MODUL 3: BOOLEAN RETRIEVAL DAN INVERTED INDEX

3.1 Deskripsi Singkat

Pada *information retrieval*, indeks dibuat terlebih dahulu untuk menghindari pencarian secara linier dari teks pada setiap *query*. *Inverted index* adalah suatu indeks dimana *term* di hubungkan dengan lokasi dokumen dimana *term* tersebut berada (*posting lists*).

Boolean Retrieval merupakan proses pencarian informasi dari *query* yang menggunakan ekspresi *Boolean*, yaitu menggunakan operator logika AND, OR dan NOT. Hasil *boolean retrieval* yaitu dokumen relevan (nilai biner: 1) atau dokumen tidak relevan (nilai biner: 0). Dalam pengerjaan operator *boolean* (AND, NOT, OR) ada urutan pengerjaannya (*operator precedence*), yaitu memprioritaskan yang berada dalam kurung (), baru selanjutnya NOT, AND, dan OR.

3.2 Tujuan Praktikum

Setelah praktikum pada modul 3 ini diharapkan mahasiswa mempunyai kompetensi sebagai berikut.

1) Dapat membuat inverted index
2) Dapat melakukan boolean retrieval dengan memproses boolean query

3.3 Material Praktikum

Tidak ada

3.4 Kegiatan Praktikum

A. Inverted Index

Diketahui terdapat 3 dokumen dengan *term* masing-masing berdasarkan hasil tokenisasi pada Modul 2C. Kemudian didapatkan *term* pada korpus (keseluruhan koleksi dokumen) yang disimpan di suatu list 2D bernama `corpus_term`.

```
doc1_term = ["pengembangan", "sistem", "informasi",
"penjadwalan"]
doc2_term = ["pengembangan", "model", "analisis", "sentimen",
"berita"]
doc3_term = ["analisis", "sistem", "input", "output"]

corpus_term = [doc1_term, doc2_term, doc3_term]
```

Berikut adalah kode untuk mendapatkan inverted index dengan term pada korpus tersebut. Tambahkan fungsi stemming sehingga term yang tersimpan di inverted index adalah term yang berupa kata dasar dengan memanggil fungsi `stemming` (sudah dipelajari di Modul 2).

```
inverted_index = {}

for i in range(len(corpus_term)):
      for item in corpus_term[i]:
            item = stemming(item)
            if item not in inverted_index:
                  inverted_index[item] = []
            if (item in inverted_index) and ((i+1) not in
inverted_index[item]):
                  inverted_index[item].append(i+1)
print(inverted_index)
```

Perhatikan isi dari variabel `inverted index`.

B. Boolean Retrieval

Untuk melakukan boolean retrieval, buat terlebih dahulu fungsi operator AND berikut.

```
def AND(posting1, posting2):
    p1 = 0
    p2 = 0
    result = list()
    while p1 < len(posting1) and p2 < len(posting2):
        if posting1[p1] == posting2[p2]:
            result.append(posting1[p1])
            p1 += 1
            p2 += 1
        elif posting1[p1] > posting2[p2]:
            p2 += 1
        else:
            p1 += 1
    return result
```

Kemudian panggil fungsi di atas dengan kode berikut untuk melakukan boolean query `sistem AND analisis`

```
AND(inverted_index['sistem'],inverted_index['analisis'])
```

Buat lagi kode dengan memanggil fungsi AND untuk melakukan boolean query berikut.

1. sistem AND informasi
2. sistem AND informasi AND jadwal

Perhatikan hasilnya. Apakah sudah sesuai?

Selanjutnya buat fungsi untuk operator OR.

```
def OR(posting1, posting2):
    p1 = 0
    p2 = 0
    result = list()
    while p1 < len(posting1) and p2 < len(posting2):
        if posting1[p1] == posting2[p2]:
            result.append(posting1[p1])
            p1 += 1
            p2 += 1
        elif posting1[p1] > posting2[p2]:
            result.append(posting2[p2])
            p2 += 1
        else:
            result.append(posting1[p1])
            p1 += 1
    while p1 < len(posting1):
        result.append(posting1[p1])
        p1 += 1
    while p2 < len(posting2):
        result.append(posting2[p2])
        p2 += 1
    return result
```

Kemudian buat kode dengan memanggil fungsi di atas untuk melakukan boolean query berikut.

1. sistem OR analisis
2. sistem OR informasi

Perhatikan hasilnya. Apakah sudah sesuai?

Selanjutnya buat fungsi untuk operator NOT.

```
def NOT(posting):
    result = list()
    i = 0
    for item in posting:
        while i < item:
            result.append(i)
            i += 1
        else:
            i += 1
    else:
        while i < NUM_OF_DOCS:
            result.append(i)
            i += 1
    return result
```

Kemudian buat kode dengan memanggil fungsi di atas untuk melakukan boolean query berikut.

1. NOT sistem
2. NOT informasi

Perhatikan hasilnya. Apakah sudah sesuai?

Pastikan Anda memahami alur pemrosesan boolean query di atas.

3.5 Penugasan

1. Menggunakan sekumpulan dokumen pada folder ”berita”, setelah dilakukan preprocessing pada penugasan Modul 2, tambahkan kode untuk menghasilkan inverted index dengan output berupa *term* dan daftar lokasinya (*posting lists*).
2. Kemudian tambahkan kode untuk melakukan boolean retrieval dari inverted index pada Penugasan 1. Perhatikan daftar dokumen yang dikembalikan ketika menuliskan query berikut.
   a. corona
   b. covid
   c. vaksin
   d. corona OR covid
   e. vaksin AND corona
   f. vaksin AND corona AND pfizer
   g. NOT vaksin
3. Modifikasi kode fungsi AND sehingga dapat melakukan optimasi query untuk list postings berikut:

```
def AND_optimized(postings):
    …
```

   Lalu implementasikan untuk query `vaksin AND corona AND pfizer`.