LAPORAN TUGAS CRAWLING DAN PREPROCESSING DATA TEXT MATA KULIAH TEXT MINING & NATURAL LANGUAGE PROCESSING



Tim Penyusun:

- 1. <5231811022> < Lathif Ramadhan>
- 2. <5231811029> < Andini Angel Meivita>
- 3. <5231811033> < Rama Panji Nararendra>
- 4. <5231811036> <Giffari Riyanda Pradithya>

PROGRAM STUDI SAINS DATA PROGRAM SARJANA FAKULTAS SAINS & TEKNOLOGI UNIVERSITAS TEKNOLOGI YOGYAKARTA 2025

1. Hastag yang digunakan: #ramadhan

2. Tuliskan langkah-langkah crawling data text:

a. Tuliskan/screenshotkan codingnya

Jawab:

Link file kodenya:

https://colab.research.google.com/drive/1wxDKwRGiu_pd9CDx6kNHN6I8RfUdxLr S#scrollTo=o2wMKSTWhoNW

```
# instal Node.js dan npm (Node Package Manager) di sistem
!curl -sL https://deb.nodesource.com/setup_18.x | sudo -E
bash -
!sudo apt-get install -y nodejs

# nama file dari data yang berhasil dikumpulkan
data = "ramadan_1.csv"

# kata kunci dari data yang ingin dicari
search_keyword = "Ramadan"

# limit baris pencarian
limit = 200

# jalankan proses crawling tweet dengan bantuan tweet-harvest
menggunakan Twitter API token.
!npx --yes tweet-harvest@2.6.1 -o "ramadan_1.csv" -s
"{Ramadan}" -1 {100} --token "--------"
```

b. Tuliskan/screenshotkan hasil crawlingnya

Jawab:

A B C D E G H G L M N O P R S 0.0,0.0,1899490458658623578, Tue Mar 11 15:59:50 +0000 2025,9151, Everyone is fasting for Ramadan at my work and this is my lunch https://t.co/5GVH8LHtl7,1899490458658623578, https://pbs.twimg.co 1.0,1.0,1899166900363333645, Mon Mar 10 18:34:08 +0000 2025,16663, Do not forget your mother in every prayer this Ramadan... https://t.co/KVaANWgELv,1899166900363333645, https://pbs.twimg.com 2.0,2.0,1897724527146111472, Thu Mar 06 19:02:39 +0000 2025,802, Iftar à Paris - Âu #Ramadan le mois de la fraternité et de la tolérance face à la dictature religieuse et lextrémisme Ân Ni contrail 3.0,3.0,1897638172114206796, Thu Mar 06 13:19:31 +0000 2025,53064, Non muslims think Ramadan is a hard time for us. They don't know that this is actually our favourite time of the whole year.,1897636 4.0,4.0,1.8! some cardinals celebrate ramadan. Islam is the biggest threat to humanity. https://t.co/4dUEL5Vaev,1899241255059546545,https://pbs.twimg.com/amplify_video_thumb/18992411476139827 5.0,5.0,1.8! our resolve against fascism in Pakistan stands vindicated. Insha Allah it is a matter of time that this junta of #gen_asim_munir_Butcher_of_Pakistan will fall on its own weight of crimes against factors. 6.0,6.0,1899246623932231961,Mon Mar 10 23:50:56 +0000 2025,39205,im not even feeling hungry or thirsty this Ramadan just extremely exhausted and sleepy like never before,1899246623932231961,... 9 7.0,7.0,1899628358410912042,Wed Mar 12 01:07:48 +0000 2025,25, Respecting women is the first step toward achieving true gender equality. Good Morning Ramadan Kareem https://t.co/k8br56nuky,1810 8.0,8.0,1898399386310455738,Sat Mar 08 15:44:18 +0000 2025,7855,A rare video of Afghanistan players breaking fast during and international match in a month of Ramadan last year. #Respect https://t.c 13 | 1.0,11.0,1898817155119890890,Sun Mar 09 19:24:22 +0000 2025,6130,6 † = 6 † 7 , ī , [En ‰gypte des chrétiens distribuent des repas aux musulmans pour liftar Ā lapproche de la rupture du jeûne 14 | 12.0,12.0,1898791650979430556,Sun Mar 09 17:43:02 +0000 2025,2967,Praying Tahajjud before suhur is a life hack! This Ramadan even if once Stand before Allah with all of your problems all of your pray 13.0,13.0,1898430490350350619,5at Mar 08 17:47:54 +0000 2025,16647,Masjid Al Haram on 8th night of Ramadan! https://t.co/8qvjZJivdw,1898430490350350619,https://pbs.twimg.com/media/GliUdeal 14.0,14.0,1899255118064656718, Tue Mar 11 00:24:41 +0000 2025,195, O Allah Owner of everything plz say yes to my all prayers and make me and my loved ones happy Ameen T, Happy 10th Ramadan Mi 17 15.0.15.0.1899626396952977706, Wed Mar 12 01:00:01 +0000 2025, 161, White man is shamed by Muslims to stop eating during Ramadan https://t.co/n8vvXxn251, 1899626396952977706, https://pbs.twimg 20 18.0,18.0,18.97966007127248999,Fri Mar 07 11:02:13 +0000 2025,7636,RamadÄn is not the same for everyone. If you have anything to spare kindly check on those in need may Allah reward you as you do. 21 19.0,19.0,1898214870677336371,,en,,177,327,4438,https://x.com/Softgirlru/status/18 22 20.0,20.0,1899153523465683211,Mon Mar 10 17:40:59 +0000 2025,6477,ὰμαὰμαθακτικό και το κα 23 21.0,21.0,1899407674225070560, Tue Mar 11 10:30:53 +0000 2025,7231, #Ramadan https://t.co/3G301760DX,1899407674225070560, https://pbs.twimg.com/media/GlwNNKZWoAAKAuJ.jpg,, qme, Kingdom 24 22.0,22.0,1898939172708081860,Mon Mar 10 03:29:14 +0000 2025,11460,If you stopped it because of Ramadan may Allah distance you from it forever if you started it because of Ramadan May Allah stre 25 23.0,23.0,1898685354950991894,Sun Mar 09 10:40:39 +0000 2025,386,6Ÿ‡-6Ÿ‡\$ Let's check in on Birmingham and how the holy month of Ramadan is going. https://t.co/lsA1cb4QQV,18986853549509918 26 24.0,24.0,1898291977256599840,Sat Mar 08 08:37:30 +0000 2025,2991,Buat fresh graduate yang belum dapet kerja pas Ramadan bisa banget nyoba remote job di beberapa situs ini https://t.co/KiJ8NSsol fix_combined_ramadan(1)

c. Berapa jumlah record yang didapatkan

Jawab: 6543 baris

3. Tuliskan langkah-langkah preprocessing data text:

Link Colab Coding Cleaning:

https://colab.research.google.com/drive/15PBqlsHWNYC5eV-kj9fXwbNN9nC9whRJ?usp=s haring

Link Colab Case Folding, Tokenizing, Filtering, Stemming, dan Simpan Data Bersih:

https://colab.research.google.com/drive/1pBv_cBklF6W0eB2FAZdkWwRCXI3_dIbM?usp=sharing#scrollTo=eW2NnAO87pD8

a. Tuliskan/screenshotkan coding cleaning

Jawab:

```
import pandas as pd
import glob
import os

path = r'/content/' # use your path
from os import listdir
from os.path import isfile, join
onlyfiles = [f for f in listdir(path) if isfile(join(path, f))]
```

```
li = []
# print(all files)S
for filename in onlyfiles:
           df = pd.read csv( r'/content/' + filename,
index col=None, header=0)
    li.append(df)
frame = pd.concat(li, axis=0, ignore index=True)
frame.to_csv("fix_combined_ramadan_real.csv", index=False)
df = pd.read csv('/content/fix combined ramadan real.csv')
df
df.drop duplicates()
df.to csv("/content/fix combined ramadan.csv", index=False)
Cleaning di preprocessing:
df.drop duplicates(subset = 'Tweet', keep = 'first', inplace
= True)
# Memfilter datanya cuma bahasanya yang English
filt = (df twit['lang'] == 'en')
df twit = df twit.loc[filt, :]
df twit.reset index(inplace=True)
#menghilangkan mention/user
def remove pattern (tweet, pattern):
 r = re.findall(pattern, tweet)
 for i in r:
    tweet = re.sub(i,'', tweet)
 return tweet
df['remove user'] = np.vectorize(remove pattern)(df['teks'],
"@[\w]*")
df['remove user']
```

b. Tuliskan/screenshotkan Case folding

Jawab:

Case folding adalah proses mengubah semua teks menjadi huruf kecil agar lebih konsisten dalam analisis.

