Retrieval exercise

In this exercise, you will implement the query likelihood model with Jelinek-Mercer smoothing. This assignment builds on the previous assignment for creating a Pyserini index.

1. Build the index

Download the MS MARCO passage collection and build an index using Pyserini. This code is similar to PART 1 of the indexing assignment.

In [17]:
```
!pip install pyserini
!pip install faiss-cpu

!git clone https://github.com/castorini/anserini.git --recurse-submodules

!wget https://msmarco.blob.core.windows.net/msmarcoranking/collection.tar.gz -P
!tar xvfz data/msmarco_passage/collection.tar.gz -C data/msmarco_passage

!cd anserini && python tools/scripts/msmarco/convert_collection_to_jsonl.py \
 --collection-path ../data/msmarco_passage/collection.tsv --output-folder ../dat

!rm data/msmarco_passage/*.tsv
!rm -rf sample_data

!python -m pyserini.index.lucene -collection JsonCollection -generator DefaultLu
-input data/msmarco_passage/collection_jsonl -index indexes/lucene-index-msmarco
```

```
Requirement already satisfied: pyserini in /usr/local/lib/python3.10/dist-package
s (0.22.0)
Requirement already satisfied: Cython>=0.29.21 in /usr/local/lib/python3.10/dist-
packages (from pyserini) (3.0.2)
Requirement already satisfied: numpy>=1.18.1 in /usr/local/lib/python3.10/dist-pa
ckages (from pyserini) (1.23.5)
Requirement already satisfied: pandas>=1.4.0 in /usr/local/lib/python3.10/dist-pa
ckages (from pyserini) (1.5.3)
Requirement already satisfied: pyjnius>=1.4.0 in /usr/local/lib/python3.10/dist-p
ackages (from pyserini) (1.5.0)
Requirement already satisfied: scikit-learn>=0.22.1 in /usr/local/lib/python3.10/
dist-packages (from pyserini) (1.2.2)
Requirement already satisfied: scipy>=1.4.1 in /usr/local/lib/python3.10/dist-pac
kages (from pyserini) (1.11.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (f
rom pyserini) (4.66.1)
Requirement already satisfied: transformers>=4.6.0 in /usr/local/lib/python3.10/d
ist-packages (from pyserini) (4.33.2)
Requirement already satisfied: sentencepiece>=0.1.95 in /usr/local/lib/python3.1
0/dist-packages (from pyserini) (0.1.99)
Requirement already satisfied: nmslib>=2.1.1 in /usr/local/lib/python3.10/dist-pa
ckages (from pyserini) (2.1.1)
Requirement already satisfied: onnxruntime>=1.8.1 in /usr/local/lib/python3.10/di
st-packages (from pyserini) (1.15.1)
Requirement already satisfied: lightgbm>=3.3.2 in /usr/local/lib/python3.10/dist-
packages (from pyserini) (4.0.0)
Requirement already satisfied: spacy>=3.2.1 in /usr/local/lib/python3.10/dist-pac
kages (from pyserini) (3.6.1)
Requirement already satisfied: pyyaml in /usr/local/lib/python3.10/dist-packages
(from pyserini) (6.0.1)
Requirement already satisfied: pybind11<2.6.2 in /usr/local/lib/python3.10/dist-p
ackages (from nmslib>=2.1.1->pyserini) (2.6.1)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages
(from nmslib>=2.1.1->pyserini) (5.9.5)
Requirement already satisfied: coloredlogs in /usr/local/lib/python3.10/dist-pack
ages (from onnxruntime>=1.8.1->pyserini) (15.0.1)
Requirement already satisfied: flatbuffers in /usr/local/lib/python3.10/dist-pack
ages (from onnxruntime>=1.8.1->pyserini) (23.5.26)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packag
es (from onnxruntime>=1.8.1->pyserini) (23.1)
Requirement already satisfied: protobuf in /usr/local/lib/python3.10/dist-package
s (from onnxruntime>=1.8.1->pyserini) (3.20.3)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages
(from onnxruntime>=1.8.1->pyserini) (1.12)
Requirement already satisfied: python-dateutil>=2.8.1 in /usr/local/lib/python3.1
0/dist-packages (from pandas>=1.4.0->pyserini) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-pac
kages (from pandas>=1.4.0->pyserini) (2023.3.post1)
Requirement already satisfied: joblib>=1.1.1 in /usr/local/lib/python3.10/dist-pa
ckages (from scikit-learn>=0.22.1->pyserini) (1.3.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in /usr/local/lib/python3.10/
dist-packages (from scikit-learn>=0.22.1->pyserini) (3.2.0)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/pyth
