```
!nvidia-smi
```

```
Sat Dec  7 22:29:19 2024
+---------------------------------------------------------------------------------------+
| NVIDIA-SMI 535.104.05             Driver Version: 535.104.05   CUDA Version: 12.2     |
|-----------------------------------------+----------------------+----------------------+
| GPU  Name                 Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp   Perf          Pwr:Usage/Cap |         Memory-Usage | GPU-Util  Compute M. |
|                                         |                      |               MIG M. |
|=========================================+======================+======================|
|   0  Tesla T4                       Off | 00000000:00:04.0 Off |                    0 |
| N/A   65C    P8              12W /  70W |      0MiB / 15360MiB |      0%      Default |
|                                         |                      |                  N/A |
+-----------------------------------------+----------------------+----------------------+

+---------------------------------------------------------------------------------------+
| Processes:                                                                            |
|  GPU   GI   CI        PID   Type   Process name                            GPU Memory |
|        ID   ID                                                             Usage      |
|=======================================================================================|
|  No running processes found                                                           |
+---------------------------------------------------------------------------------------+
```

```
import tensorflow as tf
print("GPU Available:", tf.config.list_physical_devices('GPU'))
```

```
GPU Available: [PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]
```

```
import torch
print("Is CUDA available:", torch.cuda.is_available())
```

```
Is CUDA available: True
```

```
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
# model = model.to(device)
# data = data.to(device)
```

```
# Example model setup (replace this with your actual model)
from transformers import AutoModel, AutoTokenizer

model_name = "bert-base-uncased"  # Example: Replace with your model
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModel.from_pretrained(model_name)
model = model.to(device)
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_auth.py:94: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
tokenizer_config.json: 100%  48.0/48.0 [00:00<00:00, 969B/s]
config.json: 100%  570/570 [00:00<00:00, 9.19kB/s]
vocab.txt: 100%  232k/232k [00:00<00:00, 5.97MB/s]
tokenizer.json: 100%  466k/466k [00:00<00:00, 9.85MB/s]
model.safetensors: 100%  440M/440M [00:01<00:00, 244MB/s]
```

```
!pip install langchain
```

```
Requirement already satisfied: langchain in /usr/local/lib/python3.10/dist-packages (0.3.9)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain) (3.11.9)
Requirement already satisfied: async-timeout<5.0.0,>=4.0.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (4.0.3)
Requirement already satisfied: langchain-core<0.4.0,>=0.3.21 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.3.21)
Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from langchain)
Requirement already satisfied: langsmith<0.2.0,>=0.1.17 in /usr/local/lib/python3.10/dist-packages (from langchain) (0.1.147)
```

```
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (1.26.4)
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.10.3)
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-packages (from langchain) (2.32.3)
Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain) (9.0.0)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->lang
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain) (24
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchai
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain)
Requirement already satisfied: jsonpatch<2.0,>=1.33 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.21-
Requirement already satisfied: packaging<25,>=23.2 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.21-
Requirement already satisfied: typing-extensions>=4.7 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.2
Requirement already satisfied: httpx<1,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.17->langchai
Requirement already satisfied: orjson<4.0.0,>=3.9.14 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.17->lan
Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->lang
Requirement already satisfied: pydantic-core==2.27.1 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langc
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (2.2.
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain) (2024
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain) (3.
Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>=0.1.17->
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>=0
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.10/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->la
Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-co
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.2
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0
```

```
!pip install langchain_community
```

```
Collecting langchain_community
  Downloading langchain_community-0.3.10-py3-none-any.whl.metadata (2.9 kB)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (3.11.
Collecting dataclasses-json<0.7,>=0.5.7 (from langchain_community)
  Downloading dataclasses_json-0.6.7-py3-none-any.whl.metadata (25 kB)
Collecting httpx-sse<0.5.0,>=0.4.0 (from langchain_community)
  Downloading httpx_sse-0.4.0-py3-none-any.whl.metadata (9.0 kB)
Collecting langchain<0.4.0,>=0.3.10 (from langchain_community)
  Downloading langchain-0.3.10-py3-none-any.whl.metadata (7.1 kB)
Collecting langchain-core<0.4.0,>=0.3.22 (from langchain_community)
  Downloading langchain_core-0.3.22-py3-none-any.whl.metadata (6.3 kB)
Requirement already satisfied: langsmith<0.2.0,>=0.1.125 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (0
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (1.26.4)
Collecting pydantic-settings<3.0.0,>=2.4.0 (from langchain_community)
  Downloading pydantic_settings-2.6.1-py3-none-any.whl.metadata (3.5 kB)
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-packages (from langchain_community) (2.32.3)
Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain_community)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->la
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain
Requirement already satisfied: async-timeout<6.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->la
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain_co
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchai
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langch
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchai
Collecting marshmallow<4.0.0,>=3.18.0 (from dataclasses-json<0.7,>=0.5.7->langchain_community)
  Downloading marshmallow-3.23.1-py3-none-any.whl.metadata (7.5 kB)
Collecting typing-inspect<1,>=0.4.0 (from dataclasses-json<0.7,>=0.5.7->langchain_community)