→ Di kode ini, case folding terjadi secara otomatis saat tokenizing

Parameter preserve_case=False dalam TweetTokenizer akan secara otomatis mengubah teks menjadi huruf kecil.

Kedua kode ini walaupun fungsinya untuk tokenizing dan stemming tetapi secara tidak langsung mereka juga melakukan case folding, dimana untuk tokenizer preserve_case nya di set ke False untuk di case folding.

Untuk bagian kode yang melakukan case folding di tokenizing adalah

```
tokenizer = TweetTokenizer(preserve_case=True,
strip handles=True, reduce len=True)
```

```
tweet tokens = tokenizer.tokenize(tweet)
```

Dan untuk bagian kode yang melakukan case folding di stemming adalah

```
stem word = stemmer.stem(word)
```

c. Tuliskan/screenshotkan Tokenizing

Tokenizing adalah proses memecah teks menjadi kata-kata atau token.

Jawab:

→ Bagian kode yang melakukan tokenizing:

```
tokenizer = TweetTokenizer(preserve_case=False,
strip_handles=True, reduce_len=True)
tweet tokens = tokenizer.tokenize(tweet)
```

- tokenizer.tokenize(tweet) akan memecah teks menjadi daftar kata-kata individu (token).
- preserve_case=False akan membuat kata-kata menjadi huruf kecil (bagian dari case folding juga).

d. Tuliskan/screenshotkan Filtering

Filtering adalah proses membersihkan teks dari karakter atau kata yang tidak diperlukan, seperti angka, URL, tanda baca, stopwords, dan emotikon.

Jawab:

→ Bagian kode yang melakukan filtering:

```
def tweet_clean(tweet):
    #remove angka
    tweet = re.sub('[0-9]+', '', tweet)
    # print(f"ss")
    # remove stock market tickers Like $GE
    tweet = re.sub(r'\$\w*', '', tweet)
    # remove old style retweet text "RT"
    tweet = re.sub(r'RT: [\s]+','', tweet)
```

```
tweet = re.sub(r'https?: \/\.*[\r\n]*', '', tweet)
  #remove coma
  tweet = re.sub(r', ', '', tweet)
  # remove hashtags
  # only removing the hash # sign from the word
  tweet = re.sub(r'#','', tweet)
  #Happy Emoticons
  emoticons happy = set([
  ':-)', ':)', ';)', ':0)', ':]', '3', ':c)', ':>', '=]',
':^)', ':-D', ':D' '8-D', '8D', '-3', '-3', ':-))', ":'-)"
'x-D', 'xD', 'X-D', 'XD' '>:P', 'x-p', 'xp', 'XP', ':-p',
'p', 'p', 'b', 'b', '>:)', '<3'])
 #Sad Emoticons
 emoticons sad = set([
  'L', ':-/', '>:/', 'S', '>:', '>:[','',':-(',':[', ':-||',
':-[', ':-<', '=\\', '=/', '>:(', ':(', '>.<', 'c', ':{',
'>:\\', ';(' ])
  #all emoticons (happy + sad)
  emoticons = emoticons happy.union(emoticons sad)
  #tokenize tweets
        tokenizer = TweetTokenizer(preserve case=True,
strip handles=True, reduce len=True)
  tweet tokens = tokenizer.tokenize(tweet)
  # print(f"Word after tokenizer : {tweet tokens}")
  tweets clean = []
  for word in tweet tokens:
      if (word not in stopwords english and word not in
emoticons and word not in string.punctuation): # remove
punctuation
      # print(f"Word before stemming: {word}")
      stem word = stemmer.stem(word) #stemming word
      # print(f"Word after stem: {stem word}")
      tweets clean.append(stem word)
```

