouraged to provide complete
and valid information to ensure the smooth processing of their personal data.

----------Answer from Chatbot----------:
The provided text is a privacy policy for Presight. It details how Presight collects, uses, and discloses customer
and website visitor information, including data security, retention, and user rights regarding their personal inform
ation.

Enter your question (or 'quit' to exit): quit
Thank you for using the QA system!

Streamlit web-based app: https://python-chatbot-finalterm.streamlit.app/

Xây Dựng Chat Bot Cách 2.

Tụi Em xây dựng 1 table content có chỉ mục index sau đó. Xây dựng vectordatabase cho chỉ mục index đó.

Input câu hỏi -> embedding câu hỏi-> dùng cosin so sánh với vectordatabase đưa ra 5 cái chỉ mục gần với câu hỏi nhất -> đưa 4 chỉ mục gần với câu hỏi nhất làm context cho đoạn văn -> đưa vào large language model để có thể trả lời các nội dung liên quan

In [1]:
```
!pip install stop-words
```

```
Collecting stop-words
  Downloading stop-words-2018.7.23.tar.gz (31 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: stop-words
  Building wheel for stop-words (setup.py) ... done
  Created wheel for stop-words: filename=stop_words-2018.7.23-py3-none-any.whl size=32894 sha256=d22e56884ca12fa6352
4c19012d13fa0b6be2e903e5925bd7320d0b2987d7e56
  Stored in directory: /root/.cache/pip/wheels/d0/1a/23/f12552a50cb09bcc1694a5ebb6c2cd5f2a0311de2b8c3d9a89
Successfully built stop-words
Installing collected packages: stop-words
Successfully installed stop-words-2018.7.23
```

In [2]:
```
import requests
from bs4 import BeautifulSoup
import re
from sklearn.feature_extraction.text import ENGLISH_STOP_WORDS
# from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity
import numpy as np
import google.generativeai as genai
import google.generativeai as genai2
import time
from typing import List, Dict, Any
```

1. setup API and main language model.

In [3]:
```
genai.configure(api_key="AIzaSyCUa4BWQfP9HvhViLkCtE1Q5JaDtggBaso")
genai2.configure(api_key="AIzaSyCUa4BWQfP9HvhViLkCtE1Q5JaDtggBaso")
```

2.crawl and processing the data.

In [4]:
```
class TextPreprocessor:
    """
    Performs text preprocessing including lowercase conversion,
    noise removal, word tokenization and stop words removal.
    """
    def __init__(self) -> None:
        self.stop_words = ENGLISH_STOP_WORDS

    def convert_to_lowercase(self, text: str) -> str:
        """Convert all text to lowercase."""
        return text.lower()

    def remove_noise(self, text: str) -> str:
        """
        Remove URLs, email addresses and special characters (keep only letters and whitespace).
        """
        text = re.sub(r'http[s]?://\S+', '', text)
        text = re.sub(r'[^a-zA-Z\s]', '', text)
        return text

    def tokenize(self, text: str) -> List[str]:
        """Split text into words."""
        return text.split()

    def remove_stop_words(self, words: List[str]) -> List[str]:
        """Remove stop words from word list."""
        return [word for word in words if word.lower() not in self.stop_words]

    def preprocess(self, text: str) -> List[str]:
        """Perform complete text preprocessing."""
        text = self.convert_to_lowercase(text)
        text = self.remove_noise(text)
        words = self.tokenize(text)
        words = self.remove_stop_words(words)
        return words
```

In [5]:
```python
def fetch_html_content(url: str) -> str:
    """Download and return HTML content from given URL."""
    response = requests.get(url)
    response.raise_for_status()
    return response.text
```

In [6]:
```python
def parse_html(html_content: str) -> BeautifulSoup:
    """Parse HTML content using BeautifulSoup."""
    return BeautifulSoup(html_content, 'html.parser')
```

In [7]:
```python
def extract_paragraphs_with_headers(soup: BeautifulSoup) -> Dict[int, Dict[str, str]]:
    """
    Extract content and organize it in a dictionary with sequence numbers.
    Returns a dictionary where:
    - Key: sequence number (int)
    - Value: dictionary containing:
        - 'header': h2 heading text
        - 'content': all text content under that heading combined
    """
    content_div = soup.find('div', class_='css-fugq39')
    if not content_div:
        return {}

    content_blocks = content_div.find_all('div', class_='chakra-stack css-o5l3sd')
    organized_content = {}
    sequence_number = 1

    for block in content_blocks:
        header = None
        content = []

        # Find h2 header first
        h2_tag = block.find('h2')
        if h2_tag:
            header = h2_tag.get_text(strip=True)

            # Collect all text content after h2
            for element in block.find_all(['p', 'ul', 'li', 'i']):
                if element.get_text(strip=True):
                    content.append(element.get_text(strip=True))

            # Only add to dictionary if we found both header and content
            if header and content:
                organized_content[sequence_number] = {
                    'header': header,
                    'content': header + " " + ' '.join(content)
                }
                sequence_number += 1

    return organized_content
```