  Downloading typing_inspect-0.9.0-py3-none-any.whl.metadata (1.5 kB)
Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from langchain<
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.10/dist-packages (from langchain<0.4.0,>=0.3.10->
Requirement already satisfied: jsonpatch<2.0,>=1.33 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.2
Requirement already satisfied: packaging<25,>=23.2 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.22
Requirement already satisfied: typing-extensions>=4.7 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3
Requirement already satisfied: httpx<1,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.125->langc
Requirement already satisfied: orjson<4.0.0,>=3.9.14 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.125->
Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>
Collecting python-dotenv>=0.21.0 (from pydantic-settings<3.0.0,>=2.4.0->langchain_community)
  Downloading python_dotenv-1.0.1-py3-none-any.whl.metadata (23 kB)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchai
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain_community)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain_comm
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain_comm
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain_co
Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>=0.1.125
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.10/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->
```

```
Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->la
Requirement already satisfied: pydantic-core==2.27.1 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->lar
Collecting mypy-extensions>=0.3.0 (from typing-inspect<1,>=0.4.0->dataclasses-json<0.7,>=0.5.7->langchain_community)
  Downloading mypy_extensions-1.0.0-py3-none-any.whl.metadata (1.1 kB)
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith
```

```
!pip install -U langchain-community
```

```
Requirement already satisfied: langchain-community in /usr/local/lib/python3.10/dist-packages (0.3.10)
Requirement already satisfied: PyYAML>=5.3 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (6.0.2)
Requirement already satisfied: SQLAlchemy<3,>=1.4 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (2.0.36)
Requirement already satisfied: aiohttp<4.0.0,>=3.8.3 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (3.11.9)
Requirement already satisfied: dataclasses-json<0.7,>=0.5.7 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (
Requirement already satisfied: httpx-sse<0.5.0,>=0.4.0 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.4.0
Requirement already satisfied: langchain<0.4.0,>=0.3.10 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.3.
Requirement already satisfied: langchain-core<0.4.0,>=0.3.22 in /usr/local/lib/python3.10/dist-packages (from langchain-community)
Requirement already satisfied: langsmith<0.2.0,>=0.1.125 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (0.1
Requirement already satisfied: numpy<2,>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (1.26.4)
Requirement already satisfied: pydantic-settings<3.0.0,>=2.4.0 in /usr/local/lib/python3.10/dist-packages (from langchain-community
Requirement already satisfied: requests<3,>=2 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (2.32.3)
Requirement already satisfied: tenacity!=8.4.0,<10,>=8.1.0 in /usr/local/lib/python3.10/dist-packages (from langchain-community) (9
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->lang
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-c
Requirement already satisfied: async-timeout<6.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->lang
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-comm
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langcha
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-c
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp<4.0.0,>=3.8.3->langchain-
Requirement already satisfied: marshmallow<4.0.0,>=3.18.0 in /usr/local/lib/python3.10/dist-packages (from dataclasses-json<0.7,>=0
Requirement already satisfied: typing-inspect<1,>=0.4.0 in /usr/local/lib/python3.10/dist-packages (from dataclasses-json<0.7,>=0.5
Requirement already satisfied: langchain-text-splitters<0.4.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from langchain<0.
Requirement already satisfied: pydantic<3.0.0,>=2.7.4 in /usr/local/lib/python3.10/dist-packages (from langchain<0.4.0,>=0.3.10->la
Requirement already satisfied: jsonpatch<2.0,>=1.33 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.22-
Requirement already satisfied: packaging<25,>=23.2 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.22->
Requirement already satisfied: typing-extensions>=4.7 in /usr/local/lib/python3.10/dist-packages (from langchain-core<0.4.0,>=0.3.2
Requirement already satisfied: httpx<1,>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.125->langcha
Requirement already satisfied: orjson<4.0.0,>=3.9.14 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0.1.125->la
Requirement already satisfied: requests-toolbelt<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from langsmith<0.2.0,>=0
Requirement already satisfied: python-dotenv>=0.21.0 in /usr/local/lib/python3.10/dist-packages (from pydantic-settings<3.0.0,>=2.4
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-community) (
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-commun
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3,>=2->langchain-commun
Requirement already satisfied: greenlet!=0.4.17 in /usr/local/lib/python3.10/dist-packages (from SQLAlchemy<3,>=1.4->langchain-comm
Requirement already satisfied: anyio in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>=0.1.125->
Requirement already satisfied: httpcore==1.* in /usr/local/lib/python3.10/dist-packages (from httpx<1,>=0.23.0->langsmith<0.2.0,>=0
Requirement already satisfied: h11<0.15,>=0.13 in /usr/local/lib/python3.10/dist-packages (from httpcore==1.*->httpx<1,>=0.23.0->la
Requirement already satisfied: jsonpointer>=1.9 in /usr/local/lib/python3.10/dist-packages (from jsonpatch<2.0,>=1.33->langchain-co
Requirement already satisfied: annotated-types>=0.6.0 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->lang
Requirement already satisfied: pydantic-core==2.27.1 in /usr/local/lib/python3.10/dist-packages (from pydantic<3.0.0,>=2.7.4->langc
Requirement already satisfied: mypy-extensions>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from typing-inspect<1,>=0.4.0->da
Requirement already satisfied: sniffio>=1.1 in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0.2
Requirement already satisfied: exceptiongroup in /usr/local/lib/python3.10/dist-packages (from anyio->httpx<1,>=0.23.0->langsmith<0
```

```
!pip install transformers
!pip install datasets
!pip install accelerate
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (4.46.3)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from transformers) (3.16.1)
Requirement already satisfied: huggingface-hub<1.0,>=0.23.2 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.26.
