```
pip install sentence-transformers
```

```
  Downloading sentence_transformers-2.7.0-py3-none-any.whl (171 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 171.5/171.5 kB 1.2 MB/s eta 0:00:00
Requirement already satisfied: transformers<5.0.0,>=4.34.0 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (4.40.1)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (4.66.2)
Requirement already satisfied: torch>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (2.2.1+cu121)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (1.25.2)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (1.2.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (1.11.4)
Requirement already satisfied: huggingface-hub>=0.15.1 in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (0.20.3)
Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from sentence-transformers) (9.4.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-transformers) (3.14
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-transformer
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-transformers) (2.31
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-transformers) (6
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-t
Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.15.1->sentence-transformers
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (1.12)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (3.3)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (3.1.3)
Collecting nvidia-cuda-nvrtc-cu12==12.1.105 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cuda_nvrtc_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (23.7 MB)
Collecting nvidia-cuda-runtime-cu12==12.1.105 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cuda_runtime_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (823 kB)
Collecting nvidia-cuda-cupti-cu12==12.1.105 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cuda_cupti_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (14.1 MB)
Collecting nvidia-cudnn-cu12==8.9.2.26 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cudnn_cu12-8.9.2.26-py3-none-manylinux1_x86_64.whl (731.7 MB)
Collecting nvidia-cublas-cu12==12.1.3.1 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cublas_cu12-12.1.3.1-py3-none-manylinux1_x86_64.whl (410.6 MB)
Collecting nvidia-cufft-cu12==11.0.2.54 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cufft_cu12-11.0.2.54-py3-none-manylinux1_x86_64.whl (121.6 MB)
Collecting nvidia-curand-cu12==10.3.2.106 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_curand_cu12-10.3.2.106-py3-none-manylinux1_x86_64.whl (56.5 MB)
Collecting nvidia-cusolver-cu12==11.4.5.107 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cusolver_cu12-11.4.5.107-py3-none-manylinux1_x86_64.whl (124.2 MB)
Collecting nvidia-cusparse-cu12==12.1.0.106 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_cusparse_cu12-12.1.0.106-py3-none-manylinux1_x86_64.whl (196.0 MB)
Collecting nvidia-nccl-cu12==2.19.3 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_nccl_cu12-2.19.3-py3-none-manylinux1_x86_64.whl (166.0 MB)
Collecting nvidia-nvtx-cu12==12.1.105 (from torch>=1.11.0->sentence-transformers)
  Using cached nvidia_nvtx_cu12-12.1.105-py3-none-manylinux1_x86_64.whl (99 kB)
Requirement already satisfied: triton==2.2.0 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence-transformers) (2.2.0)