#remove hyperlinks

```
return tweets_clean

df['tweet_clean'] = df['remove_user'].apply(lambda x:

tweet_clean(x))

#remove punct

def remove_punct(text):
    text = " ".join([char for char in text if char not in

string.punctuation])
    return text

df['Tweet'] = df['tweet_clean'].apply(lambda x:

remove_punct(x))
```

e. Tuliskan/screenshotkan Stemming

Stemming adalah proses mengubah kata menjadi bentuk dasarnya.

Jawab:

→ Bagian kode yang melakukan stemming menggunakan Sastrawi:

```
stem_word = stemmer.stem(word) #stemming word
```

f. Tuliskan/screenshotkan Simpan data text bersih

Jawab:

```
df.sort_values('Tweet', inplace = True)
df.drop(df.columns[[0,1]], axis = 1, inplace = True)
df.drop_duplicates(subset = 'Tweet', keep = 'first', inplace
= True)
df.to_csv('ramadan_clean_tweet.csv', encoding='utf8', index=False)
df.head(10)
```

4. Lampirkan file RAW hasil Crawling dan file bersih hasil preprocessing

Tugas Pembobotan kata menggunakan TF-IDF

1. Lakukan pelabelan pada data text yang sudah bersih, minimal 250 record

Kami pakai library VaderSentiment di python, jadi nanti hasil sentimennya dihitung otomatis.

pip install pandas vaderSentiment

```
import pandas as pd
import numpy as np
from vaderSentiment.vaderSentiment import
SentimentIntensityAnalyzer
```

1. Load Dataset

```
df =
pd.read_csv('https://raw.githubusercontent.com/LatiefDataVisionary
/text-mining-and-natural-language-processing-college-task/refs/hea
ds/main/datasets/ramadan_clean_tweet.csv')
df.head()
```

	tweet_clean	Tweet
0	['abraj', 'al', 'bait', 'clock', 'tower', 'bea	abraj al bait clock tower beams indicating com
1	['accounts', 'recognised', 'ramadan', 'none',	accounts recognised ramadan none recognised be
2	['admin', 'post', 'peaceful', 'ramadan', 'cele	admin post peaceful ramadan celebrations east
3	['admin', 'post', 'ramadan', 'norway']	admin post ramadan norway
4	['admin', 'post', 'ramadan', 'usual', 'peacefu	admin post ramadan usual peaceful start englan

2. Inisialisasi Sentiment Analyzer

```
analyzer = SentimentIntensityAnalyzer()
```

3. Fungsi untuk Analisis Sentimen

```
def get_sentiment(text):
scores = analyzer.polarity_scores(text)

# Threshold penentuan sentimen
if scores['compound'] >= 0.01:
    return 'positive'
else:
    return 'negative'
```

4. Proses Analisis Sentimen untuk Setiap Tweet

```
df['sentiment'] = df['Tweet'].apply(get_sentiment)
```

5. Tambahkan Skor Sentimen

```
def get_sentiment_scores(text):
    return analyzer.polarity_scores(text)

df['sentiment_scores'] = df['Tweet'].apply(get_sentiment_scores)
```

6. Pisahkan Skor ke Kolom Terpisah

```
df['neg'] = df['sentiment_scores'].apply(lambda x: x['neg'])

df['neu'] = df['sentiment_scores'].apply(lambda x: x['neu'])

df['pos'] = df['sentiment_scores'].apply(lambda x: x['pos'])

df['compound'] = df['sentiment_scores'].apply(lambda x: x['compound'])

df.head()
```

