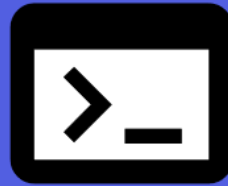


Data Science Challenge

Gold Level

Oleh:

Khairina Yasmine



**MEMBUAT API UNTUK
CLEANSING DATA TEKS
& LAPORAN ANALISIS
DATA**

Latar Belakang

Hingga tahun 2023 pengguna social media di Indonesia tercatat mencapai

167 JUTA orang

Terdapat kecenderungan pengguna social media untuk aktif berkomentar.

Dalam komentar-komentar tersebut tidak jarang dijumpai penggunaan kata kasar.

Berdasarkan hal tersebut, penelitian ini bertujuan untuk menganalisis:

- 1** Kecenderungan jumlah kata
- 2** Kecenderungan panjang karakter
- 3** Kata yang sering muncul dalam komentar
- 4** Kata abusive yang sering muncul dalam komentar

Menggunakan server API yang dibuat dengan Flask & Swagger UI

1. Mengimport library flask, pandas, dan fungsi untuk membersihkan data & menyimpan data ke dalam database

```
4 from flask import Flask, jsonify, request
5 import pandas as pd
6 from time import perf_counter
7 from flasgger import Swagger, swag_from, LazyString, LazyJSONEncoder
8 from db import (
9     create_connection, insert_dictionary_to_db,
10     insert_result_to_db, show_cleansing_result,
11     insert_upload_result_to_db, insert_abusive_occurrence_to_db
12 )
13 from cleansing_function import (
14     text_cleansing, cleansing_files,
15     count_abusive, abusive_occurrence
16 )
```


2. Menginisialisasi Flask application

```
27 # Initialize flask application
28 app = Flask(__name__)
29 # Assign LazyJSONEncoder to app.json_encoder for swagger UI
30 app.json_encoder = LazyJSONEncoder
31 # Create swagger config & swagger template
32 swagger_template = {
33     "info": {
34         "title": LazyString(lambda: "Text Cleansing API"),
35         "version": LazyString(lambda: "1.0.0"),
36         "description": LazyString(lambda: "Dokumentasi API untuk membersihkan text"),
37     },
38     "host": LazyString(lambda: request.host)
39 }
40 swagger_config = {
```

3. Membersihkan teks & menampilkan hasil cleansing

```
66 # Show cleansing result
67 @swag_from('docs/show_cleansing_result.yml', methods=['GET'])
68 @app.route('/show_cleansing_result', methods=['GET'])
69 def show_cleansing_result_api():
70     db_connection = create_connection()
71     cleansing_result = show_cleansing_result(db_connection)
72     return jsonify(cleansing_result)
73
74 # Cleansing text using form
75 @swag_from('docs/cleansing_form.yml', methods=['POST'])
76 @app.route('/cleansing_form', methods=['POST'])
77 def cleansing_form():
78     # Get text from input user
79     raw_text = request.form["raw_text"]
80     # Count abusive
81     jumlah_kata_abusive = count_abusive(raw_text)
82     # Abusive word occurrence
83     data_kemunculan_kata_abusive = abusive_occurrence
84     # Cleansing text
85     start = perf_counter()
86     clean_text = text_cleansing(raw_text)
87     end = perf_counter()
88     time_elapse = end - start
89     print(f"Processing time: {time_elapse} second")
90     result_response = {"raw_text": raw_text, "clean_text": clean_text}
91     # Insert result to database
92     db_connection = create_connection()
93     insert_result_to_db(db_connection, raw_text, clean_text, jumlah_kata_abusive)
94     insert_abusive_occurrence_to_db(db_connection, data_kemunculan_kata_abusive)
95     return jsonify(result_response)
96
97 # Cleansing text using csv upload
98 @swag_from('docs/cleansing_upload.yml', methods=['POST'])
99 @app.route('/cleansing_upload', methods=['POST'])
100 def cleansing_upload():
```

Tampilan Text Cleansing API

000/docs/#/  Swagger Supported by SMARTBEAR Explore

Text Cleansing API 1.0.0

[Base URL: 127.0.0.1:5000]
[/docs.json](#)