Collecting nvidia-nvjitlink-cu12 (from nvidia-cusolver-cu12==11.4.5.107->torch>=1.11.0->sentence-transformers)
  Using cached nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.34.0->sentence-transf
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.34.0->sentence-t
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.34.0->sentence-trans
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from scikit-learn->sentence-transformers) (1.4.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn->sentence-transformers) (3.5
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.11.0->sentence-transformers)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.15.1->se
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.15.1->sentence-trans
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.15.1->sentence
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.15.1->sentence
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-packages (from sympy->torch>=1.11.0->sentence-transformers) (1.3
Installing collected packages: nvidia-nvtx-cu12, nvidia-nvjitlink-cu12, nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cuda-r
Successfully installed nvidia-cublas-cu12-12.1.3.1 nvidia-cuda-cupti-cu12-12.1.105 nvidia-cuda-nvrtc-cu12-12.1.105 nvidia-cuda-runtime-cu12-12
```

```
pip install wordcloud
```

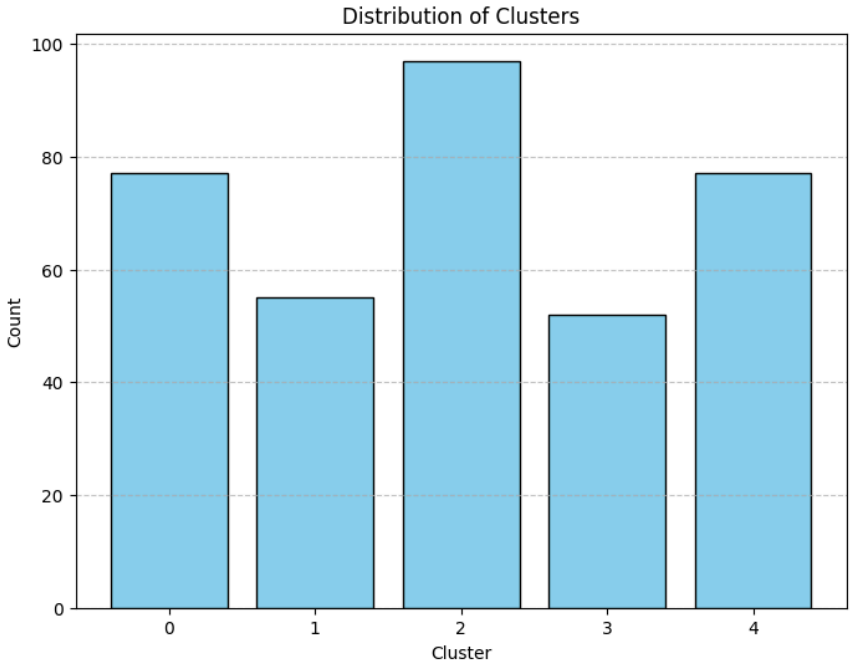
```
Requirement already satisfied: wordcloud in /usr/local/lib/python3.10/dist-packages (1.9.3)
Requirement already satisfied: numpy>=1.6.1 in /usr/local/lib/python3.10/dist-packages (from wordcloud) (1.25.2)
Requirement already satisfied: pillow in /usr/local/lib/python3.10/dist-packages (from wordcloud) (9.4.0)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.10/dist-packages (from wordcloud) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (1.2.1)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (0.12.1)
Requirement already satisfied: fonttools>=4.22.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (4.51.0)
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (1.4.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (24.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (3.1.2)
Requirement already satisfied: python-dateutil>=2.7 in /usr/local/lib/python3.10/dist-packages (from matplotlib->wordcloud) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.7->matplotlib->wordcloud) (1.16.0)
```

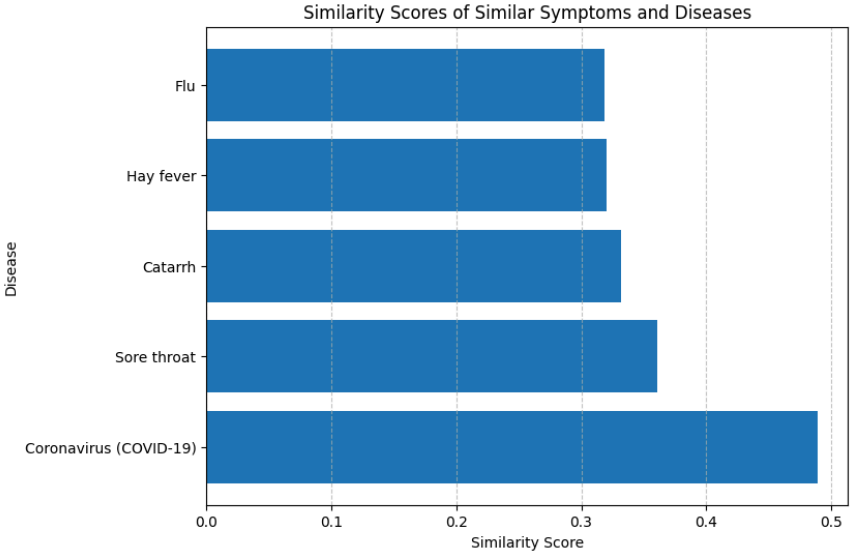
```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.cluster import KMeans
from sklearn.ensemble import IsolationForest
from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity
from transformers import BartTokenizer, BartForConditionalGeneration

# load dateset
data = pd.read_excel('/dia...dataset(2).xlsx')
```

```
data = pd.read_excel('dis dataset(5).xlsx')

# data pre-processing
tfidf_vectorizer = TfidfVectorizer(max_features=1500)
tfidf_features = tfidf_vectorizer.fit_transform(data['Symptom']).toarray()

model = SentenceTransformer('all-MiniLM-L6-v2')
bert_embeddings = model.encode(data['Symptom'].tolist())

# combination feature
features = np.hstack((tfidf_features, bert_embeddings))

# Clustering
kmeans = KMeans(n_clusters=5, random_state=42)
clusters = kmeans.fit_predict(features)
data['Cluster'] = clusters

# anomaly detection
iso_forest = IsolationForest(random_state=42)
anomalies = iso_forest.fit_predict(features)
data['Anomaly'] = anomalies == -1