	tweet_clean	Tweet	sentiment	sentiment_scores	neg	neu	pos	compound
0	['abraj', 'al', 'bait', 'clock', 'tower', 'bea	abraj al bait clock tower beams indicating com	negative	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound	0.000	1.000	0.000	0.0000
1	['accounts', 'recognised', 'ramadan', 'none',	accounts recognised ramadan none recognised be	negative	{'neg': 0.147, 'neu': 0.853, 'pos': 0.0, 'comp	0.147	0.853	0.000	-0.4767
2	['admin', 'post', 'peaceful', 'ramadan', 'cele	admin post peaceful ramadan celebrations east	positive	('neg': 0.0, 'neu': 0.714, 'pos': 0.286, 'comp	0.000	0.714	0.286	0.4939
3	['admin', 'post', 'ramadan', 'norway']	admin post ramadan norway	negative	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound	0.000	1.000	0.000	0.0000
4	['admin', 'post', 'ramadan', 'usual',	admin post ramadan usual peaceful start	positive	{'neg': 0.0, 'neu': 0.775, 'pos': 0.225, 'comp	0.000	0.775	0.225	0.4939

7. Hapus Kolom Tidak Diperlukan

```
df.drop('sentiment scores', axis=1, inplace=True)
```

8. Tampilkan Contoh Hasil

```
print(df[['Tweet', 'sentiment', 'compound']].head())
```

```
Tweet sentiment compound
0 abraj al bait clock tower beams indicating com... negative
                                                                0.0000
1 accounts recognised ramadan none recognised be... negative
                                                                -0.4767
2 admin post peaceful ramadan celebrations east ... positive
                                                                0.4939
                           admin post ramadan norway negative
                                                                0.0000
4 admin post ramadan usual peaceful start englan...
                                                                0.4939
                                                     positive
     df.to_csv('ramadan_labeled_sentiment.csv', index=False)
     df = pd.read_csv('ramadan_labeled_sentiment.csv',
usecols=['tweet clean', 'Tweet', 'sentiment'])
df.head()
```

	tweet_clean	Tweet	sentiment
0	['abraj', 'al', 'bait', 'clock', 'tower', 'bea	abraj al bait clock tower beams indicating com	negative
1	['accounts', 'recognised', 'ramadan', 'none', \dots	accounts recognised ramadan none recognised be	negative
2	['admin', 'post', 'peaceful', 'ramadan', 'cele	admin post peaceful ramadan celebrations east	positive
3	['admin', 'post', 'ramadan', 'norway']	admin post ramadan norway	negative
4	['admin', 'post', 'ramadan', 'usual', 'peacefu	admin post ramadan usual peaceful start englan	positive

2. Lakukan pembobotan data text menggunakan TF-IDF yang sudah diberi label

Dalam proyek ini, kita melakukan **analisis sentimen** terhadap tweet bertema **Ramadhan** menggunakan **Natural Language Processing (NLP)**. Fokus utama adalah memproses teks, menerapkan **TF-IDF** (**Term Frequency - Inverse Document Frequency**) untuk ekstraksi fitur, dan menganalisis kata-kata yang paling berpengaruh dalam dataset.

a. Import Library dan Data



b. Menggabungkan string list menjadi string

```
import ast
     def join text list(texts):
         texts = ast.literal_eval(texts)
         return ' '.join([text for text in texts])
     dm["tweet_join"] = dm["tweet_clean"].apply(join_text_list)
     dm["tweet_join"].head()
₹
                                             tweet_join
             abraj al bait clock tower beams indicating com...
         accounts recognised ramadan none recognised be...
      2
           admin post peaceful ramadan celebrations east ...
      3
                              admin post ramadan norway
           admin post ramadan usual peaceful start englan...
     dtype: object
```

c. Menghitung TF-IDF

```
Menghitung TF-IDF menggunakan TfidfVectorizer Untuk menghitung TF-IDF menggunakan Scikit-Learn, dapat dilakukan dengan cara berikut,

from sklearn.feature_extraction.text import TfidfVectorizer

# banyaknya term yang akan digunakan,
# di pilih berdasarkan top max_features
# yang diurutkan berdasarkan term frequency seluruh corpus
max_features = 10000

# Feature Engineering
print ("------ TF-IDF on Tweet data ------")

tf_idf = TfidfVectorizer(max_features=max_features, binary=True)
tf_idf.fit(dm["tweet_join"])  # Melakukan fitting terlebih dahulu
tfidf_mat = tf_idf.transform(dm["tweet_join"]).toarray()

print("TF-IDF ", type(tfidf_mat), tfidf_mat.shape)

TF-IDF <class 'numpy.ndarray'> (836, 3524)

Hasil dari tfidf menggunakan dataframe
```