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (1.26.4)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformers) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (2024.9.11)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from transformers) (2.32.3)
Requirement already satisfied: tokenizers<0.21,>=0.20 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.20.3)
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.4.5)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.6)
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub<1.0,>=0.23.2->tr
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub<1.0,>=
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->transformers)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (2.2.3
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (2024.
Collecting datasets
  Downloading datasets-3.1.0-py3-none-any.whl.metadata (20 kB)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from datasets) (3.16.1)
```

```
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from datasets) (1.26.4)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (17.0.0)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.10/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.10/dist-packages (from datasets) (4.66.6)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (12 kB)
Collecting multiprocess<0.70.17 (from datasets)
  Downloading multiprocess-0.70.16-py310-none-any.whl.metadata (7.2 kB)
Collecting fsspec<=2024.9.0,>=2023.1.0 (from fsspec[http]<=2024.9.0,>=2023.1.0->datasets)
  Downloading fsspec-2024.9.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.11.9)
Requirement already satisfied: huggingface-hub>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.26.3)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (2.4.4
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.1)
Requirement already satisfied: async-timeout<6.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (4.0.3
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (24.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.18.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.23.
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datase
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (2
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (2
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->datasets
Downloading datasets-3.1.0-py3-none-any.whl (480 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 480.6/480.6 kB 8.7 MB/s eta 0:00:00
Downloading dill-0.3.8-py3-none-any.whl (116 kB)
```

```
!pip install sentence_transformers
```

```
Requirement already satisfied: sentence_transformers in /usr/local/lib/python3.10/dist-packages (3.2.1)
Requirement already satisfied: transformers<5.0.0,>=4.41.0 in /usr/local/lib/python3.10/dist-packages (from sentence_transformers)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (4.66.6)
Requirement already satisfied: torch>=1.11.0 in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (2.5.1+cu121)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (1.5.2)
Requirement already satisfied: scipy in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (1.13.1)
Requirement already satisfied: huggingface-hub>=0.20.0 in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (0.2
Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from sentence_transformers) (11.0.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence_transfor
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence_
Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence_t
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence_trans
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0->sentence_transfor
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.20.0-
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence_transformers) (3.4
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence_transformers) (3.1.4
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from torch>=1.11.0->sentence_transformers)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy==1.13.1->torch>=1.11.0->se
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->sentence_t
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->sent
Requirement already satisfied: tokenizers<0.21,>=0.20 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0-
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers<5.0.0,>=4.41.0->ser
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn->sentence_transformers)
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.10/dist-packages (from scikit-learn->sentence_transfo
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.11.0->sentence_tra
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.0->ser
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0.20.
```

```
from datasets import load_dataset

ds = load_dataset("microsoft/ms_marco", "v1.1")
```

```
README.md: 100% 9.48k/9.48k [00:00<00:00, 588kB/s]
validation-00000-of-00001.parquet: 100% 21.4M/21.4M [00:00<00:00, 22.8MB/s]
train-00000-of-00001.parquet: 100% 175M/175M [00:06<00:00, 26.7MB/s]
test-00000-of-00001.parquet: 100% 20.5M/20.5M [00:01<00:00, 19.7MB/s]
Generating validation split: 100% 10047/10047 [00:00<00:00, 14344.54 examples/s]
Generating train split: 100% 82326/82326 [00:04<00:00, 19426.61 examples/s]
Generating test split: 100% 9650/9650 [00:00<00:00, 13124.08 examples/s]
```

```
documents = ds["test"]
print(documents.column_names)
```

```
['answers', 'passages', 'query', 'query_id', 'query_type', 'wellFormedAnswers']
```

```
import pandas as pd

load_data = pd.DataFrame(documents)

load_data.head(5)
```

	answers	passages	query	query_id	query_type	wellFormedAnswers
0	[Yes]	{'is_selected': [0, 0, 1, 0, 0, 0, 0], 'passag...	does human hair stop squirrels	0	description	[]
1	[Fossil fuels are basically the remains of ani...	{'is_selected': [0, 1, 0, 0, 0, 0, 0, 0, 0], '...	what are the benefits of fossil fuels	1	description	[]
2	[The apothem of a regular polygon is a line se...	{'is_selected': [0, 0, 0, 0, 0, 1, 0, 0, 0], '...	what is a apothem	2	description	[]

```
print(documents[0])
```

```
{'answers': ['Yes'], 'passages': {'is_selected': [0, 0, 1, 0, 0, 0, 0], 'passage_text': ['We have been feeding our back yard squir
```

```
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity
from typing import List

def rank_passages_tfidf(query: str, passages: List[str]) -> List[int]:
    """Rank passages by cosine similarity with the query using TF-IDF vectors."""