on3.10/dist-packages (from spacy>=3.2.1->pyserini) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/pyth
on3.10/dist-packages (from spacy>=3.2.1->pyserini) (1.0.4)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python
3.10/dist-packages (from spacy>=3.2.1->pyserini) (1.0.9)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.10/d
ist-packages (from spacy>=3.2.1->pyserini) (2.0.7)
```

```
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.1
0/dist-packages (from spacy>=3.2.1->pyserini) (3.0.8)
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /usr/local/lib/python3.10/d
ist-packages (from spacy>=3.2.1->pyserini) (8.1.12)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.10/
dist-packages (from spacy>=3.2.1->pyserini) (1.1.2)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.10/d
ist-packages (from spacy>=3.2.1->pyserini) (2.4.7)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.
10/dist-packages (from spacy>=3.2.1->pyserini) (2.0.9)
Requirement already satisfied: typer<0.10.0,>=0.3.0 in /usr/local/lib/python3.10/
dist-packages (from spacy>=3.2.1->pyserini) (0.9.0)
Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.10/dist-pa
ckages (from spacy>=3.2.1->pyserini) (0.10.2)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /usr/local/lib/python
3.10/dist-packages (from spacy>=3.2.1->pyserini) (6.4.0)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.
10/dist-packages (from spacy>=3.2.1->pyserini) (2.31.0)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<3.0.0,>=1.7.4 in /usr/loca
l/lib/python3.10/dist-packages (from spacy>=3.2.1->pyserini) (1.10.12)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages
(from spacy>=3.2.1->pyserini) (3.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packa
ges (from spacy>=3.2.1->pyserini) (67.7.2)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/lib/python3.
10/dist-packages (from spacy>=3.2.1->pyserini) (3.3.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-package
s (from transformers>=4.6.0->pyserini) (3.12.2)
Requirement already satisfied: huggingface-hub<1.0,>=0.15.1 in /usr/local/lib/pyt
hon3.10/dist-packages (from transformers>=4.6.0->pyserini) (0.17.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dis
t-packages (from transformers>=4.6.0->pyserini) (2023.6.3)
Requirement already satisfied: tokenizers!=0.11.3,<0.14,>=0.11.1 in /usr/local/li
b/python3.10/dist-packages (from transformers>=4.6.0->pyserini) (0.13.3)
Requirement already satisfied: safetensors>=0.3.1 in /usr/local/lib/python3.10/di
st-packages (from transformers>=4.6.0->pyserini) (0.3.3)
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages
(from huggingface-hub<1.0,>=0.15.1->transformers>=4.6.0->pyserini) (2023.6.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/pytho
n3.10/dist-packages (from huggingface-hub<1.0,>=0.15.1->transformers>=4.6.0->pyse
rini) (4.5.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-package
s (from python-dateutil>=2.8.1->pandas>=1.4.0->pyserini) (1.16.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python
3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy>=3.2.1->pyserini) (3.2.0)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-pac
kages (from requests<3.0.0,>=2.13.0->spacy>=3.2.1->pyserini) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/di
st-packages (from requests<3.0.0,>=2.13.0->spacy>=3.2.1->pyserini) (2.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/di
st-packages (from requests<3.0.0,>=2.13.0->spacy>=3.2.1->pyserini) (2023.7.22)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /usr/local/lib/python3.10/di
st-packages (from thinc<8.2.0,>=8.1.8->spacy>=3.2.1->pyserini) (0.7.10)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python
3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy>=3.2.1->pyserini) (0.1.2)
Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/python3.10/d
ist-packages (from typer<0.10.0,>=0.3.0->spacy>=3.2.1->pyserini) (8.1.7)
Requirement already satisfied: humanfriendly>=9.1 in /usr/local/lib/python3.10/di
st-packages (from coloredlogs->onnxruntime>=1.8.1->pyserini) (10.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-
