# MODUL 7: PROBABILISTIC INFORMATION RETRIEVAL

## 7.1 Deskripsi Singkat

Okapi BM25 atau yang biasa disebut dengan BM25 dikembangkan oleh City University London dan berdasarkan pada model probabilistik dasar yang mengurutkan dokumen dalam urutan menurun terhadap nilai relevansi sebuah dokumen terhadap informasi yang dibutuhkan. BM25 meranking dokumen berdasarkan probabilitas dan menggunakan term frequency untuk meranking similarity.

## 7.2 Tujuan Praktikum

1. Dapat melakukan pemeringkatan dokumen dengan salah satu model probalibistic information retrieval, yaitu BM25.

## 7.3 Material Praktikum

Tidak ada

## 7.4 Kegiatan Praktikum

Lakukan instalasi library rank_bm25 dengan pip.

```
pip install rank_bm25
```

Kemudian tulis kode berikut untuk menghitung skor relevansi menggunakan BM25 dari 10 dokumen yang sebelumnya digunakan pada modul 5.

```
from rank_bm25 import BM25Okapi

def tokenisasi(text):
    tokens = text.split(" ")
    return tokens

def stemming(text):
    from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
    # create stemmer
    factory = StemmerFactory()
    stemmer = factory.create_stemmer()
    # stemming process
    output   = stemmer.stem(text)
    return output

def stemming_sentence(text):
    output = ""
    for token in tokenisasi(text):
        output   = output + stemming(token) + " "
    return output[:-1]
```

```
doc_dict_raw = {}
doc_dict_raw['doc1'] = "pengembangan sistem informasi penjadwalan"
doc_dict_raw['doc2'] = "pengembangan model analisis sentimen berita"
doc_dict_raw['doc3'] = "analisis sistem input output"
doc_dict_raw['doc4'] = "pengembangan sistem informasi akademik
universitas"
doc_dict_raw['doc5'] = "pengembangan sistem cari berita ekonomi"
doc_dict_raw['doc6'] = "analisis sistem neraca nasional"
doc_dict_raw['doc7'] = "pengembangan sistem informasi layanan
statistik"
doc_dict_raw['doc8'] = "pengembangan sistem pencarian skripsi di
universitas"
doc_dict_raw['doc9'] = "analisis sentimen publik terhadap pemerintah"
doc_dict_raw['doc10'] = "pengembangan model klasifikasi sentimen
berita"

doc_dict = {}
for doc_id,doc in doc_dict_raw.items():
    doc_dict[doc_id] = stemming_sentence(doc)

tokenized_corpus = [tokenisasi(doc_dict[doc_id]) for doc_id in
doc_dict]

query = "sistem informasi statistik"
tokenized_query = tokenisasi(query)

bm25 = BM25Okapi(tokenized_corpus)

doc_scores = bm25.get_scores(tokenized_query)
print(doc_scores)
```

Kode di atas akan menampilkan skor probabilitas dokumen terhadap query ”sistem informasi statistik” dengan model BM25. Anda dapat melihat kode lengkap dari fungsi BM25Okapi pada link github berikut.

https://github.com/dorianbrown/rank_bm25/blob/master/rank_bm25.py

Kemudian jalankan kembali kode di atas dengan mengubah query menjadi ”sistem”, ”informasi”, dan ”statistik”. Bandingkan skor BM25 dari keempat query tersebut. Apa yang dapat Anda simpulkan?

Perhatikan kembali skor yang dihasilkan untuk query ”sistem informasi statistik”. Buat fungsi untuk mengembalikan top k dokumen.

```
from collections import OrderedDict
def exact_top_k(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]}
    return top_k
```

Jalankan kode di atas untuk k=3. Sebutkan tiga dokumen dengan skor BM25 tertinggi. Bandingkan dengan hasil top 3 dari perankingan dokumen dengan Vector Space Model pada modul 5.

## 7.5 Penugasan

1. Tuliskan laporan praktikum yang merangkum kegiatan praktikum yang telah Anda lakukan pada kegiatan 7.4.
2. Buat fungsi main untuk menampilkan 3 list dokumen yang terurut berdasarkan BM25 pada folder ”berita” dengan query ”vaksin corona jakarta”. Bandingkan dengan hasil perankingan cosine similarity pada modul 5. Jelaskan kode tersebut dalam laporan praktikum.
3. Pelajari tentang Pyserini pada paper https://dl.acm.org/doi/10.1145/3404835.3463238 dan Elastic Search.