    tfidf_vectorizer = TfidfVectorizer(stop_words="english")

    # Fit TF-IDF on passages and transform query
    passage_tfidf_matrix = tfidf_vectorizer.fit_transform(passages)
    query_tfidf = tfidf_vectorizer.transform([query])

    # Compute cosine similarity between query and all passages
    similarity_scores = cosine_similarity(query_tfidf, passage_tfidf_matrix).flatten()

    # Rank passages by similarity score
    ranked_indices = np.argsort(similarity_scores)[::-1]
    return ranked_indices.tolist()


# Sample query and passages
query = "What is Risk-based authentication  ?"
passages = [
    "Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to i
    "The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authority.
    "RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to taking
    "A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen and
    "Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes."
]

# Call the function to rank passages
ranked_indices = rank_passages_tfidf(query, passages)
```

```
# Display ranked passages
print("Ranked Passages:")
for idx in ranked_indices:
    print(f"{idx}: {passages[idx]}")
```

```
Ranked Passages:
4: Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes.
0: Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to
3: A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen ar
2: RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to takir
1: The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authorit
```

```
import torch
from transformers import BertTokenizer, BertModel
from sklearn.metrics.pairwise import cosine_similarity
from typing import List
from sentence_transformers import SentenceTransformer
from sentence_transformers import SentenceTransformer
from sklearn.metrics.pairwise import cosine_similarity
from typing import List

def rank_passages_bert(query: str, passages: List[str]) -> List[int]:
    """
    Rank passages based on their similarity to a given query using SentenceTransformer embeddings.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (descending order).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('distilbert-base-nli-mean-tokens')

    # Encode the query
    query_embedding = model.encode(query)

    # Encode the passages
    passage_embeddings = model.encode(passages)

    # Compute cosine similarity between query and passages
    similarity_scores = cosine_similarity([query_embedding], passage_embeddings).flatten()

    # Rank indices by similarity scores in descending order
    ranked_indices = similarity_scores.argsort()[::-1]
    return ranked_indices.tolist()

query = "What is  data-driven?"
passages = [
    "Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to i
    "The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authority.
    "RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to taking
    "A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen and
    "Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes."
]


ranked_indices = rank_passages_bert(query, passages)

print("Ranked Passages:")
for idx in ranked_indices:
    print(f"{idx}: {passages[idx]}")
```

```
modules.json: 100%  229/229 [00:00<00:00, 9.41kB/s]
config_sentence_transformers.json: 100%  122/122 [00:00<00:00, 2.72kB/s]
README.md: 100%  4.02k/4.02k [00:00<00:00, 113kB/s]
sentence_bert_config.json: 100%  53.0/53.0 [00:00<00:00, 1.33kB/s]
config.json: 100%  550/550 [00:00<00:00, 11.8kB/s]
model.safetensors: 100%  265M/265M [00:01<00:00, 182MB/s]
tokenizer_config.json: 100%  450/450 [00:00<00:00, 10.3kB/s]
vocab.txt: 100%  232k/232k [00:00<00:00, 4.62MB/s]
tokenizer.json: 100%  466k/466k [00:00<00:00, 11.5MB/s]
special_tokens_map.json: 100%  112/112 [00:00<00:00, 2.48kB/s]
1_Pooling/config.json: 100%  190/190 [00:00<00:00, 4.40kB/s]
Ranked Passages:
4: Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes.
2: RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to takir
3: A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen ar
0: Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to
1: The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authorit
```

```
# import torch
# from transformers import BertTokenizer, BertModel
# from sklearn.metrics.pairwise import cosine_similarity
# from typing import List
# from sentence_transformers import SentenceTransformer


# def rank_passages_bert(query: str, passages: List[str]) -> List[int]:
#     from transformers import BertTokenizer, BertModel
#     import torch
#     from sklearn.metrics.pairwise import cosine_similarity

#     tokenizer = BertTokenizer.from_pretrained('transformersbook/bert-base-uncased-finetuned-clinc')
#     model = BertModel.from_pretrained('transformersbook/bert-base-uncased-finetuned-clinc')

#     def cls_embedding(model_output):
#         # CLS token embedding is the first token in last_hidden_state
#         return model_output.last_hidden_state[:, 0, :]

#     # Tokenize and encode the query
#     query_tokens = tokenizer(query, return_tensors='pt', padding=True, truncation=True)
#     with torch.no_grad():
#         query_output = model(**query_tokens)
#         query_embedding = cls_embedding(query_output)

#     # Encode each passage and collect embeddings
#     passage_embeddings = []
#     for passage in passages:
#         passage_tokens = tokenizer(passage, return_tensors='pt', padding=True, truncation=True)
#         with torch.no_grad():
#             passage_output = model(**passage_tokens)
#             passage_embedding = cls_embedding(passage_output)
#         passage_embeddings.append(passage_embedding)

#     # Concatenate passage embeddings for comparison
#     passage_embeddings = torch.cat(passage_embeddings, dim=0)

#     # Compute cosine similarity between query and passages
#     similarity_scores = cosine_similarity(query_embedding.numpy(), passage_embeddings.numpy()).flatten()

#     # Rank indices by similarity scores in descending order
#     ranked_indices = similarity_scores.argsort()[::-1]
#     return ranked_indices.tolist()