```

```
packages (from jinja2->spacy>=3.2.1->pyserini) (2.1.3)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.10/dist-pac
kages (from sympy->onnxruntime>=1.8.1->pyserini) (1.3.0)
Requirement already satisfied: faiss-cpu in /usr/local/lib/python3.10/dist-packag
es (1.7.4)
fatal: destination path 'anserini' already exists and is not an empty directory.
--2023-09-19 10:53:22--  https://msmarco.blob.core.windows.net/msmarcoranking/col
lection.tar.gz
Resolving msmarco.blob.core.windows.net (msmarco.blob.core.windows.net)... 20.15
0.34.4
Connecting to msmarco.blob.core.windows.net (msmarco.blob.core.windows.net)|20.15
0.34.4|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1035009698 (987M) [application/octet-stream]
Saving to: ‘data/msmarco_passage/collection.tar.gz.1’

collection.tar.gz.1 100%[===================>] 987.06M  12.3MB/s    in 85s

2023-09-19 10:54:48 (11.6 MB/s) - ‘data/msmarco_passage/collection.tar.gz.1’ save
d [1035009698/1035009698]

collection.tsv
Converting collection...
Converted 0 docs, writing into file 1
Converted 100,000 docs, writing into file 1
Converted 200,000 docs, writing into file 1
Converted 300,000 docs, writing into file 1
Converted 400,000 docs, writing into file 1
Converted 500,000 docs, writing into file 1
Converted 600,000 docs, writing into file 1
Converted 700,000 docs, writing into file 1
Converted 800,000 docs, writing into file 1
Converted 900,000 docs, writing into file 1
Converted 1,000,000 docs, writing into file 2
Converted 1,100,000 docs, writing into file 2
Converted 1,200,000 docs, writing into file 2
Converted 1,300,000 docs, writing into file 2
Converted 1,400,000 docs, writing into file 2
Converted 1,500,000 docs, writing into file 2
Converted 1,600,000 docs, writing into file 2
Converted 1,700,000 docs, writing into file 2
Converted 1,800,000 docs, writing into file 2
Converted 1,900,000 docs, writing into file 2
Converted 2,000,000 docs, writing into file 3
Converted 2,100,000 docs, writing into file 3
Converted 2,200,000 docs, writing into file 3
Converted 2,300,000 docs, writing into file 3
Converted 2,400,000 docs, writing into file 3
Converted 2,500,000 docs, writing into file 3
Converted 2,600,000 docs, writing into file 3
Converted 2,700,000 docs, writing into file 3
Converted 2,800,000 docs, writing into file 3
Converted 2,900,000 docs, writing into file 3
Converted 3,000,000 docs, writing into file 4
Converted 3,100,000 docs, writing into file 4
Converted 3,200,000 docs, writing into file 4
Converted 3,300,000 docs, writing into file 4
Converted 3,400,000 docs, writing into file 4
Converted 3,500,000 docs, writing into file 4
Converted 3,600,000 docs, writing into file 4
```

```
Converted 3,700,000 docs, writing into file 4
Converted 3,800,000 docs, writing into file 4
Converted 3,900,000 docs, writing into file 4
Converted 4,000,000 docs, writing into file 5
Converted 4,100,000 docs, writing into file 5
Converted 4,200,000 docs, writing into file 5
Converted 4,300,000 docs, writing into file 5
Converted 4,400,000 docs, writing into file 5
Converted 4,500,000 docs, writing into file 5
Converted 4,600,000 docs, writing into file 5
Converted 4,700,000 docs, writing into file 5
Converted 4,800,000 docs, writing into file 5
Converted 4,900,000 docs, writing into file 5
Converted 5,000,000 docs, writing into file 6
Converted 5,100,000 docs, writing into file 6
Converted 5,200,000 docs, writing into file 6
Converted 5,300,000 docs, writing into file 6
Converted 5,400,000 docs, writing into file 6
Converted 5,500,000 docs, writing into file 6
Converted 5,600,000 docs, writing into file 6
Converted 5,700,000 docs, writing into file 6
Converted 5,800,000 docs, writing into file 6
Converted 5,900,000 docs, writing into file 6
Converted 6,000,000 docs, writing into file 7
Converted 6,100,000 docs, writing into file 7
Converted 6,200,000 docs, writing into file 7
Converted 6,300,000 docs, writing into file 7
Converted 6,400,000 docs, writing into file 7
Converted 6,500,000 docs, writing into file 7
Converted 6,600,000 docs, writing into file 7
Converted 6,700,000 docs, writing into file 7
