FAKULTAS
ILMU
KOMPUTER

CSCE604135 • Perolehan Informasi
Semester Ganjil 2021/2022
Fakultas Ilmu Komputer, Universitas Indonesia

Tugas 1: Dasar Pengolahan Teks Melalui Korpus

Tenggat Waktu: 19 September 2021, 23.55 WIB

Ketentuan:

1. Dataset yang digunakan pada tugas ini telah disediakan di SCeLe.
2. Buatlah program Jupyter Notebook yang menjawab pertanyaan sesuai dengan perintah soal yang disediakan.
3. Program Jupyter Notebook yang telah dibuat dikumpulkan dengan format penamaan **TugasX_NPM_Nama.ipynb**
   Contoh: Tugas1_1706979341_Lulu Ilmaknun Qurotaini.ipynb
4. Kumpulkan dokumen tersebut pada submisi yang telah disediakan di SCeLe sebelum tanggal **19 September 2021, 23.55 WIB**. Keterlambatan pengumpulan akan dikenakan pinalti.
5. Tugas ini dirancang sebagai **tugas mandiri**. Plagiarisme tidak diperkenankan dalam bentuk apapun. Adapun kolaborasi berupa diskusi (tanpa menyalin maupun mengambil jawaban orang lain) dan literasi masih diperbolehkan dengan mencantumkan kolaborator dan sumber.
6. Anda dibebaskan menggunakan bahasa pemrograman apa saja tetapi untuk mempermudah, kami merekomendasikan bahasa Python.

Petunjuk Pengerjaan Tugas

Pada tugas ini, Anda akan diberikan dua buah korpus yang dapat digunakan, yaitu berkas **Korpus.json** dan **Korpus_Ringkas.json**. Berkas **Korpus.json** yang diberikan merupakan berkas yang berisi informasi lengkap dari berita dalam bahasa Indonesia yang terdiri dari:

Kolom	Deskripsi
section	Kategori berita seperti ‘ekonomi’, ‘sosial’, ‘internasional’, dst. [String]
articleid	ID dari berita [int 6 bit / digit]
articletype	Tipe penampilan artikel seperti ‘singlepage’ [String]
createdate	Tanggal dipublikasikannya sebuah berita [YYYY-MM-DD HH:MM:SS]
author	Penulis berita [String]
originalTitle	Judul berita [String]
tag	Penggolongan berita berdasarkan kata kunci tertentu [String]
keywords	Kata kunci utama berita [String]
subsection	Kategori lanjutan dari ‘section’ seperti ‘Peristiwa’ [String]
content	Konten utama dari berita yang disampaikan [String]

Sedangkan berkas dari **Korpus_Ringkas.json** merupakan potongan dari korpus lengkap yang telah disesuaikan, isi dari berkas tersebut terdiri dari:

Kolom	Deskripsi
id	ID dari artikel [int]
content	Konten utama dari berita yang disampaikan [String]

Pada tugas ini Anda diminta untuk melakukan analisis korpus menggunakan korpus yang telah diberikan. Untuk soal di mana Anda diminta untuk menampilkan histogram dan *word cloud*, Anda **diperbolehkan** menggunakan *library* tambahan. (cth: matplotlib, wordcloud, etc.). Untuk proses segmentasi kalimat dan tokenisasi kata, Anda **diwajibkan** menggunakan library nltk.

Berikan dokumentasi pada implementasi Anda. ***Deliverable*** akhir merupakan **1 berkas jupyter notebook (dalam format ipynb)**, di mana pada berkas tersebut terdapat kode yang Anda tuliskan + dokumentasi (bisa berupa chart, teks, atau lainnya) + penjelasan terkait apa yang telah Anda kerjakan.

Untuk bagian A dan B, gunakan data yang terdapat pada *file* "**Korpus.json**". Kemudian, untuk bagian C dan D, gunakan data pada *file* "**Korpus_Ringkas.json**".