Dokumentasi API untuk membersihkan text

Menampilkan cleansing result

Menerima input text

get API Information

GET /

Show Cleansing Result from Database

GET /show_cleansing_result

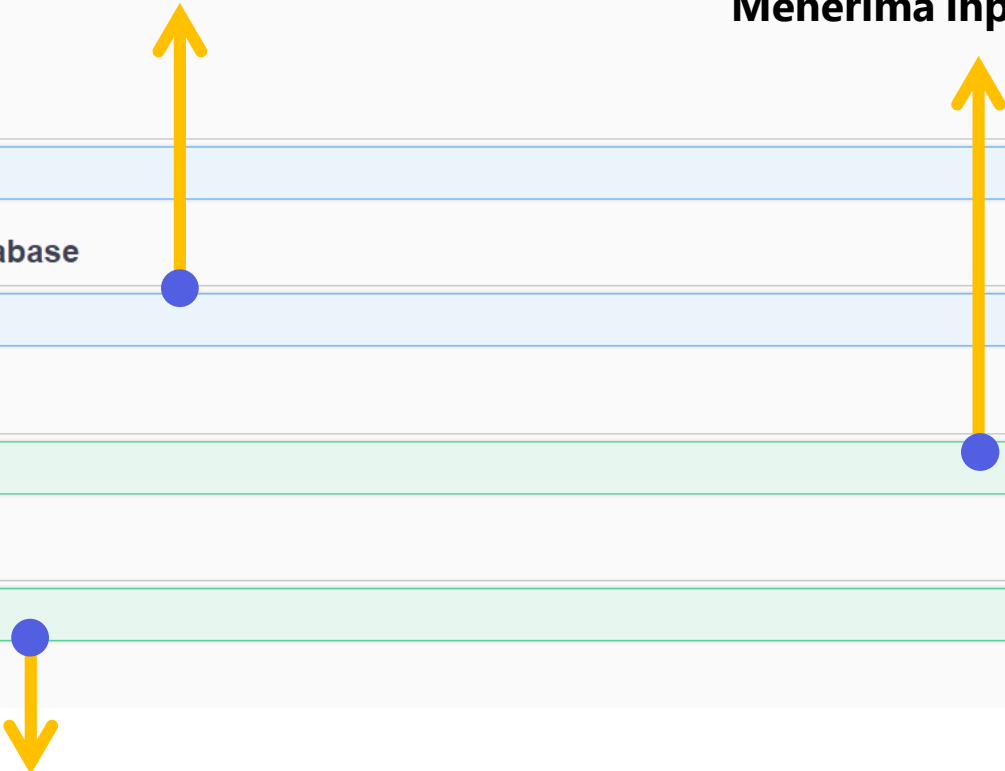
cleansing_form

POST /cleansing_form

cleansing_upload

POST /cleansing_upload

[Powered by [Flasgger](#) 0.9.5]



Menerima input file

Cleansing data dengan python

1. Import regex dan pandas

```
5 import re
6 import pandas as pd
```

2. Import csv untuk membersihkan kata alay dan kata abusive

```
8 # import csv
9 # kamus abusive
10 abusive_data = pd.read_csv("csv_data/abusive.csv")
11 # kamus alay
12 kamus_alay_data = pd.read_csv("csv_data/alay.csv", encoding="latin-1")
13 kamus_alay_data.columns = ['alay', 'baku']
14 kamus_alay_dict = dict(zip(kamus_alay_data['alay'], kamus_alay_data['baku']))
15
16 # fungsi untuk membersihkan kata alay
17 def alay_cleansing(text):
18     for key, value in kamus_alay_dict.items():
19         if key == text:
20             text = text.replace(key, value)
21     return text
22
23 # fungsi untuk mencensor kata abusive
24 def abusive_cleansing(text):
25     for word in abusive_data['ABUSIVE']:
26         if word == text:
27             text = text.replace(word, word[0] + '*' * (len(word)-1))
28     return text
```