Converted 6,800,000 docs, writing into file 7
Converted 6,900,000 docs, writing into file 7
Converted 7,000,000 docs, writing into file 8
Converted 7,100,000 docs, writing into file 8
Converted 7,200,000 docs, writing into file 8
Converted 7,300,000 docs, writing into file 8
Converted 7,400,000 docs, writing into file 8
Converted 7,500,000 docs, writing into file 8
Converted 7,600,000 docs, writing into file 8
Converted 7,700,000 docs, writing into file 8
Converted 7,800,000 docs, writing into file 8
Converted 7,900,000 docs, writing into file 8
Converted 8,000,000 docs, writing into file 9
Converted 8,100,000 docs, writing into file 9
Converted 8,200,000 docs, writing into file 9
Converted 8,300,000 docs, writing into file 9
Converted 8,400,000 docs, writing into file 9
Converted 8,500,000 docs, writing into file 9
Converted 8,600,000 docs, writing into file 9
Converted 8,700,000 docs, writing into file 9
Converted 8,800,000 docs, writing into file 9
Done!
WARNING: sun.reflect.Reflection.getCallerClass is not supported. This will impact
performance.
2023-09-19 10:56:54,246 INFO  [main] index.IndexCollection (IndexCollection.java:
380) - Setting log level to INFO
2023-09-19 10:56:54,249 INFO  [main] index.IndexCollection (IndexCollection.java:
383) - Starting indexer...
2023-09-19 10:56:54,249 INFO  [main] index.IndexCollection (IndexCollection.java:
```

```
384) - ============ Loading Parameters ============
2023-09-19 10:56:54,249 INFO  [main] index.IndexCollection (IndexCollection.java:
385) - DocumentCollection path: data/msmarco_passage/collection_jsonl
2023-09-19 10:56:54,250 INFO  [main] index.IndexCollection (IndexCollection.java:
386) - CollectionClass: JsonCollection
2023-09-19 10:56:54,250 INFO  [main] index.IndexCollection (IndexCollection.java:
387) - Generator: DefaultLuceneDocumentGenerator
2023-09-19 10:56:54,251 INFO  [main] index.IndexCollection (IndexCollection.java:
388) - Threads: 9
2023-09-19 10:56:54,251 INFO  [main] index.IndexCollection (IndexCollection.java:
389) - Language: en
2023-09-19 10:56:54,251 INFO  [main] index.IndexCollection (IndexCollection.java:
390) - Stemmer: porter
2023-09-19 10:56:54,252 INFO  [main] index.IndexCollection (IndexCollection.java:
391) - Keep stopwords? false
2023-09-19 10:56:54,252 INFO  [main] index.IndexCollection (IndexCollection.java:
392) - Stopwords: null
2023-09-19 10:56:54,252 INFO  [main] index.IndexCollection (IndexCollection.java:
393) - Store positions? true
2023-09-19 10:56:54,253 INFO  [main] index.IndexCollection (IndexCollection.java:
394) - Store docvectors? true
2023-09-19 10:56:54,253 INFO  [main] index.IndexCollection (IndexCollection.java:
395) - Store document "contents" field? false
2023-09-19 10:56:54,253 INFO  [main] index.IndexCollection (IndexCollection.java:
396) - Store document "raw" field? true
2023-09-19 10:56:54,256 INFO  [main] index.IndexCollection (IndexCollection.java:
397) - Additional fields to index: []
2023-09-19 10:56:54,256 INFO  [main] index.IndexCollection (IndexCollection.java:
398) - Optimize (merge segments)? false
2023-09-19 10:56:54,257 INFO  [main] index.IndexCollection (IndexCollection.java:
399) - Whitelist: null
2023-09-19 10:56:54,257 INFO  [main] index.IndexCollection (IndexCollection.java:
400) - Pretokenized?: false
2023-09-19 10:56:54,258 INFO  [main] index.IndexCollection (IndexCollection.java:
401) - Index path: indexes/lucene-index-msmarco-passage
2023-09-19 10:56:54,264 INFO  [main] index.IndexCollection (IndexCollection.java:
481) - ============ Indexing Collection ============
2023-09-19 10:56:54,282 INFO  [main] index.IndexCollection (IndexCollection.java:
468) - Using DefaultEnglishAnalyzer
2023-09-19 10:56:54,283 INFO  [main] index.IndexCollection (IndexCollection.java:
469) - Stemmer: porter
2023-09-19 10:56:54,283 INFO  [main] index.IndexCollection (IndexCollection.java:
470) - Keep stopwords? false
2023-09-19 10:56:54,284 INFO  [main] index.IndexCollection (IndexCollection.java:
471) - Stopwords file: null
2023-09-19 10:56:54,551 INFO  [main] index.IndexCollection (IndexCollection.java:
510) - Thread pool with 9 threads initialized.
2023-09-19 10:56:54,552 INFO  [main] index.IndexCollection (IndexCollection.java:
512) - Initializing collection in data/msmarco_passage/collection_jsonl
2023-09-19 10:56:54,557 INFO  [main] index.IndexCollection (IndexCollection.java:
521) - 9 files found
2023-09-19 10:56:54,558 INFO  [main] index.IndexCollection (IndexCollection.java:
522) - Starting to index...
2023-09-19 10:57:54,584 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 450,000 documents indexed
2023-09-19 10:58:54,586 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 1,170,000 documents indexed
2023-09-19 10:59:54,588 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 1,900,000 documents indexed
2023-09-19 11:00:54,589 INFO  [main] index.IndexCollection (IndexCollection.java:
```

```
536) - 0.00% of files completed, 2,630,000 documents indexed
2023-09-19 11:01:54,590 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 3,310,000 documents indexed
2023-09-19 11:02:54,591 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 4,060,000 documents indexed
2023-09-19 11:03:54,601 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 4,770,000 documents indexed
2023-09-19 11:04:54,605 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 5,390,000 documents indexed
2023-09-19 11:05:54,606 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 5,960,000 documents indexed
2023-09-19 11:06:54,608 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 6,630,000 documents indexed
2023-09-19 11:07:54,609 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 0.00% of files completed, 7,240,000 documents indexed
2023-09-19 11:08:07,175 DEBUG [pool-2-thread-3] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs08.json: 841823 docs ad
ded.
2023-09-19 11:08:54,614 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 11.11% of files completed, 7,821,823 documents indexed
2023-09-19 11:09:54,615 INFO  [main] index.IndexCollection (IndexCollection.java:
536) - 11.11% of files completed, 8,391,823 documents indexed
2023-09-19 11:10:13,866 DEBUG [pool-2-thread-9] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs07.json: 1000000 docs a
dded.
2023-09-19 11:10:24,281 DEBUG [pool-2-thread-1] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs00.json: 1000000 docs a
dded.
2023-09-19 11:10:26,374 DEBUG [pool-2-thread-4] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs01.json: 1000000 docs a
dded.
2023-09-19 11:10:28,703 DEBUG [pool-2-thread-2] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs02.json: 1000000 docs a
dded.
2023-09-19 11:10:29,811 DEBUG [pool-2-thread-8] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs03.json: 1000000 docs a
dded.
2023-09-19 11:10:30,773 DEBUG [pool-2-thread-6] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs04.json: 1000000 docs a
dded.
2023-09-19 11:10:33,444 DEBUG [pool-2-thread-5] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs05.json: 1000000 docs a
dded.
2023-09-19 11:10:36,790 DEBUG [pool-2-thread-7] index.IndexCollection$LocalIndexe
rThread (IndexCollection.java:345) - collection_jsonl/docs06.json: 1000000 docs a
dded.
2023-09-19 11:12:01,306 INFO  [main] index.IndexCollection (IndexCollection.java:
578) - Indexing Complete! 8,841,823 documents indexed
2023-09-19 11:12:01,307 INFO  [main] index.IndexCollection (IndexCollection.java:
579) - ============ Final Counter Values ============
2023-09-19 11:12:01,307 INFO  [main] index.IndexCollection (IndexCollection.java:
580) - indexed:        8,841,823
2023-09-19 11:12:01,307 INFO  [main] index.IndexCollection (IndexCollection.java:
581) - unindexable:            0
2023-09-19 11:12:01,308 INFO  [main] index.IndexCollection (IndexCollection.java:
582) - empty:                  0
2023-09-19 11:12:01,308 INFO  [main] index.IndexCollection (IndexCollection.java:
583) - skipped:                0
2023-09-19 11:12:01,308 INFO  [main] index.IndexCollection (IndexCollection.java:
584) - errors:                 0
```

```
2023-09-19 11:12:01,324 INFO  [main] index.IndexCollection (IndexCollection.java:
587) - Total 8,841,823 documents indexed in 00:15:07
```

