



MEMBUAT API DATA CLEANSING

TAJMAHAL GHAZA
ANTONI

challange gold binar

WORKFLOW

Python programming di
jupyter notebook

1

Membuat database
dengan sqlite

2

Membuat API via
sublime_text

3

Melakukan EDA

4

LIBRARY YANG DIGUNAKAN

flask

pandas

regex

wordcloud

matplotlib

seaborn

request

Any number from 0-9
for a timer

LOGO

[Back to Navigation Page](#)

LOGO MARK



WORD MARK

HELICOPRION

Start inspired with thousands of templates, collaborate with ease, and engage your audience with a memorable Canva Presentation.

LOGO VARIATION

RULES OF APPLICATION

Do's and Dont's



Versions of our logo can be used to match a variety of materials and applications. Make sure to apply them appropriately.

LATAR BELAKANG

Dalam era digital dan media sosial, Twitter telah menjadi salah satu platform utama bagi pengguna untuk berbagi pendapat, berita, dan informasi dalam bentuk singkat yang dikenal sebagai "tweet". Jumlah tweet yang dihasilkan setiap hari sangat besar, mencapai jutaan atau bahkan miliaran tweet. Data yang terkandung dalam tweet tersebut memiliki potensi besar untuk dianalisis guna memperoleh informasi berharga, mengidentifikasi tren, mendapatkan wawasan pasar, serta melacak sentimen pengguna terhadap topik tertentu.

Namun, sebelum data tweet dapat diolah dan dianalisis, penting untuk melakukan proses data cleansing atau pembersihan data. Hal ini diperlukan karena data tweet sering kali tidak terstruktur, mengandung kekacauan, dan memiliki berbagai masalah seperti:

1. Duplikasi: Terkadang tweet dapat diunggah secara berulang kali oleh pengguna atau oleh beberapa akun palsu. Duplikasi ini dapat mempengaruhi akurasi dan validitas hasil analisis.
2. Kesalahan Teks: Pengguna Twitter seringkali menggunakan bahasa yang tidak formal, membuat kesalahan pengetikan, menggunakan singkatan, slang, atau frasa yang tidak baku. Hal ini menyebabkan kesulitan dalam analisis teks dan pengambilan informasi yang akurat.
3. Bahasa dan Emotikon: Twitter digunakan secara luas oleh pengguna di berbagai negara dengan berbagai bahasa. Selain itu, pengguna juga sering menggunakan emotikon dan emoji untuk mengekspresikan emosi. Oleh karena itu, pemrosesan bahasa dan pemahaman makna di dalam konteks yang tepat menjadi penting.
4. Noise dan Irrelevansi: Banyak tweet yang dihasilkan di Twitter tidak relevan dengan tujuan analisis yang dilakukan. Misalnya, tweet yang mengandung spam, iklan, atau konten yang tidak relevan dengan topik yang sedang dipelajari.
5. Format dan Struktur yang tidak konsisten: Tweet sering kali memiliki format yang tidak konsisten, misalnya, beberapa tweet dapat mengandung tautan, gambar, video, atau mencantumkan pengguna lain dengan menggunakan tanda @. Struktur ini perlu disesuaikan agar data dapat dianalisis dengan baik.

PEMBUATAN FUNGSI

VIA JUPYTERNOTEBOOK

```
def processing_word(input_text):
    new_text = [] # set up new list
    new_new_text = [] # set up new new list
    text = input_text.split(" ") # split input_text menjadi list of words
    for word in text: # untuk setiap word in 'text'
        if word in abusive['ABUSIVE'].tolist(): # check word di dalam list_of_abusive_words
            continue # jika ada, skip
        else:
            new_text.append(word) # jika tidak ada, masukkan ke dalam list new_text

    for word in new_text:
        new_word = new_kamus_alay.get(word, word) # check ke new_kamus_alay, apakah word ada di dictionarinya. kalau ga ada, return word
        new_new_text.append(new_word)

    text = " ".join(new_new_text)
    return text

def processing_text(input_text):
    text = re.sub(r'\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}\b', 'EMAIL', input_text) #ganti email ke kata 'EMAIL'
    text = text.lower() # jadikan lowercase semua
    text = re.sub(r'[^w\s]', '', text) # hapus semua punctuation (tanda baca)
    text = text.replace(" 62", " 0")
    text = re.sub(r"\b\d{4}\s?\d{4}\s?\d{4}\b", "NOMOR_TELEPON", text) #ganti nomor telepon ke kata 'NOMOR_TELEPON'
    text = text.replace("USER", "")
    text = text.strip()

    text = processing_word(text)
    return text
```