In [8]:
```python
company_html = fetch_html_content("https://www.presight.io/privacy-policy.html")
company_soup = parse_html(company_html)
company_content = extract_paragraphs_with_headers(company_soup)
```

In [9]:
```python
print(company_content)
```

{1: {'header': 'Information Collection and Use', 'content': 'Information Collection and Use We collect several diffe
rent types of information for various purposes to provide and improve our Service to you.'}, 2: {'header': 'Types of
Data Collected', 'content': 'Types of Data Collected Personal Data While using our Service, we may ask you to provid
e us with certain personally identifiable information that can be used to contact or identify you ("Personal Data").
Personally identifiable information may include, but is not limited to: Email addressFirst name and last namePhone n
umberAddress, State, Province, ZIP/Postal code, CityCookies and Usage Data Email address First name and last name Ph
one number Address, State, Province, ZIP/Postal code, City Cookies and Usage Data Usage Data We may also collect inf
ormation that your browser sends whenever you visit our Service or when you access the Service by or through a mobil
e device ("Usage Data"). This Usage Data may include information such as your computer\'s Internet Protocol address
(e.g. IP address), browser type, browser version, the pages of our Service that you visit, the time and date of your
visit, the time spent on those pages, unique device identifiers, and other diagnostic data.'}, 3: {'header': 'Use of
Data', 'content': 'Use of Data Presight uses the collected data for various purposes: To provide and maintain our Se
rviceTo notify you about changes to our ServiceTo allow you to participate in interactive features of our Service wh
en you choose to do soTo provide customer supportTo gather analysis or valuable information so that we can improve o
ur ServiceTo monitor the usage of our ServiceTo detect, prevent and address technical issues To provide and maintain
our Service To notify you about changes to our Service To allow you to participate in interactive features of our Se
rvice when you choose to do so To provide customer support To gather analysis or valuable information so that we can
improve our Service To monitor the usage of our Service To detect, prevent and address technical issues'}, 4: {'head
er': 'Consent', 'content': 'Consent As personal information is collected, you will be asked to confirm that your inf
ormation is correct prior to submitting it to Presight.'}, 5: {'header': 'Access to Personal Information', 'conten
t': 'Access to Personal Information You have the right to access all of your personal information that we hold. Thro
ugh the application, you can correct, amend, or append your personal information by logging into the application and
navigating to your settings and profile. Presight employs automated edit checks to ensure that data entry fields are
completed properly when collecting personal information. These edit checks help maintain data integrity and accurac
y. You are encouraged to provide complete and valid information to ensure the smooth processing of their personal da
ta.'}, 6: {'header': 'Disclosure of Information', 'content': 'Disclosure of Information We may disclose your applica
tion data to third-party service providers who help us provide our services such as Datadog, AWS, Google Cloud and G
oogle Workspace. We may also disclose your information in response to a legal request, such as a subpoena or court o
rder, or to protect our rights or the rights of others.'}, 7: {'header': 'Sharing of Personal Data', 'content': 'Sha
ring of Personal Data Your personal data will not be subject to sharing, transfer, rental or exchange for the benefi
t of third parties, including AI models.'}, 8: {'header': 'Google User Data and Google Workspace APIs', 'content':
'Google User Data and Google Workspace APIs In all cases when users authenticate the platform to Google Workspace, t
he following applies: We do not retain or use Google User Data to develop, improve, or train generalized/non-persona
lized AI and/or ML models.We do not use Google Workspace APIs to develop, improve, or train generalized/non-personal
ized AI and/or ML models.We do not transfer Google User Data to third-party AI tools for the purpose of developing,
improving, or training generalized or non-personalized AI and/or ML models. We do not retain or use Google User Data
to develop, improve, or train generalized/non-personalized AI and/or ML models. We do not use Google Workspace APIs
to develop, improve, or train generalized/non-personalized AI and/or ML models. We do not transfer Google User Data
to third-party AI tools for the purpose of developing, improving, or training generalized or non-personalized AI an
d/or ML models.'}, 9: {'header': 'Data Security', 'content': 'Data Security All data is encrypted both in transit an
d at rest, using industry-standard encryption methods.We regularly perform security audits and vulnerability assessm
ents to ensure the safety of our platform and the data stored within it.Our employees are trained on best practices
for data security, and access to customer data is restricted on a need-to-know basis. All data is encrypted both in
transit and at rest, using industry-standard encryption methods. We regularly perform security audits and vulnerabil
ity assessments to ensure the safety of our platform and the data stored within it. Our employees are trained on bes
t practices for data security, and access to customer data is restricted on a need-to-know basis.'}, 10: {'header':
'Data Retention & Disposal', 'content': 'Data Retention & Disposal Customer data is retained for as long as the acco
unt is in active status. Data enters an â\x80\x9cexpiredâ\x80\x9d state when the account is voluntarily closed. Expi
red account data will be retained for 60 days. After this period, the account and related data will be removed.'}, 1
1: {'header': "Quality, Including Data Subjects' Responsibilities for Quality", 'content': "Quality, Including Data
Subjects' Responsibilities for Quality We are committed to maintaining the quality and accuracy of the personal info
rmation we collect and process.We rely on data subjects to provide accurate and up-to-date information.Data subjects
have the responsibility to inform us of any changes or inaccuracies in their personal data.If you believe that any i
nformation we hold about you is inaccurate, incomplete, or outdated, please contact us promptly to rectify the infor
mation. We are committed to maintaining the quality and accuracy of the personal information we collect and process.
We rely on data subjects to provide accurate and up-to-date information. Data subjects have the responsibility to in
form us of any changes or inaccuracies in their personal data. If you believe that any information we hold about you
is inaccurate, incomplete, or outdated, please contact us promptly to rectify the information."}, 12: {'header': 'Mo
nitoring and Enforcement', 'content': 'Monitoring and Enforcement We regularly monitor its data processing activitie
s to ensure compliance with this privacy policy and applicable data protection laws.In the event of a data breach or
any unauthorized access to your personal information, we will notify you and the appropriate authorities as required
by law.We committed to cooperating with data protection authorities and complying with their advice and decisions re
garding data protection and privacy matters. We regularly monitor its data processing activities to ensure complianc
e with this privacy policy and applicable data protection laws. In the event of a data breach or any unauthorized ac
cess to your personal information, we will notify you and the appropriate authorities as required by law. We committ
ed to cooperating with data protection authorities and complying with their advice and decisions regarding data prot
ection and privacy matters.'}, 13: {'header': 'Cookies', 'content': 'Cookies We use cookies to enhance your experien
ce on our website. You can control the use of cookies through your web browser settings.'}, 14: {'header': 'Third-Pa
rty Websites', 'content': 'Third-Party Websites Our website may contain links to third-party websites. We are not re
sponsible for the privacy practices or content of those websites.'}, 15: {'header': 'Changes to Privacy Policy', 'co
ntent': 'Changes to Privacy Policy We may update this Privacy Policy from time to time. The updated Privacy Policy w
ill be posted on our website.'}, 16: {'header': 'Contact Us', 'content': 'Contact Us If you have any questions about
this Privacy Policy, please contact us through the customer portal or by email atpresight@presight.io.'}, 17: {'head
er': 'Purposeful Use Only', 'content': "Purposeful Use Only We commit to only use personal information for the purpo
ses identified in the entity's privacy policy."}}