# query = "What is  data-driven?"
# passages = [
#     "Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to
#     "The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authorit
#     "RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to takir
```

```
#     "A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen ar
#     "Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes."
# ]


# ranked_indices = rank_passages_bert(query, passages)

# print("Ranked Passages:")
# for idx in ranked_indices:
#     print(f"{idx}: {passages[idx]}")




import torch
from transformers import BertTokenizer, BertModel
from sklearn.covariance import EmpiricalCovariance
from scipy.spatial.distance import mahalanobis
from typing import List

from sentence_transformers import SentenceTransformer
from sklearn.covariance import EmpiricalCovariance
from scipy.spatial.distance import mahalanobis
from typing import List
import numpy as np

def rank_passages_bert_mahalanobis(query: str, passages: List[str]) -> List[int]:
    """
    Rank passages based on their similarity to a given query using SentenceTransformer embeddings
    and Mahalanobis distance.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (ascending Mahalanobis distance).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('distilbert-base-nli-mean-tokens')

    # Encode query
    query_embedding = model.encode(query)

    # Encode passages
    passage_embeddings = model.encode(passages)

    # Calculate covariance matrix and its inverse
    cov_estimator = EmpiricalCovariance()
    cov_estimator.fit(passage_embeddings)
    cov_matrix_inv = cov_estimator.precision_

    # Calculate Mahalanobis distances
    distances = [
        mahalanobis(query_embedding, passage_embedding, cov_matrix_inv)
        for passage_embedding in passage_embeddings
    ]

    # Rank indices based on distances
    ranked_indices = sorted(range(len(distances)), key=lambda i: distances[i])
    return ranked_indices

query = "What is  atomizer?"
passages = [
    "Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to i
    "The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authority.
    "RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to taking
    "A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen and
    "Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes."
]


ranked_indices = rank_passages_bert_mahalanobis(query, passages)

print("Ranked Passages:")
for idx in ranked_indices:
    print(f"{idx}: {passages[idx]}")
```

```
Ranked Passages:
3: A rebuildable atomizer (RBA), often referred to as simply a 'rebuildable,' is a special type of atomizer used in the Vape Pen ar
1: The Reserve Bank of Australia (RBA) came into being on 14 January 1960 as Australia's central bank and banknote issuing authorit
4: Risk-based authentication (RBA) is a method of applying varying levels of stringency to authentication processes.
2: RBA uses a data-driven, decision-making process to help communities and organizations get beyond talking about problems to takir
0: Results-Based Accountability® (also known as RBA) is a disciplined way of thinking and taking action that communities can use to
```

###Evaluation

```
import numpy as np
import pandas as pd
from typing import List

def evaluate_metrics(df: pd.DataFrame, relevance_col: str, rank_col: str, k: int = 3):
    """
    Function to calculate
        - MRR (Mean Reciprocal Rank)
        - nDCG (Normalized Discounted Cumulative Gain)
        - Precision@k
        - Recall@k
        - MAP (Mean Average Precision)
    """
    mrr_scores = []
    ndcg_scores = []
    precision_at_k = []
    recall_at_k = []
    average_precisions = []

    for _, row in df.iterrows():
        relevance = np.array(row[relevance_col])  # Convert relevance list to numpy array
        rank = np.array(row[rank_col])  # Ranked indices
        relevance_at_rank = relevance[rank]

        # MRR: Find the rank of the first relevant document
        first_relevant = np.where(relevance_at_rank == 1)[0]
        if len(first_relevant) > 0:
            mrr_scores.append(1 / (first_relevant[0] + 1))
        else:
            mrr_scores.append(0)

        # nDCG
        dcg = sum((2 ** relevance_at_rank[i] - 1) / np.log2(i + 2) for i in range(len(relevance_at_rank)))
        ideal_relevance = sorted(relevance, reverse=True)
        idcg = sum((2 ** ideal_relevance[i] - 1) / np.log2(i + 2) for i in range(len(ideal_relevance)))
        ndcg = dcg / idcg if idcg > 0 else 0
        ndcg_scores.append(ndcg)

        # Top-k relevance
        relevance_at_k = relevance_at_rank[:k]

        # Precision@k
        precision = relevance_at_k.sum() / k
        precision_at_k.append(precision)

        # Recall@k
        total_relevant = relevance.sum()
        recall = relevance_at_k.sum() / total_relevant if total_relevant > 0 else 0
        recall_at_k.append(recall)

        # Average Precision (AP)
        num_relevant_retrieved = 0
        cumulative_precision = 0
        for i in range(len(relevance_at_rank)):
            if relevance_at_rank[i] == 1:
                num_relevant_retrieved += 1
                cumulative_precision += num_relevant_retrieved / (i + 1)
        ap = cumulative_precision / total_relevant if total_relevant > 0 else 0
        average_precisions.append(ap)

    # Aggregate results
    metrics = {
        "MRR": np.mean(mrr_scores),
```

```
        "nDCG": np.mean(ndcg_scores),
        f"Precision at {k}": np.mean(precision_at_k),