MEMBUAT DATABASE

VIA sqlite3

```
n [28]: import sqlite3

n [29]: conn = sqlite3.connect("C:/Users/ASUS/DSC_binar/Challage Gold - Baseline/tmp.db")

n [30]: conn.execute("""CREATE TABLE IF NOT EXISTS tweet_cleaning (id INTEGER PRIMARY KEY AUTOINCREMENT, previous_text char(1000),
conn.commit()

n [31]: conn
out[31]: <sqlite3.Connection at 0x18b0411a240>

n [32]: cursor=conn.cursor()

n [34]: for value_1, value_2 in df[['Tweet', 'cleaned_new_tweet']].values:
    value_1 = value_1.encode('utf-8')
    value_2 = value_2.encode('utf-8')
    query = f"INSERT INTO tweet_cleaning (previous_text,cleaned_text) VALUES (?, ?);"
    cursors = conn.execute(query, (value_1, value_2))
    conn.commit()
```

Homepage

1 -> Input Text

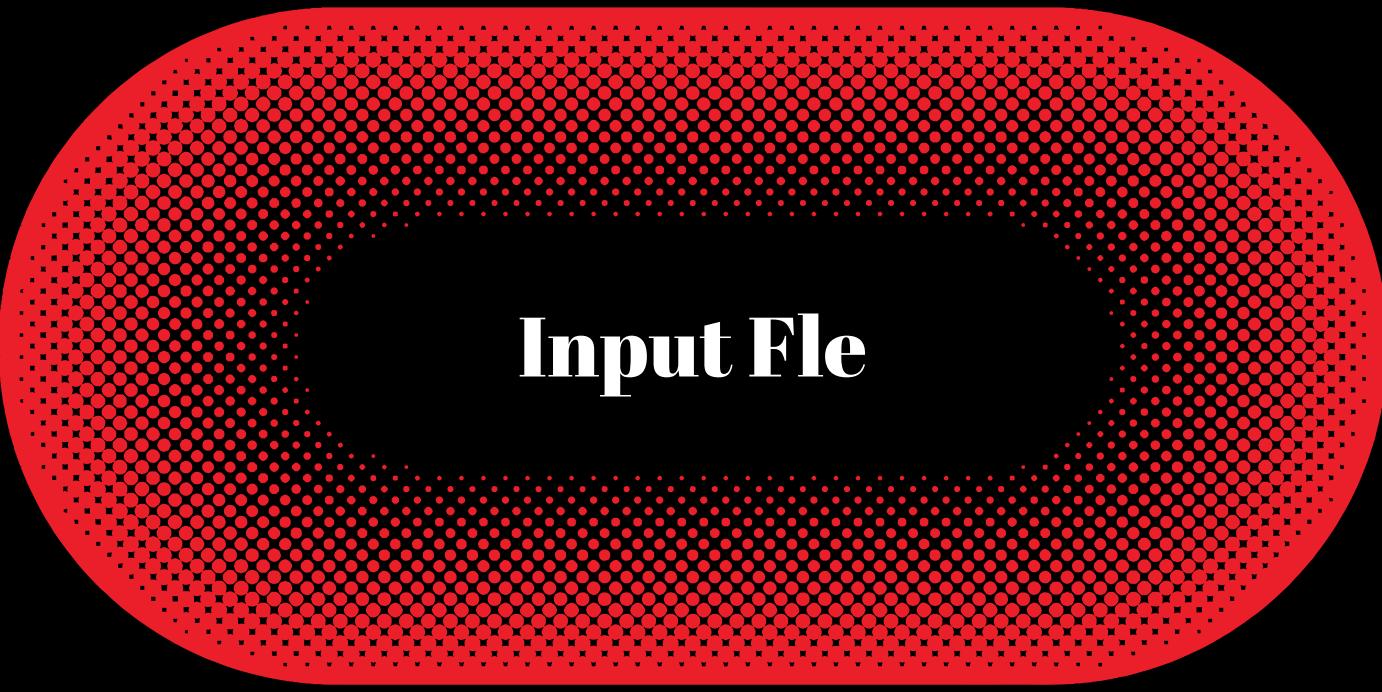
2 -> Input File

3 -> Read Database

Submit

Input Text

sukkan Text Yang Mau Di Cleansing:



Input File

Masukkan File Yang Mau Di Cleansing:

[Choose File](#)

No file chosen

[Submit](#)

Read Database

Masukkan Index Yang Mau Ditampilkan:

Masukkan Keywords Yang Mau Ditampilkan:

Masukkan Menu:

1 -> [Tampilkan Tweets](#)

2 -> [Tampilkan Attribut_Tweets](#)

Read Database

Masukkan Index Yang Mau Ditampilkan:

Masukkan Keywords Yang Mau Ditampilkan:

Masukkan Menu:

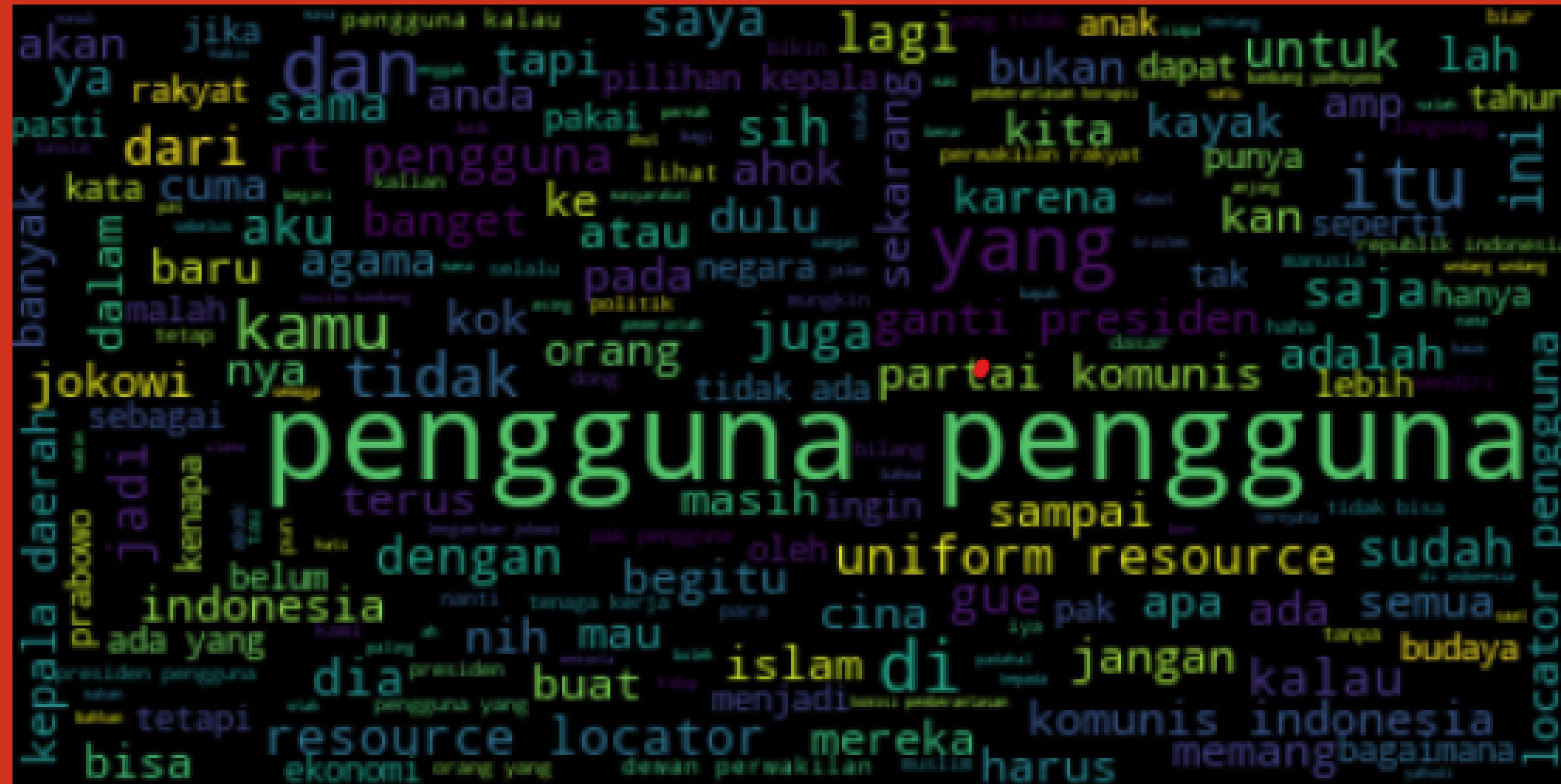
1 -> [Tampilkan Tweets](#)