2. Download and read the query file

You will rank MSMARCO passages for this set of queries.

In [18]:
```
!wget http://gem.cs.ru.nl/IR-Course/queries.txt

queries = dict()
with open("queries.txt", "r") as f:
    for line in f:
        cols = line.split("\t")
        queries[cols[0].strip()] = cols[1].strip()
```

```
--2023-09-19 11:12:01--  http://gem.cs.ru.nl/IR-Course/queries.txt
Resolving gem.cs.ru.nl (gem.cs.ru.nl)... 131.174.31.31
Connecting to gem.cs.ru.nl (gem.cs.ru.nl)|131.174.31.31|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2275 (2.2K) [text/plain]
Saving to: ‘queries.txt.1’

queries.txt.1       100%[===================>]   2.22K  --.-KB/s    in 0s

2023-09-19 11:12:02 (291 MB/s) - ‘queries.txt.1’ saved [2275/2275]
```

3. Implement the retrieval model

You will implement language model with Jelinek-Mercer (JM) smoothing:

$$score(q,d) = \sum_{t \in q} log((1-\lambda)\frac{c(t,d)}{|d|} + \lambda\frac{c(t,C)}{|C|}),$$

where $c(t,d)$ and $c(t,C)$ represent frequency of a term in a document and collection, respectively.

Notes about your implementation:

- Skip a term if it does not exist in the whole collection. This avoids $log(0)$.
- Make sure to use the right form of a query (analyzed vs. not analyzed)
- Use natural logarithm

3.1. Obtain collection length

In this code, the global variable `len_C` denotes collection length.

In [19]:
```
from pyserini.index.lucene import IndexReader

global len_C

# =======Your code=======
index_reader = IndexReader('indexes/lucene-index-msmarco-passage')
```

```
len_C = index_reader.stats()['total_terms']
# ======================
```

Run this to test your code. If everything is correct, you should not get errors here.

In [20]:
```
assert len_C == 352316036
```

3.2. Obtain document length

Here you need compute the length of document (as it is stored in the index).

Hint: You first need to get the document vector from your Pyserini index. Consult Pyseniri documentation to find the right function.

In [21]:
```
def len_doc(d):
    # =======Your code=======
    doc_vec = index_reader.get_document_vector(d)
    len_d = sum(doc_vec.values())
    # ======================
    return len_d
```

In [22]:
```
# Test your code
assert len_doc("2674124") == 31
```

3.3. Obtain collection frequency of a term

Obtain number of times a term appers in the whole collection.

In [23]:
```
def coll_freq(t):
    # =======Your code=======
    df, cf = index_reader.get_term_counts(t)
    # ======================
    return cf
```

In [24]:
```
# Test your code
assert coll_freq("record") == 226439
```

3.4. Obtain term frequency

Obtain number of times a term appears in a document.