d. Hasil Matriks TF-IDF

```
Hasil dari tfidf menggunakan dataframe
  [ ] import pandas as pd
                   # Mengambil nama fitur (term)
                   feature names = tf idf.get feature names out()
                   # Membuat DataFrame
                   df tfidf = pd.DataFrame(tfidf mat, columns=feature names)
                   # Menampilkan DataFrame
                   display(df_tfidf.head(20))
                aa aameen aamiin aaron aas abandoned abandoning abducted abdul abdullah ... zahra zakat zarafshan zardari zaria zayn zazzau zealand zelensky zumrat
1 0.0 0.0 0.0 0.0 0.0
                                                                     0.0
                                                                                       0.0
                                                                                                    0.0 0.0
                                                                                                                              0.0 ... 0.0 0.0
                                                                                                                                                                             0.0 0.0 0.0 0.0 0.0 0.0

    3
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
    0.0
                                                                                                                                                                                                                                                       0.0
                                                                                                                                                                                                                                                      0.0 0.0
                                                                    0.0 0.0 0.0 0.0 0.0 ... 0.0 0.0
        5 0.0 0.0 0.0 0.0 0.0
                                                                                                                                                                             0.0 0.0 0.0 0.0 0.0 0.0
                                                                                                                                                                                                                                                       0.0 0.0

      7
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
                                                                                                                                                                                                                          0.0
                                                                                                                                                                                                                                        0.0
                                                                                                                                                                                                                                                        0.0
                                                                                                                                                                                                                          0.0 0.0
                                                                                                                                                                                                                                                      0.0 0.0
                                                                                                                                                                                                                          0.0
                                                                                                                                                                                                                                                        0.0

      11
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.
                                                                     0.0 0.0 0.0 0.0
                                                                                                                                0.0 ...
        13 0.0
                        0.0
                                    0.0 0.0 0.0
                                                                                                                                                  0.0 0.0
                                                                                                                                                                              0.0
                                                                                                                                                                                           0.0 0.0 0.0

      15
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.0
      0.
                                                                                                                                                                                                                                                      0.0 0.0
        17 00
                         0.0
                                    00 00 00
                                                                       0.0
                                                                                        0.0
                                                                                                       0.0 0.0
                                                                                                                                 0.0
                                                                                                                                                   0.0 0.0
                                                                                                                                                                              0.0
                                                                                                                                                                                           00 00 00
                                                                                                                                                                                                                                        0.0
       19 00
                       20 rows × 3524 columns
melihat list top 50 term vang meiliki TF-IDF terbesar
```

Baris mewakili tiap dokumen/data, sedangkan kolom menunjukkan semua kata yang ada di dalam korpus. Jika sebuah kata muncul dalam suatu dokumen, maka pada perpotongan baris dan kolom tersebut akan terisi nilai TF-IDF-nya.

e. Menampilkan ke dalam list

50 term teratas dengan total TF-IDF tertinggi.

```
terms = tf_idf.get_feature_names_out()
    # sum tfidf frequency of each term through documents
    sums = tfidf_mat.sum(axis=0)
    # connecting term to its sums frequency
    data = []
    for col, term in enumerate(terms):
        data.append((term, sums[col] ))
     ranking = pd.DataFrame(data, columns=['term','rank'])
     ranking.sort_values('rank', ascending=False)
₹
               term
                         rank
     2480
          ramadan 54.951769
     2000
             month 20.407185
           mubarak 16.447736
     2021
     1913
               may
                    16.075416
      109
               allah
                    15.525257
     2681
                      0.176130
                 ГΧ
     1828
            location
                      0.176130
     1687
            keyless
                      0.176130
      93
            akpabio
                      0.176130
     3441
               wike
                      0.176130
    3524 rows x 2 columns
```

```
data = pd.read_csv("/content/ramadan_clean_tweet.csv")
data.to_excel("ramadan_clean_tweet_excel.xlsx", index=False)
```