A - Corpus Statistics (25 Poin)

1. **[1]** Berapakah jumlah data yang terdapat pada "**Korpus.json**"?
2. **[6]** Pada soal ini, lakukan analisa untuk fitur "section" pada korpus dengan menjawab pertanyaan-pertanyaan berikut ini!
    a. **[2]** Hitunglah berapa banyak jumlah "section" yang **unik**. Sebutkan "section" apa saja yang terdapat dalam korpus tersebut!
    b. **[4]** Tampilkan jumlah data untuk setiap "section" dalam bentuk **histogram**!
3. **[12]** Pada soal ini, lakukan analisa untuk fitur "content" pada korpus dengan menjawab pertanyaan-pertanyaan berikut ini! Untuk mempermudah pengerjaan, suatu token dalam korpus dapat dianggap sebagai kata jika seluruh karakter berupa ***alphanumeric*** (A-z, 0-9).
    a. **[3]** Hitunglah jumlah kata yang **unik** dalam korpus. Catatan: Kata "Jakarta" dan "jakarta" dianggap sebagai kata yang sama.
    b. **[5]** Tampilkan 100 kata diurutkan berdasarkan jumlah kemunculan terbanyak dalam bentuk **histogram**! Kemudian, sebutkan 10 kata terbanyak dalam korpus beserta jumlahnya!
    c. **[4]** Sebutkan 10 kata **bigram** terbanyak dalam korpus!
4. **[6]** Pada soal ini, lakukan analisa untuk fitur "originalTitle" pada korpus dengan menjawab pertanyaan-pertanyaan berikut ini! Identik dengan soal sebelumnya, suatu token dalam korpus dapat dianggap sebagai kata jika seluruh karakter berupa ***alphanumeric*** (A-z, 0-9).
    a. **[2]** Berapa **rata-rata** jumlah kata yang terdapat dalam "originalTitle"?
    b. **[4]** Tampilkan ***word cloud*** untuk fitur "originalTitle" dalam korpus!

B - Regex (25 Poin)

Salah satu simbol yang unik dan sering digunakan secara langsung dalam teks adalah "*at symbol*". Pada awalnya, karakter "@" umumnya digunakan pada bidang akuntansi atau penulisan *invoice* sebagai pengganti istilah "*at a rate of*" atau dengan kata lain menyatakan harga satuan suatu barang. Saat ini, karakter "@" lebih sering digunakan pada akun Email dan akun media sosial seperti Twitter, Instagram, TikTok, dan lainnya.

Sebagai calon IR engineer yang mahir, tentunya diharapkan kita dapat membedakan penggunaan karakter "@" ini dalam konteks yang berbeda-beda. Menggunakan dataset "**Korpus.json**", khususnya fitur "content" pada korpus tersebut, **tentukan** apakah setiap karakter "@" yang ada dalam korpus **termasuk dalam penggunaan untuk jenis** akun Email, Twitter, atau Instagram!

Anda diharuskan untuk merancang **regex** (boleh lebih dari satu) untuk menjawab soal ini. Gunakan *guideline* berikut ini untuk membantu Anda. Asumsikan bahwa seluruh penggunaan karakter "@" pada korpus ini **hanya** termasuk pada jenis akun Email, Twitter, atau Instagram (tidak ada media sosial atau penggunaan lainnya).

Ketentuan Akun Email / Twitter / Instagram

1. Setiap akun pastinya hanya memiliki satu karakter "@"
2. Setiap akun juga tidak dapat diakhiri dengan tanda "."
3. Akun Email memiliki karakter "@" yang diapit oleh dua token. Dengan kata lain, suatu akun Email berbentuk [Token1]@[Token2].
    a. Setiap token pada suatu akun Email dapat terdiri dari karakter-karakter ***alphanumeric*** (A-z, 0-9), tanda ".", tanda "-", atau tanda "_"
    b. Khusus untuk Token 2, harus terdapat minimal 1 tanda "."

c. Beberapa contoh akun email **valid**:
    i. akun-ir@gmail.com
    ii. akun__ir@cs.ui__
    iii. ir.4.life@my-hobbies.uk.id-

d. Beberapa contoh akun email **tidak valid**:
    i. not$valid_@cs.ui
    ii. m4ta_bat1n@magician__
    iii. raiden-shogun-op@genshin.life.