In [25]:
```
def term_freq(t, d):
    # =======Your code=======
    doc_vec = index_reader.get_document_vector(d)
    tf = doc_vec[t] if t in doc_vec else 0
    # ======================
    return tf
```

In [26]:
```
# Test your code
assert term_freq("record", "2674124") == 2
assert term_freq("presence", "2674124") == 0
```

3.5. Compute JM-smoothed probability for a single term

Here, you need to implement the following formula:

$$P_{JM}(t,d) = (1-\lambda)\frac{c(t,d)}{|d|} + \lambda\frac{c(t,C)}{|C|}$$

In [27]:
```
def prob_t_Md(t, d, lambd):
    # =======Your code=======
    p_t_Md = ((1-lambd)*(term_freq(t,d)/len_doc(d)))+(lambd*(coll_freq(t)/len_C)
    # ======================
    return p_t_Md
```

In [28]:
```
# Test your code
assert prob_t_Md("record", "2674124", 0.1) == 0.05812878768549357
assert prob_t_Md("darcig", "2674124", 0.1) == 0
```

3.6. Compute JM-smoothed probability for a query

In [29]:
```
import math

def score_doc(q, d, lambd):
    # =======Your code=======
    p_q_Md = 0
    for term in q:
      # Skip term analysis:
      df, cf = index_reader.get_term_counts(term, analyzer=None)
      if cf == 0 :
        continue
      p_q_Md += math.log(prob_t_Md(term,d,lambd))
    # ======================
    return p_q_Md
```

In [33]:
```
q1 = index_reader.analyze("are naturalization records public")
q2 = index_reader.analyze("kemeet land")
doc = "2674124"
assert score_doc(q1, doc, 0.1) == -9.227787624348021
assert score_doc(q2, doc, 0.1) == -10.254756777887694
```

4. Rank documents for the given queries

Ranking is done in two steps:

1. First pass retrieval: Use a fast ranker (i.e., Pyserini LuceneSearcher) ro rank all documents for a given query.
2. Second pass retrieval: Re-rank top-100 documents from the 1st pass retrieval using your retrieval model. This is to make the ranking process efficient.

Notes:

- You need to change the default values of LuceneSearcher functions to obtain top-100 documents
- Set the value of lambda to 0.1

- Store your final ranking results in the `results` variable. Every item in the `results` list is a list containing queryID, documentID, and score. This is an example how the content of results should look like:

```
[['23849', '4348282', -10.65],  ['23849', '7119957', -12.63],
['23849', '', -17.687729001682484],  ...]
```

In [41]:
```
from pyserini.search.lucene import LuceneSearcher
lambd = 0.1
results = []
searcher = LuceneSearcher("indexes/lucene-index-msmarco-passage")
for qid, q in queries.items():
    # =======Your code=======
    hits = searcher.search(q,100)
    analyzed_q = index_reader.analyze(q)
    for hit in hits:
      score = score_doc(analyzed_q,hit.docid,lambd)
      results.append([qid,hit.docid,score])
    # ======================
```

In [38]:
```
# Test your code
print(round(sum([item[2] for item in results]), 3))
assert round(sum([item[2] for item in results]), 3) == -160109.875
```

```
-160109.875
```

Write your results into a file. Submit this file together with the completed notebook.

In [43]:
```
# check duplicates
check = set()
for res in results:
    if ((res[0], res[1])) in check:
        raise Exception("Error: Duplicate query-doc is found", res[0], res[1])
    check.add((res[0], res[1]))

# write results in a file
output_str = "\n".join([l[0] + "\tQ0\t" + l[1] + "\t0\t" + str(l[2]) + "\tlm_jm"
open("lm_jm.run", "w").write(output_str)
```

Out[43]: 246907

Handing in

Submit the result file (ranked documents), the filled-in notebook, and the pdf version of your notebook:

- The result file should be named STUDENTNUMBER_FIRSTNAME_LASTNAME_lm_jm.run
- The notebook should be named STUDENTNUMBER_FIRSTNAME_LASTNAME_retrieval.ipynb
- The pdf version of your notebook should be named STUDENTNUMBER_FIRSTNAME_LASTNAME_retrieval.pdf

In [39]: