KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

PROGRAM STUDI ILMU KOMPUTER DEPARTEMEN ILMU KOMPUTER DAN ELEKTRONIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS GADJAH MADA

TEMU KEMBALI INFORMASI

Tugas 5 Metode Pembuatan Indeks

DISUSUN OLEH:

ADAM YOGISYAH PUTRA	**20/455439/PA/19654**
MUHAMMAD ARSYA PUTRA	**20/462186/PA/20158**
HIZKYA FIRSTADIPA HARTOKO	**20/455447/PA/19662**

DOSEN:

Dr. Lukman Heryawan, S.T., M.T.

I. Metode

Metode yang digunakan pada pembuatan indeks ini adalah membuat Inverted Index Matrix yang mengetahui frekuensi setiap kata muncul pada suatu dokumen, dan di dokumen mana saja

A. Tokenisasi

Proses pemecahan kalimat-kalimat/dokumen ke bentuk yang lebih kecil bernama token.

B. Penggabungan token setiap kolom menjadi satu dokumen

Penggabungan token ini bertujuan agar setiap kata yang terdapat dalam dokumen tergabung menjadi satu untuk mempermudah proses selanjutnya.

C. Pembuatan Formation of Vector

Token yang sudah digabungkan kemudian diambil termnya dan letak asal dokumennya, sehingga membuat suatu formasi term dan id dokumen

D. Pengurutan term

Term-term yang berada di document term diurutkan secara lexicographical untuk mempermudah proses pembuatan inverted index

E. Inverted Index

Inverted index adalah hasil akhir dari proses pembuatan index ini. Frekuensi kemunculan term dalam dokumen dapat diketahui melalui matriks inverted index, sekaligus letak term tersebut pada indeks

II. Proses Pembuatan Indeks

A. Import dataset

```
import pandas as pd
import numpy as np

url = 'https://raw.githubusercontent.com/HizkiFirst/tki-scraping-obat/main/clean-dataset.csv'
df = pd.read_csv(url)
df.head(5)
```

	product_name	category	deskripsi	indikasi_umum	
0	cetirizine sirup 60 ml	alergi	cetirizine merupakan antihistamin yang secara...	informasi obat ini hanya untuk kalangan medis...	peng
1	ocuson 10 tablet	alergi	ocuson tablet adalah obat kombinasi kortikost...	informasi obat ini hanya untuk kalangan medis...	peng
2	interhistin 50 mg 10 tablet	alergi	interhistin 50 mg tablet merupakan obat denga...	informasi obat ini hanya untuk kalangan medis...	peng
3	ctm 4 mg 12 tablet	alergi	ctm 4 mg tablet merupakan obat anti alergi ya...	obat ini digunakan untuk mengatasi gejala ale...	dev
4	dexteem plus 10 tablet	alergi	dexteem plus tablet adalah obat kombinasi kor...	informasi obat ini hanya untuk kalangan medis...	peng

B. Tokenisasi

Tokenisasi

```
df_parse = df.copy()

cols = ["product_name","category","deskripsi","indikasi_umum","dosis","aturan_pakai"]
for col in cols:
  df_parse[col] = df_parse[col].str.split()

df_parse
```

	product_name	category	deskripsi	indikasi_umum	
0	[cetirizine, sirup, 60, ml]	[alergi]	[cetirizine, merupakan, antihistamin, yang, se...	[informasi, obat, ini, hanya, untuk, kalangan,...	
1	[ocuson, 10, tablet]	[alergi]	[ocuson, tablet, adalah, obat, kombinasi, kort...	[informasi, obat, ini, hanya, untuk, kalangan,...	
2	[interhistin, 50, mg, 10, tablet]	[alergi]	[interhistin, 50, mg, tablet, merupakan, obat,...	[informasi, obat, ini, hanya, untuk, kalangan,...	
3	[ctm, 4, mg, 12, tablet]	[alergi]	[ctm, 4, mg, tablet, merupakan, obat, anti, al...	[obat, ini, digunakan, untuk, mengatasi, gejal...	[dev
4	[dexteem, plus, 10, tablet]	[alergi]	[dexteem, plus, tablet, adalah, obat, kombinas...	[informasi, obat, ini, hanya, untuk, kalangan,...	

