The logic is this:

- Take a sentence, convert it into a vector.
- Take many other sentences, and convert them into vectors.
- Find sentences that have the smallest distance (Euclidean) or smallest angle (cosine similarity) between them — more on that here.
- We now have a measure of semantic similarity between sentences — easy!

For the first set of use case JPETSTORE

```
pip install sentence-transformers
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wh
Collecting sentence-transformers
  Downloading sentence-transformers-2.2.2.tar.gz (85 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 86.0/86.0 kB 2.0 MB/s eta
  Preparing metadata (setup.py) ... done
Collecting transformers<5.0.0,>=4.6.0
  Downloading transformers-4.27.4-py3-none-any.whl (6.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 6.8/6.8 MB 43.1 MB/s eta 0
Requirement already satisfied: tqdm in /usr/local/lib/python3.9/dist-packages (
Requirement already satisfied: torch>=1.6.0 in /usr/local/lib/python3.9/dist-pa
Requirement already satisfied: torchvision in /usr/local/lib/python3.9/dist-pac
Requirement already satisfied: numpy in /usr/local/lib/python3.9/dist-packages
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.9/dist-pa
Requirement already satisfied: scipy in /usr/local/lib/python3.9/dist-packages
Requirement already satisfied: nltk in /usr/local/lib/python3.9/dist-packages (
Collecting sentencepiece
  Downloading sentencepiece-0.1.97-cp39-cp39-manylinux_2_17_x86_64.manylinux201
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.3/1.3 MB 30.6 MB/s eta 0
Collecting huggingface-hub>=0.4.0
  Downloading huggingface_hub-0.13.4-py3-none-any.whl (200 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 200.1/200.1 kB 15.8 MB/s eta
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/pyt
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.9/dist-pac
Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.9/dist
Requirement already satisfied: requests in /usr/local/lib/python3.9/dist-packag
Requirement already satisfied: filelock in /usr/local/lib/python3.9/dist-packag
Requirement already satisfied: sympy in /usr/local/lib/python3.9/dist-packages
Requirement already satisfied: triton==2.0.0 in /usr/local/lib/python3.9/dist-p
Requirement already satisfied: jinja2 in /usr/local/lib/python3.9/dist-packages
Requirement already satisfied: networkx in /usr/local/lib/python3.9/dist-packag
Requirement already satisfied: lit in /usr/local/lib/python3.9/dist-packages (f
Requirement already satisfied: cmake in /usr/local/lib/python3.9/dist-packages
Collecting tokenizers!=0.11.3,<0.14,>=0.11.1
  Downloading tokenizers-0.13.3-cp39-cp39-manylinux_2_17_x86_64.manylinux2014_x
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 7.8/7.8 MB 90.7 MB/s eta 0
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.9/di
Requirement already satisfied: joblib in /usr/local/lib/python3.9/dist-packages
Requirement already satisfied: click in /usr/local/lib/python3.9/dist-packages
```

```
      Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.9
      Requirement already satisfied: pillow!=8.3.*,>=5.3.0 in /usr/local/lib/python3.
      Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.9/dist
      Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.9/dist-pa
      Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/pyth
      Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.9/d
      Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.
      Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.9/dist-pa
      Building wheels for collected packages: sentence-transformers
        Building wheel for sentence-transformers (setup.py) ... done
        Created wheel for sentence-transformers: filename=sentence_transformers-2.2.2
        Stored in directory: /root/.cache/pip/wheels/71/67/06/162a3760c40d74dd40bc855
      Successfully built sentence-transformers
      Installing collected packages: tokenizers, sentencepiece, huggingface-hub, tran
      Successfully installed huggingface-hub-0.13.4 sentence-transformers-2.2.2 sente
```

```
sentences = [
    "Browsing the Product List",
    "Browsing the Catalog",
    "Searching the Catalog",
    "Browsing the items List",
    "Add  cart Items",
    "Update items in the Cart",
    "Remove items from cart",
    "View Cart Items",
    "Make payment",
    "List Order Items",
    "View Order Status",
    "Confirm Order",
    "Get Total amount",
    "Change Shipping Info ",
    "Signing Up",
    "Signing In",
    "Signout",
    "Manage Item",
    "Manage Account",
    "Manage Order",
    "Manage product/category"
]
```

```
from sentence_transformers import SentenceTransformer

model = SentenceTransformer('bert-base-nli-mean-tokens')
```