3. Cleansing function untuk membersihkan data dari input teks dan input file

```
44 def text_cleansing(text):
45     # lowercase
46     clean_text = str(text).lower()
47     # membersihkan URL
48     clean_text = re.sub(r'(http\S+|www\S+)', '', clean_text).strip()
49     # bersihkan tanda baca (selain huruf dan angka)
50     clean_text = re.sub(r'^[a-zA-Z0-9\s]', ' ', clean_text)
51     # membersihkan username
52     clean_text = re.sub('user', ' ', clean_text)
53     # mensubtitusikan kata alay dengan kata baku
54     clean_text = ' '.join([alay_cleansing(j) for j in clean_text.split()])
55     # mencensor kata abusive
56     clean_text = ' '.join([abusive_cleansing(i) for i in clean_text.split()])
57     clean_text = re.sub('uniform resource locator', ' ', clean_text).strip()
58     return clean_text
59
60 def cleansing_files(file_upload):
61     # Ambil hanya kolom pertama saja
62     df_upload = pd.DataFrame(file_upload.iloc[:,0])
63
64     # Rename kolom menjadi "raw_text"
65     df_upload.columns = ["raw_text"]
66
67     # Bersihkan text menggunakan fungsi text_cleansing
68     # Simpan di kolom "clean_text"
69     df_upload["clean_text"] = df_upload["raw_text"].apply(text_cleansing)
70     #df_upload["jumlah_kata_alay"] = df_upload["raw_text"].apply(count_alay)
71     df_upload["jumlah_kata_abusive"] = df_upload["raw_text"].apply(count_abusive)
72     print("Cleansing text success!")
73     return df_upload
```

Code Function Cleansing

Penyimpanan data dalam SQLite menggunakan modul SQLite3

Code untuk menginsert hasil cleansing ke database

```
9 import pandas as pd
10 import sqlite3
11
12 def create_connection():
13     conn = sqlite3.connect('test2.db')
14     return conn
15
16 def insert_dictionary_to_db(conn):
17     abusive_csv_file = "csv_data/abusive.csv"
18     alay_csv_file = "csv_data/alay.csv"
19
20     # Read csv file to dataframe
21     print("Reading csv file to dataframe...")
22     df_abusive = pd.read_csv(abusive_csv_file)
23     df_alay = pd.read_csv(alay_csv_file, encoding="latin-1")
24
25     # Standardize column name
26     df_abusive.columns = ['word']
27     df_alay.columns = ['alay_word', 'formal_word']
28
29     # Insert dataframe to database
30     print("Inserting dataframe to database...")
31     df_abusive.to_sql('abusive', conn, if_exists='replace', index=False)
32     df_alay.to_sql('alay', conn, if_exists='replace', index=False)
33     print("Inserting dataframe to database success!")
34
35 def insert_result_to_db(conn, raw_text, clean_text, jumlah_kata_abusive):
36     # Insert result to database
37     print("Inserting result to database...")
38     df = pd.DataFrame({'raw_text': [raw_text], 'clean_text': [clean_text], 'jumlah_kata_abusive': [jumlah_kata_abusive]})
39     df.to_sql('cleansing_result', conn, if_exists='append', index=False)
40     print("Inserting result to database success!")
41
42 def insert_upload_result_to_db(conn, clean_df):
43     # Insert result to database
44     print("Inserting result to database...")
45     clean_df.to_sql('cleansing_result', conn, if_exists='append', index=False)
46     print("Inserting result to database success!")
47
48 def insert_abusive_occurrence_to_db(conn, abusive_occurrence):
49     # Insert result to database
50     print("Inserting abusive occurrence data to database...")
51     df = pd.DataFrame({'abusive_occurrence': abusive_occurrence})
52     df.to_sql('abusive_occurrence', conn, if_exists='append', index=False)
53     print("Inserting abusive occurrence data to database success!")
54
```