C. Penggabungan token setiap kolom menjadi satu dokumen

Merging setiap kolom menjadi satu dokumen terms

```
df_mergeparse = pd.DataFrame(columns = ["document"])

temp = df_parse["product_name"]

cols = ["category","deskripsi","indikasi_umum","dosis","aturan_pakai"]
for col in cols:
  temp = temp + df_parse[col]
  df_mergeparse["document"] = temp
df_mergeparse
```

	document
0	[cetirizine, sirup, 60, ml, alergi, cetirizine...
1	[ocuson, 10, tablet, alergi, ocuson, tablet, a...
2	[interhistin, 50, mg, 10, tablet, alergi, inte...
3	[ctm, 4, mg, 12, tablet, alergi, ctm, 4, mg, t...
4	[dexteem, plus, 10, tablet, alergi, dexteem, p...

D. Pembuatan Formation of Vector

```
[ ] df_term = pd.DataFrame(columns = ["term","docID"])

size = df_mergeparse.shape[0]
for y in range(size):
  row = df_mergeparse['document'].iloc[y]

  for item in row:
   df_term= df_term.append({"term":item, "docID":y}, ignore_index=True)
```

+ Code + Text

```
[ ] df_term.head(10)
```

	term	docID
0	cetirizine	0
1	sirup	0
2	60	0
3	ml	0
4	alergi	0

E. Pengurutan term

QuickSort Term secara Alfabetik

```
[ ]  df_term_sorted = df_term.sort_values(by=['term'])
```

F. Inverted Index

Inverted Index

```
[ ]  #inisialisasi inverted index
     df_invertedIndex = pd.DataFrame(columns = ["term","docFrequency",'postingList'])
```

```
#mengcopy nilai term yang unik
size = df_term_sorted.shape[0]
df_invertedIndex['term'] = df_term_sorted['term'].unique()

df_invertedIndex.tail(50)
```

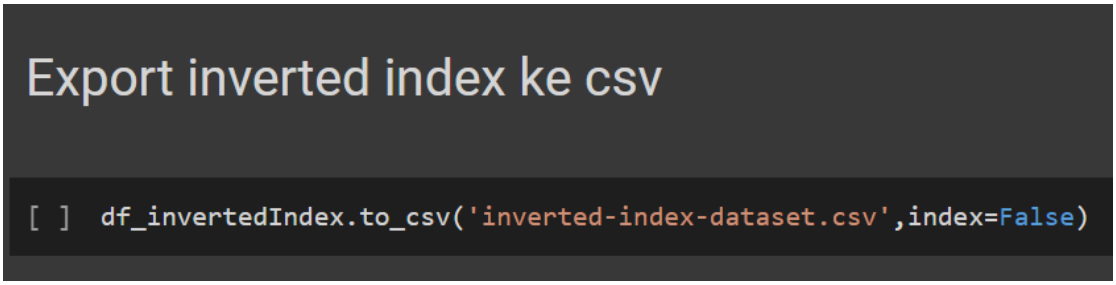
	term	docFrequency	postingList
3927	vitalitas	NaN	NaN
3928	vitamin	NaN	NaN
3929	vitan	NaN	NaN
3930	vitro	NaN	NaN

```
#inisialisasi nilai
df_invertedIndex['docFrequency']=0
df_invertedIndex.tail(50)
```

	term	docFrequency	postingList
3927	vitalitas	0	NaN
3928	vitamin	0	NaN
3929	vitan	0	NaN

```
#logic df inverted index
size = df_term_sorted.shape[0]
invertedIndexCounter = 0
temp_list = []
for i in range(size):
  if(df_term_sorted['term'].iloc[i] == df_invertedIndex['term'].iloc[invertedIndexCounter] ):
    df_invertedIndex['docFrequency'].iloc[invertedIndexCounter] += 1
    temp_list.append(df_term_sorted['docID'].iloc[i])
    df_invertedIndex['postingList'].iloc[invertedIndexCounter] = temp_list
  else:
    temp_list = []
    invertedIndexCounter+=1
    df_invertedIndex['docFrequency'].iloc[invertedIndexCounter] += 1
    temp_list.append(df_term_sorted['docID'].iloc[i])
    df_invertedIndex['postingList'].iloc[invertedIndexCounter] = temp_list
```

G. Export dataset



III. Hasil

Hasil yang diperoleh adalah matriks inverted index yang telah digabungkan dan diurutkan yang terdiri dari term dictionary, frekuensi kemunculan di setiap dokumen, dan posting list yang berisi document ID dimana term tersebut muncul.

Link indeks:
https://github.com/HizkiFirst/tki-scraping-obat/blob/main/inverted-index-dataset.csv

IV. Lampiran

Link Repository: https://github.com/HizkiFirst/tki-scraping-obat

Link Google Collab Source Code:
https://colab.research.google.com/drive/1LLpyk1KC3retW3YzhQMabFoTOVaYRGPG?usp=sharing