4. Akun media sosial memiliki karakter "@" di awal dan diikuti oleh satu token. Dengan kata lain, suatu akun media sosial berbentuk @[Token].

    a. Token maksimal terdiri dari 15 karakter

    b. Token pada akun media sosial hanya bisa terdiri dari karakter-karakter ***alphanumeric*** (A-z, 0-9), tanda ".", atau tanda "_"

    c. Beberapa contoh akun media sosial **valid**:
        i. @iwuvu3000
        ii. @the_king.ar
        iii. @1panic__

    d. Beberapa contoh akun media sosial **tidak valid**:
        i. @info-retrieve
        ii. @lulus2021.
        iii. @fake_soc_med_account

    e. Khusus untuk akun Twitter, dapat dipastikan bahwa **tidak** terdapat karakter "." pada token

Soal:

1. **[10]** Tuliskan **seluruh akun Email** yang terdapat pada korpus tersebut dan **jumlah** setiap akun Email tersebut **muncul** dalam korpus!
2. **[9]** Tuliskan **seluruh akun media sosial** yang terdapat pada korpus tersebut dan **jumlah** setiap akun media sosial tersebut **muncul** dalam korpus!
3. **[6]** Berdasarkan jawaban dari nomor 2, tentukan akun yang **dapat dipastikan** merupakan akun Instagram (harus dengan regex untuk menentukannya)!

Catatan:

Jawaban Anda untuk bagian ini diharuskan berbentuk baris-baris yang berisi "[NAMA_AKUN] - [JUMLAH_MUNCUL]" untuk setiap akun yang ditemukan. Contoh:

- Untuk akun Email:

  perolehan_informasi@cs.ui.ac.id - 2
  genshin-anniversary@mihoyo.com - 6

- Untuk akun media sosial:

  @naed_aar - 4
  @dota2 - 3

C - Tokenization (15 Poin)

1. **[2]** Transformasi semua karakter dalam "**Korpus_Ringkas.json**" ke format ***lowercase***!
2. **[3] Hilangkan** seluruh karakter berupa angka, whitespace berlebih, dan tanda baca!
3. **[6]** Lakukan **tokenisasi** pada setiap baris data. Tampilkan hasil tokenisasi berupa ***list of tokens*** dari 5 data pertama!
4. **[4]** Untuk setiap baris data tentukan token yang dapat dikategorikan sebagai ***stopwords***! Hilangkan kata-kata tersebut dari *list of tokens*!

D - Stemming dan Lemmatization (35 Poin)

1. **[11]** Menggunakan *list of tokens* yang diperoleh dari bagian C, lakukan ***stemming*** dengan menggunakan ***library* Sastrawi** untuk masing-masing token pada setiap baris data. Kemudian, buatlah pemetaan antara token sebelum di-*stem* dengan token setelah di-*stem* menggunakan Sastrawi **(*before stem & after stem*)**. Cukup tampilkan pemetaan pada token dari 5 data pertama saja.

2. **[8]** Menggunakan *list of tokens* yang diperoleh dari bagian C, lakukan ***lemmatization*** untuk masing-masing token pada setiap baris data menggunakan ***library* Stanza**. Kemudian, tampilkan hasil *list of tokens* dari 5 data pertama!

3. **[4]** Amati pemetaan yang diperoleh pada nomor 1. Apakah Anda menemukan **suatu pola** tertentu antara token sebelum dengan sesudah di-*stem*? Kemudian, apakah Anda merasa algoritma yang digunakan *library* Sastrawi dalam melakukan *stemming* sudah **sempurna**? Berikan analisis singkat Anda (minimal 3 kalimat).