Menampilkan database menggunakan aplikasi DBeaver

cleansing_result			Enter a SQL expression to filter results (use Ctrl+Space)
Grid	ABC raw_text	ABC clean_text	
1	- disaat semua cowok berusaha melacak perhatian gue.	di saat semua cowok berusaha melacak perhatian gue k	
2	RT USER: USER siapa yang telat ngasih tau elu?edan sara	rt siapa yang telat memberi tau kamu e**** s**** gue ber	
3	41. Kadang aku berfikir, kenapa aku tetap percaya pada T	41 kadang aku berpikir kenapa aku tetap percaya pada t	
4	USER USER AKU ITU AKU\n\nKU TAU MATAMU SIPIT TAF	aku itu aku dan ku tau matamu s**** tapi dilihat dari ma	
5	USER USER Kaum cebong kapir udah keliatan dongokny:	kaum c**** k**** sudah kelihatan dongoknya dari awal	
6	USER Ya bani taplak dkk \xf0\x9f\x98\x84\xf0\x9f\x98\x8	ya b**** t**** dan kawan kawan xfo x9f x98 x84 xfo x9f x	
7	deklarasi pilkada 2018 aman dan anti hoax warga dukuh	deklarasi pilihan kepala daerah 2018 aman dan anti hoal	
8	Gue baru aja kelar re-watch Aldnoah Zero!!! paling kamp	gue baru saja selesai re watch aldnoah zero paling k****	
9	Nah admin belanja satu lagi port terbaik nak makan Ais l	nah admin belanja satu lagi port terbaik nak makan ais l	
10	USER Enak lg klo smbil ngewe'	enak lagi kalau sambil n****	
11	Setidaknya gw punya jari tengah buat lu, sebelum gw uk	setidaknya gue punya jari tengah buat kamu sebelum g	
12	USER USER USER USER BANCIL KALENG MALU GA BISA JA	b**** kaleng malu tidak bisa jawab pertanyaan kami dar	
13	Kalo belajar ekonomi mestinya jago memprivatisasi hati	kalau belajar ekonomi mestinya jago memprivatisasi ha	
14	Aktor huruhara 98 Prabowo S ingin lengserkan pemerint	aktor huru hara 98 prabowo si ingin lengserkan pemerin	
15	USER Bu guru enakan jadi jablay atau guru esde sih.\nKa	bu guru enakan jadi j**** atau guru sekolah dasar sih ki	
16	USER USER USER USER USER USER Lawan bicara gw gak i	lawan bicara gue tidak intelek kayak kamu yang otak tid	
17	Belakangan ini kok pikiran ampas banget ya'	belakangan ini kok pikiran a**** banget ya	
18	Ari sarua beki mah repeh monyet\xfb\x9f\x98\x86\xfb\xfb	ari sama beki adalah rapi m**** xfo x9f x98 x86 xfo x9f >	
19	Jadi cowo itu harus Gantle kalo ga Gantle itu namanya B	jadi cowok itu harus gantle kalau tidak gantle itu namar	
20	USER Slga mnr bom \xf0\x9f\x98\x82'	alga mnr bom xfo x9f x98 x82	
21	Asw ya tapi gua jarang ngambek, tacut wkwwkwkwkw gu	a**** ya tapi gue jarang mengambek takut wkwk gue k	
22	USER kalo kamu noob pasti peluang disakitin nya lebih g	kalau kamu n**** pasti peluang disakiti nya lebih gede sil	
23	USER Joko Widodo dinilai sebagai presiden terlemah dal	joko widodo dinilai sebagai presiden terlemah dalam se	
24	PELAJAR SMA KEC BILAH HILIR DEKLARASI ANTI HOAX	pelajar sama kecamatan bilah hilir deklarasi anti hoaks a	
25	Bandara Udara Internasional Kertajati dibangun oleh Gu	bandara udara internasional kertajati dibangun oleh gu	
26	Siapapun gubernur dan presidennya, rakyatnya, ya kitaA	siapapun gubernur dan presidennya rakyatnya ya kita ju	
27	Ini si USER kerjaannya delay mulu! Setan!	ini sih kerjaannya delay mulu s****	
28	menurutku pintu sorga ada yaitu pintu sorga yang asli di	menurutku pintu sorga ada yaitu pintu sorga yang asli c	
29	RT USER USER USER PKI hanya muncul jika jelang pemilih	rt partai k**** indonesia hanya muncul jika jelang per	
30	USER USER Itu mah sdh nenek-nenek sy heran sama cebu	itu adalah sudah nenek nenek saya heran sama c**** bi	
31	USER USER USER USER USER USER USER USER Islam	islam nusantara produk jil dipasarkan dengan gencar ol	
32	Dari habis sahur sampe jam 10. Sibayik udah nete 4x. Skg	dari habis sahur sampai jam 10 si bayi sudah n**** empat	
33	USER Gak kak emak mah gak demen ama yang sipit wkww	tidak kak emak adalah tidak suka sama yang s**** wkww	
34	RT USER: Kelakuan homok jaman now, ngentot aja samb	rt kelakuan h**** jaman now n**** saja sambil live di bl	
35	USER Ga kak, gua bukan orang jawa maaf yak :(; gua ori	tidak kak gue bukan orang jawa maaf ya gue orang pale	

Metode Statistika

Metode statistika yang digunakan adalah metode **Descriptive Analytics**.