        f"Recall at {k}": np.mean(recall_at_k),
        "MAP": np.mean(average_precisions),
    }

    return metrics
```

```
!pip install datasets
```

```
Requirement already satisfied: datasets in /usr/local/lib/python3.10/dist-packages (3.1.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from datasets) (3.16.1)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from datasets) (1.26.4)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (17.0.0)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.3.8)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from datasets) (2.2.2)
Requirement already satisfied: requests>=2.32.2 in /usr/local/lib/python3.10/dist-packages (from datasets) (2.32.3)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.10/dist-packages (from datasets) (4.66.6)
Requirement already satisfied: xxhash in /usr/local/lib/python3.10/dist-packages (from datasets) (3.5.0)
Requirement already satisfied: multiprocess<0.70.17 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.70.16)
Requirement already satisfied: fsspec<=2024.9.0,>=2023.1.0 in /usr/local/lib/python3.10/dist-packages (from fsspec[http]<=2024.9.0,
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.11.9)
Requirement already satisfied: huggingface-hub>=0.23.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.26.3)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from datasets) (24.2)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (2.4.4)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.1)
Requirement already satisfied: async-timeout<6.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (4.0.3)
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (24.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.5.0)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (0.2.1)
Requirement already satisfied: yarl<2.0,>=1.17.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.18.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.23.0
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (2.2
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (202
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->datasets)
```

```
from datasets import load_dataset
import pandas as pd

# Load the dataset
ds = load_dataset("microsoft/ms_marco", "v1.1")

# Convert the validation split to a pandas DataFrame
df = ds['validation'].to_pandas()
# Extract the passage text and is_selected fields
df["passage_text"] = df["passages"].apply(lambda x: x["passage_text"].tolist())  # Convert NumPy array to list
df["is_selected"] = df["passages"].apply(lambda x: x["is_selected"].tolist())    # Convert NumPy array to list



# Extract passage text and selection status
# df["passage_text"] = df["passages"].apply(lambda x: x["passage_text"])
# df["passage_text"] = df["passage_text"].apply(lambda x: x.split(". ") if isinstance(x, str) else x)
# df["is_selected"] = df["passages"].apply(lambda x: x["is_selected"])

# Drop the original passages column and set query_id as index
df.drop("passages", axis=1, inplace=True)
df.set_index("query_id", inplace=True)

# Display the first few rows
df.head()
```

query_id	answers	query	query_type	wellFormedAnswers	passage_text	is_selected
9652	[Approximately $15,000 per year.]	walgreens store sales average	numeric	[]	[The average Walgreens salary ranges from appr...	[1, 0, 0, 0, 0, 0]
9653	[$21,550 per year, The average hourly wage for...	how much do bartenders make	numeric	[]	[A bartender’s income is comprised mostly of t...	[0, 1, 0, 0, 0, 0, 0, 0]
9654	[A boil, also called a furuncle, is a deep fol...	what is a furuncle boil	description	[]	[Knowledge center. A boil, also known as a fur...	[0, 0, 0, 0, 0, 0, 1, 0]
9655	[Detect and assess a wide range of disorders, ...	what can urinalysis detect	description	[]	[Urinalysis: One way to test for bladder cance...	[0, 0, 0, 0, 1, 0, 0, 0, 0]
9656	[Shigellosis, diseases of the nervous system, ...	what is vitamin a used for	description	[]	[Since vitamin A is fat-soluble it is not need...	[0, 0, 0, 0, 1, 0, 0, 0, 0]

```
!pip install pandarallel
```

```
Collecting pandarallel
  Downloading pandarallel-1.6.5.tar.gz (14 kB)
  Preparing metadata (setup.py) ... done
Requirement already satisfied: dill>=0.3.1 in /usr/local/lib/python3.10/dist-packages (from pandarallel) (0.3.8)
Requirement already satisfied: pandas>=1 in /usr/local/lib/python3.10/dist-packages (from pandarallel) (2.2.2)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages (from pandarallel) (5.9.5)
Requirement already satisfied: numpy>=1.22.4 in /usr/local/lib/python3.10/dist-packages (from pandas>=1->pandarallel) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas>=1->pandarallel) (2.8
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas>=1->pandarallel) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas>=1->pandarallel) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas>=1->pandara
Building wheels for collected packages: pandarallel
  Building wheel for pandarallel (setup.py) ... done
  Created wheel for pandarallel: filename=pandarallel-1.6.5-py3-none-any.whl size=16674 sha256=1a6038db57bbee8c344b2a09d92d1cd16a4
  Stored in directory: /root/.cache/pip/wheels/50/4f/1e/34e057bb868842209f1623f195b74fd7eda229308a7352d47f
Successfully built pandarallel
Installing collected packages: pandarallel
Successfully installed pandarallel-1.6.5
```

```
from pandarallel import pandarallel
pandarallel.initialize(progress_bar=True)
```

```
INFO: Pandarallel will run on 1 workers.