Downloading (…)821d1/.gitattributes: 100% 391/391 [00:00<00:00, 9.26kB/s]

Downloading (…)_Pooling/config.json: 100% 190/190 [00:00<00:00, 4.85kB/s]

Downloading (…)8d01e821d1/README.md: 100% 3.95k/3.95k [00:00<00:00, 166kB/s]

Downloading (…)d1/added_tokens.json: 100% 2.00/2.00 [00:00<00:00, 73.2B/s]

Downloading (…)01e821d1/config.json: 100% 625/625 [00:00<00:00, 20.0kB/s]

Downloading (…)ce_transformers.json: 100% 122/122 [00:00<00:00, 4.91kB/s]

Downloading pytorch_model.bin: 100% 438M/438M [00:08<00:00, 55.9MB/s]

Downloading (…)nce_bert_config.json: 100% 53.0/53.0 [00:00<00:00, 1.84kB/s]

Downloading (…)cial_tokens_map.json: 100% 112/112 [00:00<00:00, 4.19kB/s]

Downloading (…)821d1/tokenizer.json: 100% 466k/466k [00:00<00:00, 5.08MB/s]

Downloading (…)okenizer_config.json: 100% 399/399 [00:00<00:00, 13.7kB/s]

Downloading (…)8d01e821d1/vocab.txt: 100% 232k/232k [00:00<00:00, 3.15MB/s]

Downloading (…)1e821d1/modules.json: 100% 229/229 [00:00<00:00, 11.2kB/s]

```
sentence_embeddings = model.encode(sentences)
```

```
sentence_embeddings.shape
```

```
      (21, 768)
```

```
from sklearn.metrics.pairwise import cosine_similarity


Dic={}
outer={}

for x in range (0,len(sentences)):
  similarity = cosine_similarity(
    [sentence_embeddings[x]],
    sentence_embeddings[0:]
)

  print("for the sentence : ",sentences[x])
  print()
  for y in range (0,len(sentences)):
      Dic[sentences[y]]=similarity[0][y]
  outer[sentences[x]]=Dic
  sorted_data = sorted(Dic.items(), key = lambda kv: kv[1],reverse=True)
  # print(outer)
  for z in range (0, len(sorted_data)):
    print(sorted_data[z])
  print()
  print()
```

```
('Browsing the items List', 0.63043207)
('Update items in the Cart', 0.6282257)
('Browsing the Product List', 0.5662687)
('Remove items from cart', 0.48557103)
('Browsing the Catalog', 0.44406214)
('Searching the Catalog', 0.43231627)


for the sentence :  Manage product/category

('Manage product/category', 1.0000001)
('Manage Item', 0.8838122)
('Manage Account', 0.7950597)
('Manage Order', 0.7899674)
('View Order Status', 0.78056365)
('List Order Items', 0.7681759)
('Browsing the Product List', 0.7309641)
('Signing Up', 0.7144836)
('Make payment', 0.7043941)
('Browsing the items List', 0.70267475)
('Confirm Order', 0.69539785)
('View Cart Items', 0.682162)
('Signing In', 0.6544866)
('Signout', 0.64798605)
('Update items in the Cart', 0.6409887)
('Add  cart Items', 0.63894653)
('Change Shipping Info ', 0.636227)
('Get Total amount', 0.62367815)
('Searching the Catalog', 0.5176725)
('Browsing the Catalog', 0.50912994)
('Remove items from cart', 0.4941065)
```

EXPECTED CATEGORY:

1. CART
2. CATALOG
3. ORDER
4. ACCOUNT
5. MANAGE

for the second use case (TFWA)

```
sentences1=[
"Login user",
"Forgot password",
"Change Password",
"Students can View Papers for Feedback",
"Students can Record Teacher's Feedback using online portal",
"Students can Record Teacher's Feedback using Chatbot",
```

```
"View Feedback by TIC",
"View Teacher_Subject Feedback Data by TIC",
"View Class Wise Feedback Data by TIC",
"View All Teacher Feedback Data by TIC",
"View Class Feedback Status by TIC",
"View Department-wise Feedback by pricipal",
"View Department_Class Feedback Data by pricipal",
"View Department_Teacher Feedback by pricipal",
"View Department_Subject Feedback Data by pricipal",
"View Top five Teachers by pricipal",
"Manage Department",
"Manage Course",
"Manage Papers",
"Manage User Role",
"Manage Question template",
"Manage users",
"Manage Feedback"

]


sentence_embeddings1 = model.encode(sentences1)


sentence_embeddings1.shape
```

```
(23, 768)
```

```
Dic1={}
outer1={}

for x in range (0,len(sentences1)):
  similarity1 = cosine_similarity(
    [sentence_embeddings1[x]],
    sentence_embeddings1[0:]
)

  print("for the sentence : ",sentences1[x])
  print()
  for y in range (0,len(sentences1)):
      Dic1[sentences1[y]]=similarity1[0][y]
  outer1[sentences1[x]]=Dic1
  sorted_data1 = sorted(Dic1.items(), key = lambda kv: kv[1],reverse=True)
  # print(outer)
  for z in range (0, len(sorted_data1)):
    print(sorted_data1[z])
  print()
  print()
```

```
('View Department_Class Feedback Data by pricipal', 0.82956016)
('View Class Feedback Status by TIC', 0.81189144)
('View Class Wise Feedback Data by TIC', 0.798061)
('View Department-wise Feedback by pricipal', 0.79544204)
('View Department_Subject Feedback Data by pricipal', 0.78639734)
("Students can Record Teacher's Feedback using online portal", 0.76012665)
("Students can Record Teacher's Feedback using Chatbot", 0.7592676)
('View All Teacher Feedback Data by TIC', 0.71303463)
('Students can View Papers for Feedback', 0.64940405)
('Manage Question template', 0.6414368)
('Login user', 0.63574135)
('Manage User Role', 0.61263084)
('Manage Feedback', 0.5993335)
('View Feedback by TIC', 0.5903765)
('Manage Department', 0.58618367)
('Manage users', 0.57827973)
('View Top five Teachers by pricipal', 0.5147963)
('Change Password', 0.4942721)
('Manage Papers', 0.46989098)
('Manage Course', 0.39877197)
('Forgot password', 0.2871524)


for the sentence :  View Department_Subject Feedback Data by pricipal

('View Department_Subject Feedback Data by pricipal', 0.99999994)
('View Department_Class Feedback Data by pricipal', 0.9357836)
('View Department-wise Feedback by pricipal', 0.9039457)
('View Department_Teacher Feedback by pricipal', 0.78639734)
('Manage Question template', 0.78380126)
('Manage User Role', 0.7835289)
('View Class Feedback Status by TIC', 0.7834054)
('View Class Wise Feedback Data by TIC', 0.7824395)
('View Teacher_Subject Feedback Data by TIC', 0.78074014)
('Manage users', 0.76876533)
('Manage Department', 0.76567537)
('Login user', 0.7595122)
('Manage Feedback', 0.7366221)
('View Feedback by TIC', 0.7182591)
('Change Password', 0.6253759)
('Manage Course', 0.5950766)
('Manage Papers', 0.58669806)
('View All Teacher Feedback Data by TIC', 0.56695837)
("Students can Record Teacher's Feedback using online portal", 0.5305013)
("Students can Record Teacher's Feedback using Chatbot", 0.50555795)
('Students can View Papers for Feedback', 0.49276942)
('View Top five Teachers by pricipal', 0.39743066)
('Forgot password', 0.34493905)
```

Expected category:

1. Authentication
2. Analytics

3. Feedback
4. Manage

** flight apllication usecases**

```
sentences2=[
    "Create a session",
    "Remove a session",
    "Invalidate a session",
    "Get session details",
    "AddNew Flight",
    "Drop Flights",
    "Update Flight",
    "Store Flight Segment",
    "Search flight",
    "Search flight by FlightId",
    "Search Flight by Airportand ",
    "Departure Date",
    "Search Flight by Airport ",
    "Manage Airport Details",
    "Customer can Book a flight",
    "Customer can View a booking",
    "Customer can Cancel a Booking",
    "Customer can View all the booking",
    "Register a customer",
    "update customer details",
    "Drop Customers Details",
    "Get customer details",
    "Manage Customer details"
]


sentence_embeddings2 = model.encode(sentences2)


sentence_embeddings2.shape

      (23, 768)


######

Dic2={}
outer2={}
5ww7623327
for x in range (0,len(sentences2)):
  similarity2 = cosine_similarity(
    [sentence_embeddings2[x]],
    sentence_embeddings2[0:]
)

  print("for the sentence : ",sentences2[x])
  print()
  for y in range (0,len(sentences2)):
      Dic2[sentences2[y]]=similarity2[0][y]
  outer2[sentences2[x]]=Dic2
  sorted_data2 = sorted(Dic2.items(), key = lambda kv: kv[1],reverse=True)
```

```
# print(outer)
for z in range (0, len(sorted_data2)):
  print(sorted_data2[z])
print()
print()
```

```
    for the sentence :  Create a session

    ('Create a session', 0.99999994)
    ('Get session details', 0.87540096)
    ('Manage Customer details', 0.6850364)
    ('Get customer details', 0.6800939)
    ('update customer details', 0.66372716)
    ('Register a customer', 0.6281044)
    ('Customer can View a booking', 0.60733056)
    ('AddNew Flight', 0.6033235)
    ('Departure Date', 0.57390034)
    ('Update Flight', 0.56623775)
    ('Remove a session', 0.55081964)
    ('Store Flight Segment', 0.53024715)
    ('Invalidate a session', 0.49191973)
    ('Manage Airport Details', 0.46891737)
    ('Search flight', 0.46649396)
    ('Customer can View all the booking', 0.4479288)
    ('Customer can Cancel a Booking', 0.43001363)
    ('Drop Customers Details', 0.36784807)
    ('Customer can Book a flight', 0.35784835)
    ('Search flight by FlightId', 0.34311315)
    ('Search Flight by Airportand ', 0.2987969)
    ('Drop Flights', 0.29568282)
    ('Search Flight by Airport ', 0.25287297)


    for the sentence :  Remove a session

    ('Remove a session', 1.0)
    ('Invalidate a session', 0.9382962)
    ('Drop Customers Details', 0.8498621)
    ('Customer can Cancel a Booking', 0.73000765)
    ('Drop Flights', 0.7186761)
    ('Departure Date', 0.6451287)
    ('Create a session', 0.55081964)
    ('Get session details', 0.5155742)
    ('Store Flight Segment', 0.49902213)
    ('Search flight', 0.4971367)
    ('AddNew Flight', 0.48838183)
    ('Register a customer', 0.48200127)
    ('Update Flight', 0.4536746)
    ('Manage Customer details', 0.42899337)
    ('Get customer details', 0.41147214)
    ('update customer details', 0.40177268)
    ('Customer can View a booking', 0.3444426)
    ('Manage Airport Details', 0.3411491)
    ('Search Flight by Airportand ', 0.32728508)
    ('Search flight by FlightId', 0.3234243)
    ('Search Flight by Airport ', 0.28914216)
```

```
('Customer can Book a flight', 0.23631492)
('Customer can View all the booking', 0.1649265)


for the sentence :  Invalidate a session

('Invalidate a session', 1.0)
('Remove a session', 0.0282062)
```

Mam from the above results in each case we are getting atleast top-2 usecases are from the similar microservices usecasses. but the result is not true for the use-case named departure date which should be belong to micro-services named flight but it giving random result