Metode ini digunakan untuk mencari tahu kondisi data, dan menemukan tren serta pola data.

Analisis & visualisasi data dilakukan menggunakan **jupyter lab**, dengan mengimport library **pandas, matplotlib, & seaborn**

Exploratory Data Analysis

Univariate Analysis

Analisis yang melibatkan 1 variable, digunakan untuk melihat kondisi data masing-masing variable seperti jumlah kata, jumlah karakter, serta jumlah kata abusive.

Bivariate Analysis

Analisis yang melibatkan 2 variable, digunakan untuk mencari korelasi antar variable.

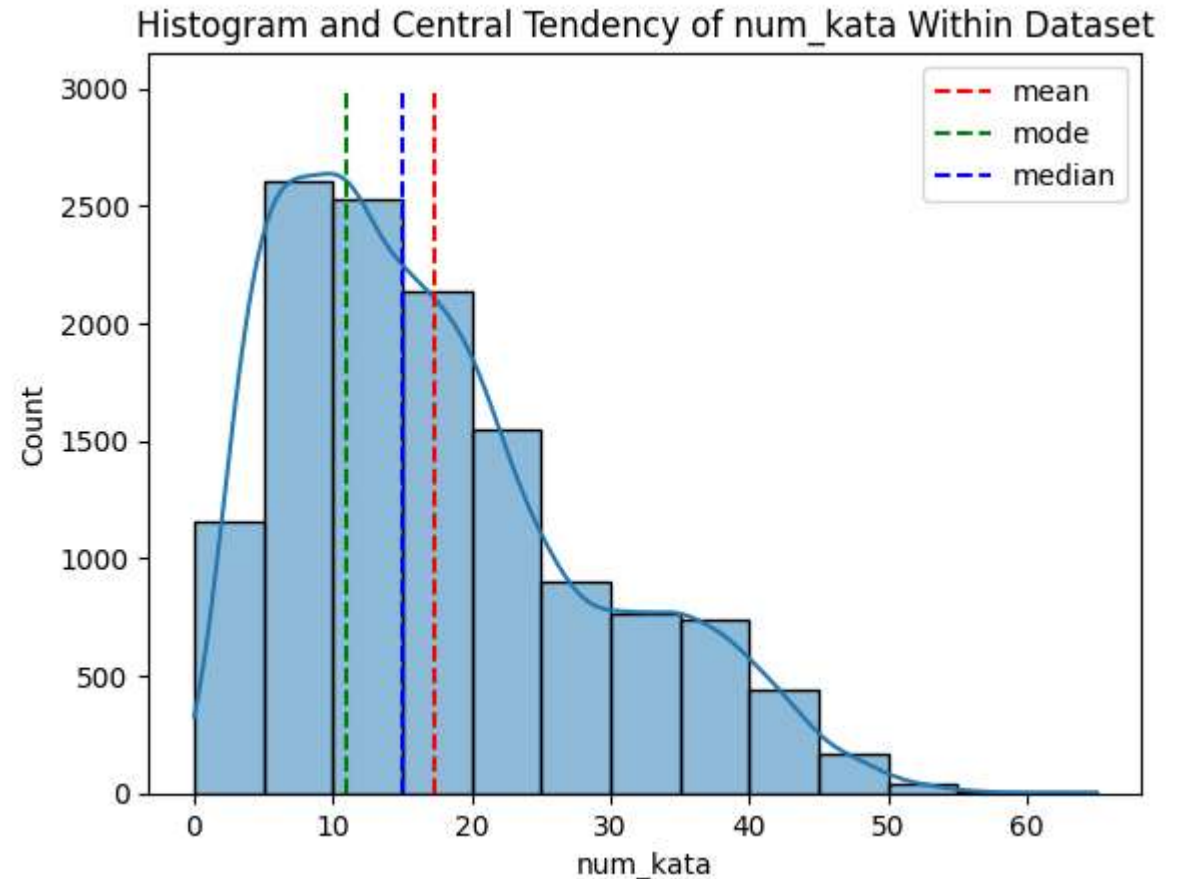
Kecenderungan jumlah kata

Dari hasil analisis menggunakan pandas, didapatkan:

- Total jumlah kata dari dataset sebanyak: **225,955** kata
- Rata-rata jumlah kata dalam teks di dataset: **17.350** kata