# Load BART model and word divider
bart_model = BartForConditionalGeneration.from_pretrained('facebook/bart-large-cnn')
bart_tokenizer = BartTokenizer.from_pretrained('facebook/bart-large-cnn')

# Similarity search function (Look for the 5 most relevant diseases)
def find_similar_symptoms(input_symptom, n_results=5):
    input_embedding = model.encode([input_symptom])[0]
    input_feature = np.hstack((tfidf_vectorizer.transform([input_symptom]).toarray()[0], input_embedding))
    similarity_scores = cosine_similarity([input_feature], features)[0]
    top_indexes = np.argsort(similarity_scores)[-n_results:]
    return data.iloc[top_indexes][['Symptom', 'Disease name', 'Cluster', 'Anomaly', 'Treatment recommendation']], similarity_scores[top_in

# User input symptom
query_symptom = input("Please enter a description of your symptoms:")
similar_symptoms, similarity_scores = find_similar_symptoms(query_symptom)

# Output similar symptoms and diseases and their similarity
print("Five diseases with symptoms similar to those you describe and how similar they are: ")
print(similar_symptoms[['Disease name']])
print("Similarity scores ranked from low to high:", similarity_scores)

# Output the most similar symptoms and diseases
highest_similarity_index = np.argmax(similarity_scores)
highest_similarity_symptom = similar_symptoms.iloc[highest_similarity_index]
highest_similarity_score = similarity_scores[highest_similarity_index]
highest_similarity_score_percent = highest_similarity_score * 100    # 转换为百分比
print('The disease most similar to the one you described and its probability:', highest_similarity_symptom['Disease name'], f'{highest_s

# Summarize treatment recommendations for diseases with the highest similarity
treatment_text = highest_similarity_symptom['Treatment recommendation']

# segment handling
paragraphs = treatment_text.split("\n")
input_text = ""
output_text = ""
for paragraph in paragraphs:
    if len(input_text) + len(paragraph) < 1024:
        input_text += paragraph + "\n"
    else:
        print("The unsummarized sentence is:", input_text)
        inputs = bart_tokenizer(input_text, return_tensors="pt", max_length=1024, truncation=True)
        summary_ids = bart_model.generate(inputs['input_ids'], num_beams=5, early_stopping=True)
        output_text += bart_tokenizer.decode(summary_ids[0], skip_special_tokens=True) + "\n"
        input_text = paragraph + "\n"

if input_text:
    print("The unsummarized sentence is:", input_text)    # Process the remaining paragraphs
    inputs = bart_tokenizer(input_text, return_tensors="pt", max_length=1024, truncation=True)
    summary_ids = bart_model.generate(inputs['input_ids'], num_beams=7, early_stopping=True)
    output_text += bart_tokenizer.decode(summary_ids[0], skip_special_tokens=True)

print("Summary of treatment recommendations:", output_text)

print("\nData clustering and anomaly detection results:")
print(data[['Disease name', 'Cluster', 'Anomaly']])
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10 to 'aut
  warnings.warn(
config.json: 100%                                                 1.58k/1.58k [00:00<00:00, 61.4kB/s]
model.safetensors: 100%                                           1.63G/1.63G [00:19<00:00, 76.0MB/s]
generation_config.json: 100%                                      363/363 [00:00<00:00, 14.5kB/s]
vocab.json: 100%                                                  899k/899k [00:00<00:00, 13.9MB/s]