INFO: Pandarallel will use Memory file system to transfer data between the main process and workers.
```

Model with cosine

```
ranked_indices_bert = rank_passages_bert(df.iloc[0]["query"], df.iloc[0]["passage_text"])
```

```
print("Ranked Indices of bert with cosine:", ranked_indices_bert)
```

```
Ranked Indices of bert with cosine: [5, 1, 0, 3, 4, 2]
```

```
df["rank_passages_bert"] = df.apply(lambda x: rank_passages_bert(x["query"], x["passage_text"]), axis=1)
df.head()
```

```
---------------------------------------------------------------------------
OutOfMemoryError                          Traceback (most recent call last)
<ipython-input-29-db541c095c10> in <cell line: 1>()
----> 1 df["rank_passages_bert"] = df.apply(lambda x: rank_passages_bert(x["query"], x["passage_text"]), axis=1)
      2 df.head()

16 frames
/usr/local/lib/python3.10/dist-packages/transformers/models/distilbert/modeling_distilbert.py in forward(self, input_ids,
input_embeds)
    129         position_embeddings = self.position_embeddings(position_ids)  # (bs, max_seq_length, dim)
    130
--> 131         embeddings = input_embeds + position_embeddings  # (bs, max_seq_length, dim)
    132         embeddings = self.LayerNorm(embeddings)  # (bs, max_seq_length, dim)
    133         embeddings = self.dropout(embeddings)  # (bs, max_seq_length, dim)

OutOfMemoryError: CUDA out of memory. Tried to allocate 20.00 MiB. GPU 0 has a total capacity of 14.75 GiB of which 11.06 MiB is
free. Process 4838 has 14.73 GiB memory in use. Of the allocated memory 13.87 GiB is allocated by PyTorch, and 756.26 MiB is
reserved by PyTorch but unallocated. If reserved but unallocated memory is large try setting
PYTORCH_CUDA_ALLOC_CONF=expandable_segments:True to avoid fragmentation.  See documentation for Memory Management
```

```
result1 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert')
result1
```

model 2 with mahalanobis

```
ranked_indices_bert_mahalanobis = rank_passages_bert_mahalanobis(df.iloc[0]["query"], df.iloc[0]["passage_text"])
```

```
print("Ranked Indices of bert with mahalanobis:", ranked_indices_bert_mahalanobis)
```

```
df["rank_passages_bert_mahalanobis"] = df.apply(lambda x: rank_passages_bert_mahalanobis(x["query"], x["passage_text"]), axis=1)
df.head()
```

```
result2 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert_mahalanobis')
result2
```

Model 3 with Dot

```
from sentence_transformers import SentenceTransformer
from typing import List
import numpy as np

def rank_passages_bert_dot(query: str, passages: List[str]) -> List[int]:
    """
    Rank passages based on their similarity to a query using SentenceTransformer embeddings
    and dot product scoring.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (descending order).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('distilbert-base-nli-mean-tokens')

    # Encode the query
    query_embedding = model.encode(query)

    # Encode the passages
    passage_embeddings = model.encode(passages)

    # Compute dot product similarity scores
    similarity_scores = np.dot(passage_embeddings, query_embedding)

    # Rank indices by similarity scores in descending order
    ranked_indices = np.argsort(similarity_scores)[::-1].tolist()
    return ranked_indices
```

```
ranked_indices_bert_dot = rank_passages_bert_dot(df.iloc[0]["query"], df.iloc[0]["passage_text"])


print("Ranked Indices of bert with dot:", ranked_indices_bert_dot)


df["rank_passages_bert_dot"] = df.apply(lambda x: rank_passages_bert_dot(x["query"], x["passage_text"]), axis=1)
df.head()


result3 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert_dot')
result3
```

model 4 with Euclidean

```
from sentence_transformers import SentenceTransformer
import numpy as np
from typing import List

def rank_passages_bert_Euclidean(query: str, passages: List[str]) -> List[int]:
    """
    Rank passages based on their similarity to a query using SentenceTransformer embeddings
    and Euclidean distance.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (ascending order of Euclidean distance).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('distilbert-base-nli-mean-tokens')

    # Encode the query
    query_embedding = model.encode(query)

    # Encode the passages
    passage_embeddings = model.encode(passages)

    # Compute Euclidean distances
    distances = np.linalg.norm(passage_embeddings - query_embedding, axis=1)

    # Rank indices by ascending distances (smaller distance = more similar)
    ranked_indices = np.argsort(distances)
    return ranked_indices.tolist()



ranked_indices_bert_Euclidean = rank_passages_bert_Euclidean(df.iloc[0]["query"], df.iloc[0]["passage_text"])


print("Ranked Indices of bert with Euclidean:", ranked_indices_bert_Euclidean)


df["rank_passages_bert_Euclidean"] = df.apply(lambda x: rank_passages_bert_Euclidean(x["query"], x["passage_text"]), axis=1)
df.head()


result4 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert_Euclidean')
result4
```

Model 5 BM25

```
!pip install rank-bm25


from rank_bm25 import BM25Okapi