2 -> [Tampilkan Attribut_Tweets](#)



WORDCLOUD

Sebelum cleansing

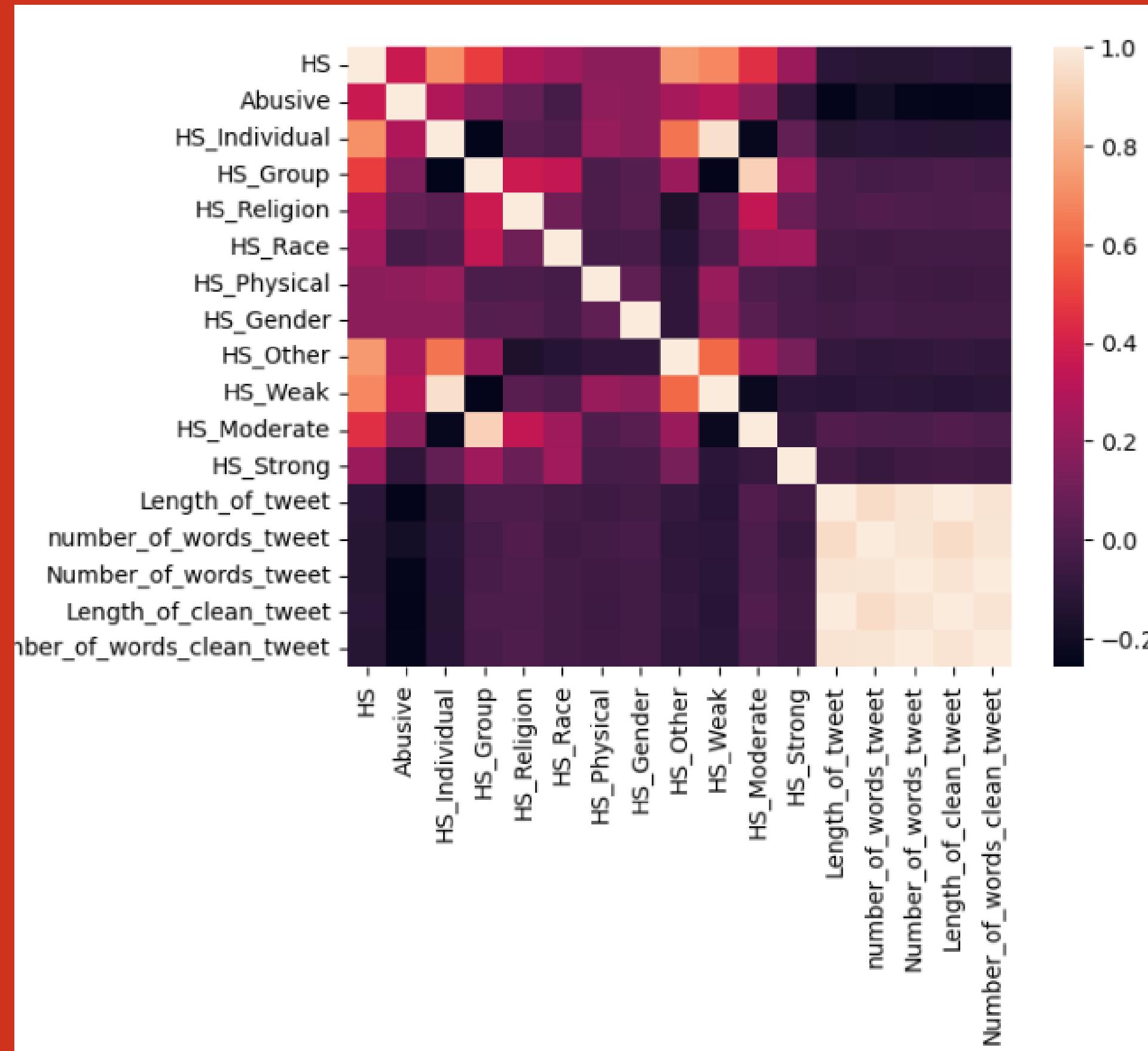


WORDCLOUD

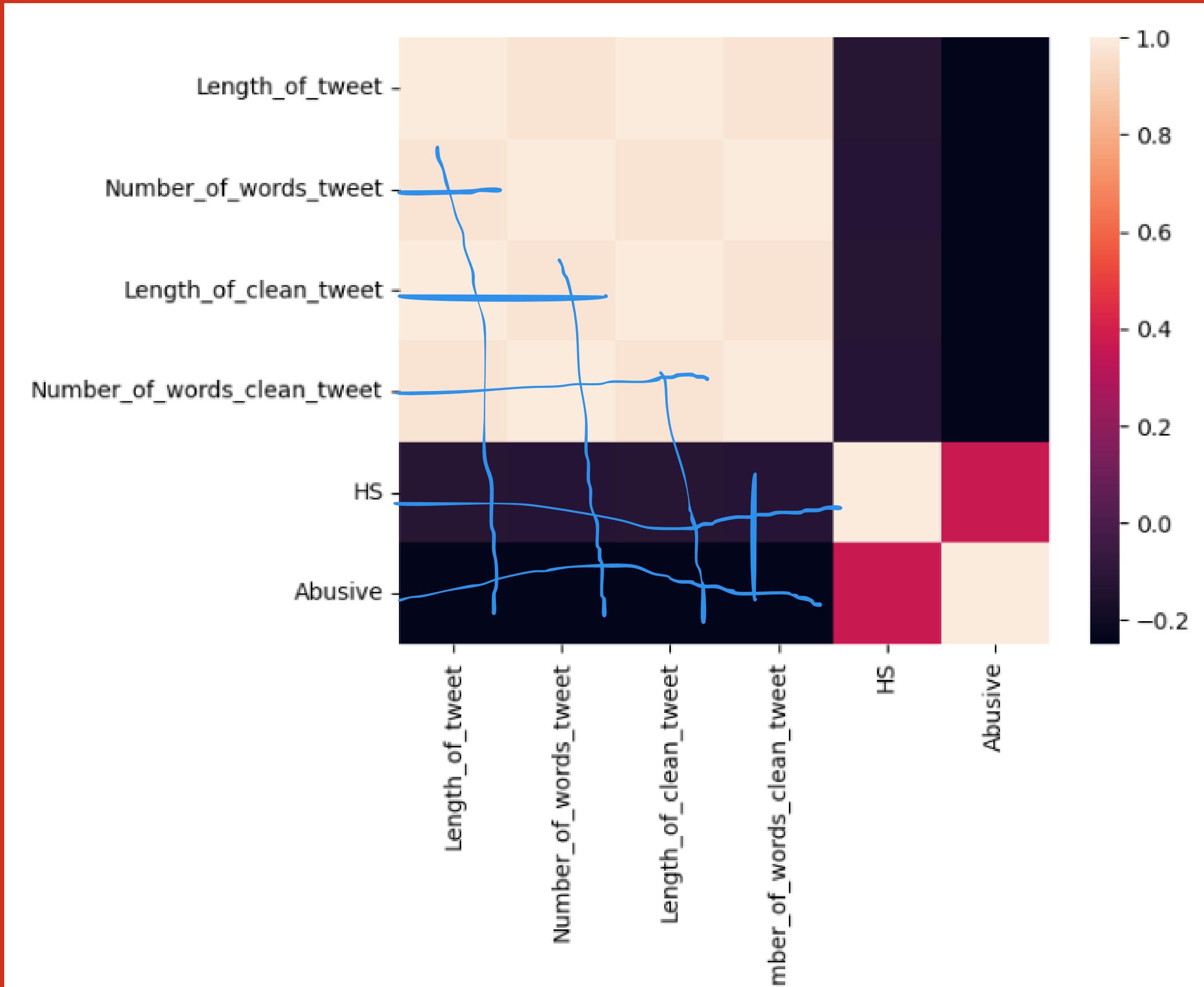
Setelah cleansing

EDA

Mencari korelasi dengan heatmap



Korelasi tinggi yang memungkinkan



Daerah arsir yang di
kira memiliki
korelasi relatif tinggi

	Length_of_tweet	Number_of_words_tweet	Length_of_clean_tweet	Number_of_words_clean_tweet	HS	Abusive
Length_of_tweet	1.000000	0.973410	1.000000	0.973410	-0.118947	-0.250159
Number_of_words_tweet	0.973410	1.000000	0.973410	1.000000	-0.126906	-0.245276
Length_of_clean_tweet	1.000000	0.973410	1.000000	0.973410	-0.118947	-0.250159