	Common_words	count
0	yang	4997
1	dan	3549
2	tidak	3197
3	di	3166
4	itu	2136
5	kamu	1812

5 kata yang paling sering muncul adalah **yang, dan, tidak, di, itu, & kamu**



Nilai **skewness** > 0 , artinya skewness bernilai positif
Modus $>$ Median $>$ Mean

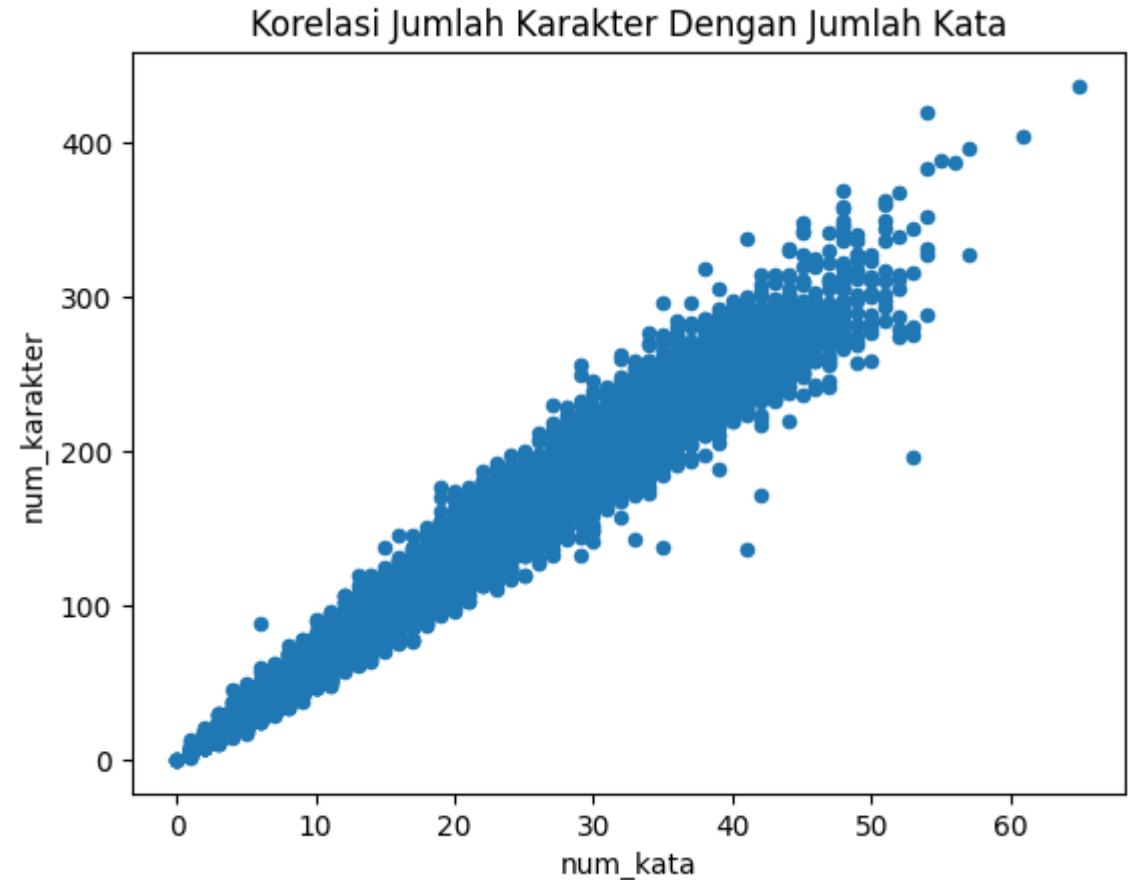
Kecenderungan jumlah karakter

Dari hasil analisis menggunakan pandas, didapatkan:

- Total jumlah karakter dari dataset sebanyak: **1,437,426** karakter
- Rata-rata jumlah karakter dalam teks di dataset: **110.376** karakter

Menggunakan **Bivariate Analysis** dicari korelasi antara jumlah karakter dan jumlah kata dengan **scatter plot**.

Didapatkan korelasi positif antara jumlah karakter dan jumlah kata.