3. Transform the data and embedding.

In [10]:
```python
# def create_text_vector(text: str, model: SentenceTransformer) -> np.ndarray:
#     """Create vector embedding from text using SentenceTransformer model."""
#     return model.encode(text)
def create_text_vector(text: str, model: genai.GenerativeModel) -> np.ndarray:
    """Create vector embedding from text using Gemini API."""
```

```
    result = genai.embed_content(
            model="models/text-embedding-004",
            content=text)

    # print(str(result['embedding']))
    return np.array(result['embedding'])
```

In [11]:
```
def build_vector_index(content_dict: Dict[int, Dict[str, str]],  model: genai.GenerativeModel) -> List[Dict[str, Any
    """
    Build vector index from content dictionary.

    Args:
        content_dict: Dictionary with structure {sequence_number: {'header': header_text, 'content': content_text}}
        model: SentenceTransformer model for creating embeddings

    Returns:
        List of dictionaries containing vectors and their corresponding sequence numbers
    """
    preprocessor = TextPreprocessor()
    output_data = []

    for seq_num, data in content_dict.items():
        # Process content text
        processed_content = preprocessor.preprocess(data['content'])
        processed_text = " ".join(processed_content)
        vector = create_text_vector(processed_text, model)

        # Store vector with sequence number for reference
        output_data.append({
            "vector": vector,
            "seq_num": seq_num,
            "header": data['header']  # Optionally store header for reference
        })

    return output_data
```

