Nama : Muhamad Feriyanto

Kelas : 3SD1

NIM : 222011347

Praktikum 8

Hasil Perbandingan Perankingan dengan Okapi BM25 vs VSM

Query = "*vaksin corona jakarta*"

Corpus = folder berita

Okapi BM25

```
In [77]: from collections import OrderedDict
         def exact_top_k_bm25(doc_dict, rank_score, k):
             relevance_scores = {}
             i = 0
             for doc_id in doc_dict.keys():
                 relevance_scores[doc_id] = rank_score[i]
                 i = i + 1

         #         sorted_value = OrderedDict(sorted(relevance_scores.items(), key=lambda x: x[1], reverse = True))
         #         top_k = {j: sorted_value[j] for j in list(sorted_value)[:k]}

                 sorted_value = OrderedDict(sorted(relevance_scores.items(),key = lambda x:x[1], reverse = True))
                 top_k = {j : sorted_value[j] for j in list(sorted_value)[:k]}
             return top_k

In [80]: exact_top_k_bm25(no_sw, querinisasi("vaksin corona jakarta"), 3)

Out[80]: {'berita3': 1.068210132282897,
          'berita2': 0.8332116429494796,
          'berita5': 0.544069042444546}
```

VSM

```
In [114]: # Menyimpan skor kemiripan dalam suatu list
          from collections import OrderedDict
          def exact_top_k_vsm(doc_dict, TD, q,k):
              relevance_scores = {}
              i = 0
              for doc_id in doc_dict.keys():
                  relevance_scores[doc_id] = cosine_sim(q, TD[:,i])
                  i  = i + 1

              sorted_value = OrderedDict(sorted(relevance_scores.items(),key = lambda x:x[1], reverse = True))
              top_k = {j : sorted_value[j] for j in list(sorted_value)[:k]}
              return top_k
          top_3_vsm = exact_top_k_vsm(no_sw, TD, TQ[:, 0], 3)
          print(top_3_vsm)

          {'berita2': 0.9937562849886573, 'berita3': 0.9932276247792821, 'berita1': 0.9918235081246563}
```

Tabel Perbandingan

Query = “vaksin corona jakarta”

Corpus = folder berita

Ranking	Okapi BM25		VSM	
	Dokumen	Nilai	Dokumen	Nilai
1	Berita 3	1,0682	Berita 2	0,9938
2	Berita 2	0,8332	Berita 3	0,9932
3	Berita 5	0,5441	Berita 1	0,9918

Dari tabel berikut terlihat bahwa metode perankingan diambil 3 rangking dengan query = “vaksin corona jakarta” pada folder berita memberikan urutan yang berbeda (namun bisa dibilang mirip, karena keterbatasan term yang ada). Pada metode Okapi BM25 memberikan *retrieve* 3 dokumen teratas yakni ‘Berita 3’, ‘Berita 2’, dan ‘Berita 5’. Sedangkan pada metode VSM me-*retrieve* 3 dokumen teratas ‘Berita 2’, ‘Berita 3’, dan ‘Berita 1’.