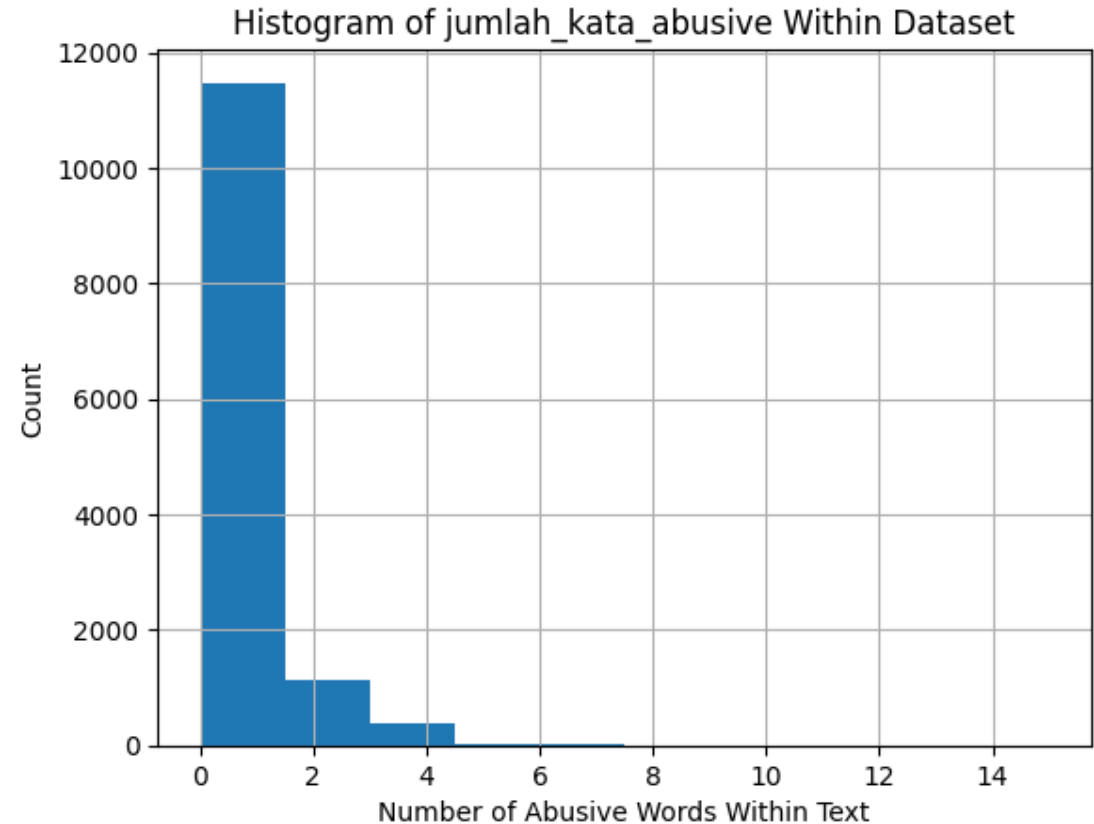
Analisis kata abusive dalam dataset

Dari hasil analisis menggunakan pandas, didapatkan:

- Total jumlah kata abusive dari dataset sebanyak: **8,769** kata
- Rata-rata jumlah kata abusive dalam teks di dataset: **0.673** kata

	Common_words	count
0	cebong	518
1	asing	427
2	komunis	357
3	rezim	357
4	kafir	302
5	antek	264

5 kata abusive yang paling sering muncul adalah **cebong, asing, komunis, rezim, kafir, & antek**



- Mayoritas jumlah kemunculan kata abusive dalam teks adalah 0 – 1 kata
- **>70%** teks memiliki jumlah kata abusive sebanyak **0**

Kesimpulan

Univariate Analysis

- Dalam Descriptive Statistic menunjukkan data yang diolah memiliki outlier dari sisi batas atas.
- Total jumlah kata pada dataset: 225,955 kata
- Rata-rata kata per teks: 17.350 kata.
- Total jumlah karakter pada dataset: 1,437,426 karakter
- Rata-rata karakter per teks: 110.376 karakter.
- Kata yang paling banyak muncul: **yang, dan, tidak, di, itu, & kamu**
- Kata yang abusive paling banyak muncul: **cebong, asing, komunis, rezim, kafir, & antek**

Bivariate Analysis

- Dalam Descriptive Statistic menunjukkan variabel total karakter dan total kata memiliki korelasi positif.