In [12]:
```
# tạo một vector database
vector_indexs = build_vector_index(company_content, genai)
```

In [13]:
```
def search_top_matches(
    question: str,
    index: List[Dict[str, Any]],
     model: genai.GenerativeModel,
    k: int = 5
) -> List[Dict[str, Any]]:
    """
    Search for top k sections most relevant to the question.
    Returns list of matches with their sequence numbers, headers, and similarity scores.
    """
    preprocessor = TextPreprocessor()
    processed_question = preprocessor.preprocess(question)
    question_vector = create_text_vector(" ".join(processed_question), genai)

    similarity_results = []
    for index_entry in index:
        index_vector = index_entry["vector"]
        # Ensure index_vector has correct dimensions
        if index_vector.ndim == 2:
            index_vector = index_vector.reshape(index_vector.shape[1],)

        similarity = cosine_similarity([question_vector], [index_vector])[0][0]
        similarity_results.append({
            "seq_num": index_entry["seq_num"],
            "header": index_entry["header"],
            "score": similarity
        })

    similarity_results.sort(key=lambda x: x["score"], reverse=True)
    return similarity_results[:k]
```

In [27]:
```
search_top_matches("what is company policy", vector_indexs, genai)
```

Out[27]:
```
[{'seq_num': 15,
  'header': 'Changes to Privacy Policy',
  'score': 0.5005112826787221},
 {'seq_num': 14,
  'header': 'Third-Party Websites',
  'score': 0.4573274012901565},
 {'seq_num': 9, 'header': 'Data Security', 'score': 0.4483942994885045},
 {'seq_num': 12,
  'header': 'Monitoring and Enforcement',
  'score': 0.43348876324873864},
 {'seq_num': 11,
  'header': "Quality, Including Data Subjects' Responsibilities for Quality",
  'score': 0.42357830858746603}]
```

4. Answer the question

In [42]:
```python
def answer_question(
    question: str,
    content_dict: Dict[int, Dict[str, str]],
    model: genai.GenerativeModel,
    model2: genai.GenerativeModel
) -> str:
    """Use language model to answer question based on content dictionary."""
    vector_index = build_vector_index(content_dict, model)
    top_matches = search_top_matches(question, vector_index, model)

    if not top_matches:
        return "No relevant information found."

    # Get the most relevant content
    matching_content = content_dict[top_matches[0]['seq_num']]['content'] + content_dict[top_matches[1]['seq_num']][
    # print(matching_content)

    prompt = f"Based on the following this context -- {matching_content}. --  fully become a  agent assistant  of th

    try:
        response = model2.generate_content(prompt)
        print("------------------------------------------------------")
        print("the question:", question)
        print("Assistent Answer:", response.text)
        return
        # return response.text
    except Exception as e:
        return f"Error: {e}"
```

In [43]:
```python
# setting config gemini chat.
generation_config = {
  "temperature": 0,
  "top_p": 0.95,
  "top_k": 40,
  "max_output_tokens": 8192,
  "response_mime_type": "text/plain",
}
model2 = genai2.GenerativeModel(
  model_name="gemini-2.0-flash-exp",
  generation_config=generation_config,
)
```

In [44]:
```python
answer_question("what is company policy", company_content, genai, model2)
```

```
------------------------------------------------------
the question: what is company policy
Assistent Answer: The company's policy includes:

*   **Privacy Policy Updates:** The Privacy Policy may be updated, with changes posted on the website.
*   **Third-Party Links:** The website may contain links to third-party websites, for which the company is not respo
nsible for privacy practices or content.
*   **Data Security:** Data is encrypted in transit and at rest using industry-standard methods. Regular security au
dits and vulnerability assessments are performed. Employee access to customer data is restricted, and employees are
trained on data security best practices.
*   **Monitoring and Enforcement:** Data processing activities are regularly monitored for compliance with the priva
cy policy and applicable laws. In the event of a data breach, affected users and authorities will be notified as req
uired by law. The company is committed to cooperating with data protection authorities.

Please ask if you have more questions about the company.
```

In [46]:
```python
answer_question("what is data use for?", company_content, genai, model2)
```

```
------------------------------------------------------------
the question: what is data use for?
Assistent Answer: Data is used to:

*   Provide and maintain the Service.
*   Notify users about changes to the Service.
*   Enable user participation in interactive features.
*   Provide customer support.
*   Gather analysis to improve the Service.
*   Monitor Service usage.
*   Detect, prevent, and address technical issues.

Please ask if you have more questions about the company.
```