Number_of_words_clean_tweet	0.973410	1.000000	0.973410	1.000000	-0.126906	-0.245276
HS	-0.118947	-0.126906	-0.118947	-0.126906	1.000000	0.365292
Abusive	-0.250159	-0.245276	-0.250159	-0.245276	0.365292	1.000000

Interval Koefisien tingkat Hubungan

0	0,0-0,19	Sangat rendah
1	0,2-0,39	Rendah
2	0,4-0,59	Sedang
3	0,6-0,79	Kuat
4	0,8-1	Sangat kuat

Perhatikan nilai korelasi

Bandingkan hasil

KESIMPULAN DAN SARAN

Ditemukan korelasi sangat kuat adalah setiap kombinasi 'Tweet','Length_of_tweet','Number_of_words_tweet','cleaned_new_tweet','Length_of_clean_tweet','Number_of_words_clean_tweet'

Ditemukan korelasi rendah Abusive dengan 'HS','Tweet','Length_of_tweet','Number_of_words_tweet','cleaned_new_tweet','Length_of_clean_tweet','Number_of_words_clean_tweet'

Ditemukan korelasi sangat rendah HS dengan 'Tweet','Length_of_tweet','Number_of_words_tweet','cleaned_new_tweet','Length_of_clean_tweet','Number_of_words_clean_tweet'

informasi menarik terdapat kecenderungan positif rendah, antara Abusive dan HS, artinya semakin tinggi penggunaan abusive maka terdapat kecenderungan rendah semakin tinggi pula penggunaan HS

Ditemukan kata yang paling banyak digunakan adalah USER atau Pengguna