merges.txt: 100%                                                  456k/456k [00:00<00:00, 17.9MB/s]
tokenizer.json: 100%                                              1.36M/1.36M [00:00<00:00, 33.0MB/s]
Please enter a description of your symptoms:I don't feel well and can't go to work or do normal activities. I have a slight fever. I have been c
Five diseases with symptoms similar to those you describe and how similar they are:
              Disease name
124                    Flu
144              Hay fever
56                 Catarrh
282            Sore throat
82   Coronavirus (COVID-19)
Similarity scores ranked from low to high: [0.31844361 0.32004693 0.33191397 0.36058827 0.4888533 ]
The disease most similar to the one you described and its probability: Coronavirus (COVID-19) 48.89%
The unsummarized sentence is: drink fluids like water to keep yourself hydrated
get plenty of rest
wear loose, comfortable clothing - don't try to make yourself too cold
take over-the-counter medications like paracetamol - always follow the manufacturer's instructions

Summary of treatment recommendations: drink fluids like water to keep yourself hydrated and get plenty of rest.wear loose, comfortable clothing

Data clustering and anomaly detection results:
                            Disease name  Cluster  Anomaly
0                Abdominal aortic aneurysm        3    False
1                                     Acne        2    False
2                    Acute cholecystitis        3    False
3            Acute lymphoblastic leukaemia        2    False
4   Acute lymphoblastic leukaemia: Children       2    False
..                                    ...      ...      ...
353                             Leukemia        2    False
354                          oral cancer        2    False
355          Zika Virus Infection (Zika)        2    False
356                             Yersenia        3    False
357                        West Nile Virus       2    False
```

```
import matplotlib.pyplot as plt


plt.figure(figsize=(8, 6))
plt.hist(data['Cluster'], bins=range(6), align='left', rwidth=0.8, color='skyblue', edgecolor='black')
plt.xlabel('Cluster')
plt.ylabel('Count')
plt.title('Distribution of Clusters')
plt.xticks(range(5))
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()



plt.figure(figsize=(8, 6))
plt.barh(similar_symptoms['Disease name'], similarity_scores)
plt.xlabel('Similarity Score')
plt.ylabel('Disease')
plt.title('Similarity Scores of Similar Symptoms and Diseases')
plt.gca().invert_yaxis()
plt.grid(axis='x', linestyle='--', alpha=0.7)
plt.show()

from wordcloud import WordCloud


wordcloud = WordCloud(width=800, height=400, background_color='white').generate(output_text)


plt.figure(figsize=(10, 8))
plt.imshow(wordcloud, interpolation='bilinear')
plt.title('Word Cloud for Treatment Recommendation Summary')
plt.axis('off')
plt.show()
```


Distribution of Clusters
Count
Cluster
0
1
2
3
4
0
20
40
60
80
100


Similarity Scores of Similar Symptoms and Diseases
Disease
Similarity Score
Flu
Hay fever
Catarrh
Sore throat
Coronavirus (COVID-19)
0.0
0.1
0.2
0.3
0.4
0.5


Word Cloud for Treatment Recommendation Summary
fluids
water
drink
keep
plenty
hydrated
take
make
don
cold
instructions
follow
try
counter
t
s
medications
manufacturer
always
paracetamol
clothing
loose
rest
comfortable
wear