from transformers import BertTokenizer
from typing import List

def rank_passages_bert_bm25(query: str, passages: List[str]) -> List[int]:
    tokenizer = BertTokenizer.from_pretrained('transformersbook/bert-base-uncased-finetuned-clinc')

    # Tokenize passages
    tokenized_passages = [tokenizer.tokenize(passage) for passage in passages]
```

```
    # Initialize BM25
    bm25 = BM25Okapi(tokenized_passages)

    # Tokenize query
    tokenized_query = tokenizer.tokenize(query)

    # Compute BM25 scores
    bm25_scores = bm25.get_scores(tokenized_query)

    # Get ranked indices based on BM25 scores
    ranked_indices = sorted(range(len(bm25_scores)), key=lambda i: bm25_scores[i], reverse=True)
    return ranked_indices



ranked_indices_bert_bm25 = rank_passages_bert_bm25(df.iloc[0]["query"], df.iloc[0]["passage_text"])


print("Ranked Indices of bert with bm25:", ranked_indices_bert_bm25)


df["rank_passages_bert_bm25"] = df.apply(lambda x: rank_passages_bert_bm25(x["query"], x["passage_text"]), axis=1)
df.head()


result5 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert_bm25')
result5
```

Model 6 with KNN

```
from sentence_transformers import SentenceTransformer
from sklearn.neighbors import NearestNeighbors
from typing import List

def rank_passages_bert_knn(query: str, passages: List[str], k: int = 5) -> List[int]:
    """
    Rank passages based on their similarity to a query using SentenceTransformer embeddings
    and K-Nearest Neighbors (KNN).

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.
        k (int): Number of nearest neighbors to retrieve.

    Returns:
        List[int]: Indices of the top-k most similar passages ranked by similarity.
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('distilbert-base-nli-mean-tokens')

    # Encode the query and passages
    query_embedding = model.encode(query)
    passage_embeddings = model.encode(passages)

    # Use KNN to find the k nearest neighbors
    knn = NearestNeighbors(n_neighbors=k, metric='euclidean')
    knn.fit(passage_embeddings)
    distances, indices = knn.kneighbors([query_embedding])

    # Flatten indices (KNN returns a 2D array for batching)
    ranked_indices = indices[0].tolist()
    return ranked_indices



ranked_indices_bert_knn = rank_passages_bert_knn(df.iloc[0]["query"], df.iloc[0]["passage_text"])


print("Ranked Indices of bert with knn :", ranked_indices_bert_knn)


df["rank_passages_bert_knn"] = df.apply(lambda x: rank_passages_bert_knn(x["query"], x["passage_text"]), axis=1)
df.head()
```

```
result6 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_bert_knn')
result6
```

Model: 7 with mpnet cosine

```
from sentence_transformers import SentenceTransformer
from sklearn.preprocessing import normalize
from sklearn.metrics.pairwise import cosine_similarity
from typing import List

def rank_passages_mpnet(query: str, passages: List[str], model_name: str = 'all-mpnet-base-v2') -> List[int]:
    """
    Rank passages based on their similarity to a given query using SentenceTransformer embeddings.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.
        model_name (str): The name of the SentenceTransformer model.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (descending order).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer(model_name)

    # Encode and normalize the query and passages
    query_embedding = normalize(model.encode(query).reshape(1, -1))
    passage_embeddings = normalize(model.encode(passages))

    # Compute cosine similarity between query and passages
    similarity_scores = cosine_similarity(query_embedding, passage_embeddings).flatten()

    # Rank indices by similarity scores in descending order
    ranked_indices = similarity_scores.argsort()[::-1]
    return ranked_indices.tolist()



ranked_indices_mpnet = rank_passages_mpnet(df.iloc[0]["query"], df.iloc[0]["passage_text"])


print("Ranked Indices of mpnet :", ranked_indices_mpnet)


df["rank_passages_mpnet"] = df.apply(lambda x: rank_passages_mpnet(x["query"], x["passage_text"]), axis=1)
df.head()


result7 = evaluate_metrics(df, relevance_col='is_selected', rank_col='rank_passages_mpnet')
result7
```

model:8 mpnet with mahalanobis

```
from sentence_transformers import SentenceTransformer
from sklearn.preprocessing import normalize
import numpy as np
from typing import List

def rank_passages_mpnet_mahalanobis(query: str, passages: List[str]) -> List[int]:
    """
    Rank passages based on their similarity to a given query using SentenceTransformer embeddings
    and Mahalanobis distance.

    Args:
        query (str): The input query.
        passages (List[str]): A list of passages to rank.

    Returns:
        List[int]: Indices of passages ranked by similarity to the query (ascending Mahalanobis distance).
    """
    # Load the SentenceTransformer model
    model = SentenceTransformer('all-mpnet-base-v2')

    # Encode query
    query_embedding = model.encode(query)
```

```
# Encode passages
passage_embeddings = model.encode(passages)

# Calculate covariance matrix and its inverse
```