4. **[4]** Misalkan terdapat suatu token yang sebenarnya memiliki konteks atau makna ganda sebelum di-*stem*. Kemudian, token tersebut kehilangan makna atau ambigu setelah dilakukan *stemming* (contoh: **mengawani** yang bisa memiliki kata dasar **awan** atau **kawan**). Berikan analisis singkat Anda (minimal 3 kalimat) terkait pengaruhnya dalam **pembentukan** sebuah sistem IR apabila dilihat dari perspektif kompleksitas perancangan dan performa yang dihasilkan.

5. **[4]** Jika Anda dihadapkan dengan korpus Bahasa Indonesia dimana komposisi korpus **didominasi** dengan ejaan tidak baku atau *slang words* (misal dinamakan korpus XYZ), apakah menurut Anda *stemming* dan/atau *lemmatization* dapat memberikan **manfaat** dalam membentuk sebuah sistem IR yang baik? Jelaskan analisis singkat Anda (minimal 3 kalimat).

   Contoh salah satu kalimat pada korpus XYZ:
   gile sis, cape banget gue tiap hari harus nugas, jadi kangen liburan lagi deh, mana hari ini ada tugas pula, duh

6. **[4]** Bahasa Indonesia memiliki konsep **homonim, homofon,** dan **homograf** yang menyebabkan adanya dinamika makna dan pengucapan pada kata. Apakah menurut Anda proses *stemming* dan *lemmatization* saja **cukup** untuk membuat sebuah sistem IR yang baik? Jelaskan analisis singkat Anda (minimal 3 kalimat).

Lampiran

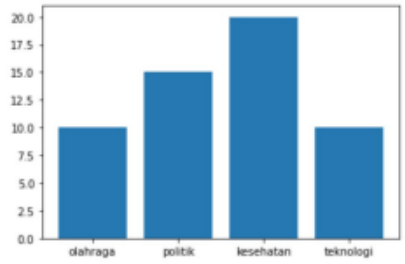
- Sastrawi: https://github.com/har07/PySastrawi
- Stanza: https://stanfordnlp.github.io/stanza/
- Spacy: https://spacy.io/usage

- Contoh menampilkan histogram dengan *library* matplotlib

```
x = ['olahraga', 'politik', 'kesehatan', 'teknologi']
y = [10, 15, 20, 10]

import matplotlib
matplotlib.pyplot.bar(x, y)
```

```
<BarContainer object of 4 artists>
```



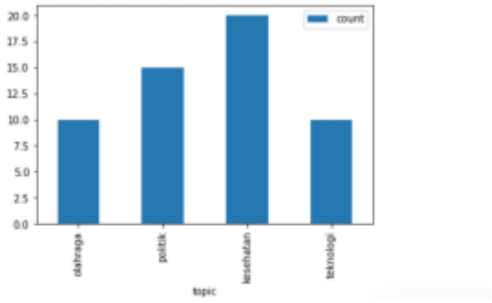
- Contoh menampilkan histogram dari data frame Pandas

```
import pandas as pd

df = pd.DataFrame([
    {'topic': 'olahraga', 'count': 10},
    {'topic': 'politik', 'count': 15},
    {'topic': 'kesehatan', 'count': 20},
    {'topic': 'teknologi', 'count': 10},
])

df.plot.bar(x='topic', y='count')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fa46f49ee48>
```



- Contoh menampilkan *word cloud* dengan library [wordcloud]

```
from wordcloud import WordCloud
import matplotlib.pyplot as plt

text = "Indonesia disebut juga dengan Republik Indonesia (RI) atau Negara Kesatuan Republik
wordcloud = WordCloud(background_color ='white').generate(text)
plt.figure(figsize=(6, 3), dpi= 100, facecolor='w', edgecolor='k') # adjust figure size
plt.imshow(wordcloud)
plt.axis("off")
```

```
(-0.5, 399.5, 199.5, -0.5)
```