HA3: Neural ranking

- Install SentenceTransformers

https://www.sbert.net/
https://huggingface.co/sentence-transformers

- Read documentation, look at examples, pay special attention to search section

https://www.sbert.net/examples/applications/semantic-search/README.html

https://www.sbert.net/examples/applications/retrieve_rerank/README.html

- Use one of the models trained on MS MARCO

https://www.sbert.net/docs/pretrained-models/msmarco-v5.html

Performance

Performance is evaluated on TREC-DL 2019 and TREC-DL 2020, which are a query-passage retrieval task where multiple queries have been annotated as with their relevance with respect to the given query. Further, we evaluate on the MS Marco Passage Retrieval dataset.

Approach	MRR@10 (MS Marco Dev)	NDCG@10 (TREC DL 19 Reranking)	NDCG@10 (TREC DL 20 Reranking)	Queries (GPU / CPU
Models tuned with normalized embeddings				
msmarco-MiniLM-L6-cos-v5	32.27	67.46	64.73	18,000 / 750
msmarco-MiniLM-L12-cos-v5	32.75	65.14	67.48	11,000 / 400
msmarco-distilbert-cos-v5	33.79	70.24	66.24	7,000 / 350
multi-qa-MiniLM-L6-cos-v1		65.55	64.66	18,000 / 750
multi-qa-distilbert-cos-v1		67.59	66.46	7,000 / 350
multi-qa-mpnet-base-cos-v1		67.78	69.87	4,000 / 170
Models tuned for dot-product				
msmarco-distilbert-base-tas-b	34.43	71.04	69.78	7,000 / 350
msmarco-distilbert-dot-v5	37.25	70.14	71.08	7,000 / 350
msmarco-bert-base-dot-v5	38.08	70.51	73.45	4,000 / 170
multi-qa-MiniLM-L6-dot-v1		66.70	65.98	18,000 / 750
multi-qa-distilbert-dot-v1		68.05	70.49	7,000 / 350
multi-qa-mpnet-base-dot-v1		70.66	71.18	4,000 / 170

Task

- Re-rank top20 documents returned for WikiIR test queries by Elasticsearch (or another retriever used in HA2)
- Use cosine similarity between query and document embeddings to rank documents
- Evaluate new rankings using P@10, p@20, MAP@20

Task*

- Re-rank documents based on a combination of BM25 scores and cosine similarity of query/document embeddings
- Sample 100-500 queries from **train** subset
- Get top50 documents for each query using Elasticsearch (BM25)
- Min-max normalize BM25 scores, so they are in the range [0,1]
- Get cosines for query/document embeddings
- Find an alpha that maximizes MAP@20 on **train** data
  *alpha*BM25 + (1-alpha)*q_d_cosine_similarity*
- Apply the formula to the **test** data (again, to top50)
- Evaluate new rankings using P@10, p@20, MAP@20