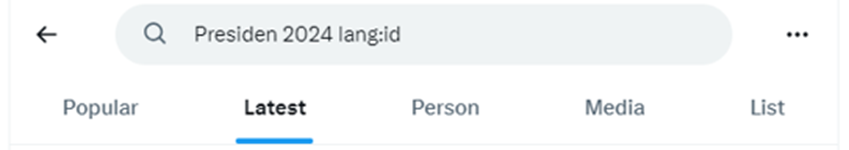
1. **CRAWLING DATA TWITTER (1 tanpa batas waktu)**
2. **Twitter Authentikasi**
3. Pastikan bahwa kamu sudah login pada twitter.
4. Ketikkan judul yang ingin kamu cari (disini kami contohkan: presiden 2024 lang:id).



1. Lalu pilih latest untuk jumlah data crawling yang lebih maksimal



1. Seteleh itu klik kanan lalu pilih inspect

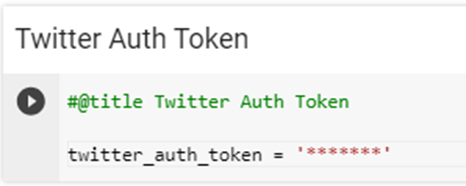


1. Pilihlah application(1), pilih cookies (2), kemudian pilih<https://twitter.com> (3), dan pilihlah auth\_token(4). Jika sudah dipilih maka nanti akan muncul seperti ini, copy saja token autentikasinya(5) (ini bersifat rahasia jadi gunakanlah seefektif mungkin.)





1. **Proses Crawling Data Twitter**
2. Buka google colabs lalu ketikkan kode seperti dibawah ini. Jika jangan lupa untuk menginputkan autentikasi token yang kita dapat dari twitter yang sudaah kita copy tadi. Lalu run (ganti (\*) dengan token kalian masing-masing)



Import dan instalasi beberapa shell yang dieksekusi dalam lingkungan Python menggunakan sintaksis khusus (dimulai dengan !).

1. Lakukan Pemasangan Paket Python Pandas:



1. lakukan instalasi Node.js

# Install Node.js (because tweet-harvest built using Node.js)

!sudo apt-get update

!sudo apt-get install -y ca-certificates curl gnupg

!sudo mkdir -p /etc/apt/keyrings

!curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/nodesource.gpg

!NODE\_MAJOR=20 && echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node\_$NODE\_MAJOR.x nodistro main" | sudo tee /etc/apt/sources.list.d/nodesource.list

!sudo apt-get update

!sudo apt-get install nodejs -y

!node -v

1. Crawling Data dari Twitter API.

lakukan proses crawling dengan menggunakan kode seperti

# Crawl Data

filename = 'Presiden2024.csv'*(sesuaikan dengan nama file yang kalian mau)*

search\_keyword = 'Presiden2024 lang :id'*(sesuaikan dengan keyword masing-masing)*

limit = 1500

!npx --yes tweet-harvest@2.2.8 -o "{filename}" -s "{search\_keyword}" -l {limit} --token {twitter\_auth\_token}

1. Membaca dan Menampilkan Data yang Sebelumnya Telah Dikumpulkan dari Twitter dan Disimpan dalam Format CSV**.**

lakukan lah dengan menuliskan kode seperti berikut:

**import pandas as pd**

**# Specify the path to your CSV file**

**file\_path = f"tweets-data/{filename}"**

**# Read the CSV file into a pandas DataFrame**

**df = pd.read\_csv(file\_path, delimiter=";")**

**# Display the DataFrame**

**display(df)**

1. Cek Data Yang diDapatkan

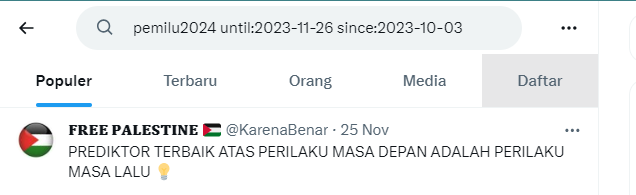
Lakukaan dengan menjalankan kode seperti dibawah ini:

**# Cek jumlah data yang didapatkan**

**num\_tweets = len(df)**

**print(f"Jumlah tweet dalam dataframe adalah {num\_tweets}.")**

1. **CRAWLING DATA TWITTER (2 menggunakan batas waktu)**
2. **Menyiapkan topik yang akan di crawling**
3. Pastikan kamu sudah login di akun twitter
4. Ketikkan judul yang ingin kamu cari (disini kami contohkan: pemilu2024 until:2023-11-26 since:2023-10-03).



1. Seteleh itu klik kanan lalu pilih inspect



1. Pilihlah application(1), pilih cookies (2), kemudian pilih<https://twitter.com> (3), dan pilihlah auth\_token(4). Jika sudah dipilih maka nanti akan muncul seperti ini, copy saja token autentikasinya(5) (ini bersifat rahasia jadi gunakanlah seefektif mungkin.)





1. **Proses Crawling Data Twitter dengan Batasan waktu**
2. Buka google colabs lalu ketikkan kode seperti di bawah ini. Ini merupakan proses install package dan beberapa Node.js

# Import required Python package

!pip install pandas

# Install Node.js (because tweet-harvest built using Node.js)

!sudo apt-get update

!sudo apt-get install -y ca-certificates curl gnupg

!sudo mkdir -p /etc/apt/keyrings

!curl -fsSL https://deb.nodesource.com/gpgkey/nodesource-repo.gpg.key | sudo gpg --dearmor -o /etc/apt/keyrings/nodesource.gpg

!NODE\_MAJOR=20 && echo "deb [signed-by=/etc/apt/keyrings/nodesource.gpg] https://deb.nodesource.com/node\_$NODE\_MAJOR.x nodistro main" | sudo tee /etc/apt/sources.list.d/nodesource.list

!sudo apt-get update

!sudo apt-get install nodejs -y

!node -v

1. Selanjutnya adalah proses crawling datanya, masukkan filename, keyword yang anda cari di twitter tadi, lalu pasang limitnya maksimal 500

# Crawl Data

filename = 'pemilu2024.csv'

search\_keyword = 'pemilu2024 until:2023-11-26 since:2023-10-03'

limit = 50

!npx --yes tweet-harvest@latest -o "{filename}" -s "{search\_keyword}" -l {limit} --token ""

Nanti akan diminta token, token tersebut di dapat dari kode autentikasi di twitter yang sudah disiapkan tadi.

1. Selanjutnya adalah proses membaca file CSV

import pandas as pd

# Specify the path to your CSV file

file\_path = f"tweets-data/{filename}"

# Read the CSV file into a pandas DataFrame

df = pd.read\_csv(file\_path, delimiter="pemilu2024;", engine='python')

# Display the DataFrame

display(df)

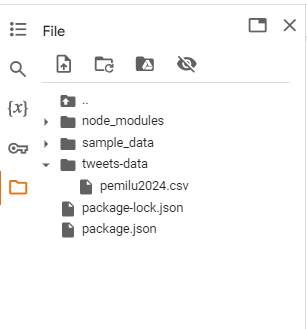
Selanjutnya adalah mengecek berapa banyak jumlah data yang didapatkan

# Cek jumlah data yang didapatkan

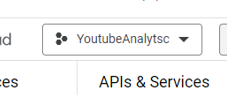
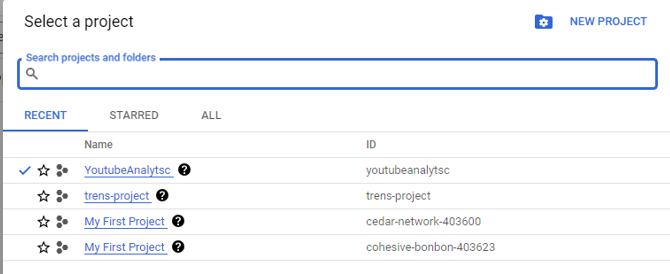
num\_tweets = len(df)

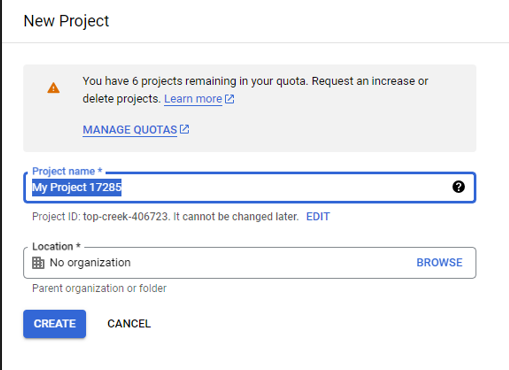
print(f"Jumlah tweet dalam dataframe adalah {num\_tweets}.")

1. Nah setelah menyelesaikan serangkaian proses diatas maka kita nanti dapat melihat file hasil crawling nya pada file tweets-data seperti dibawah, dan anda tinggal download hasilnya

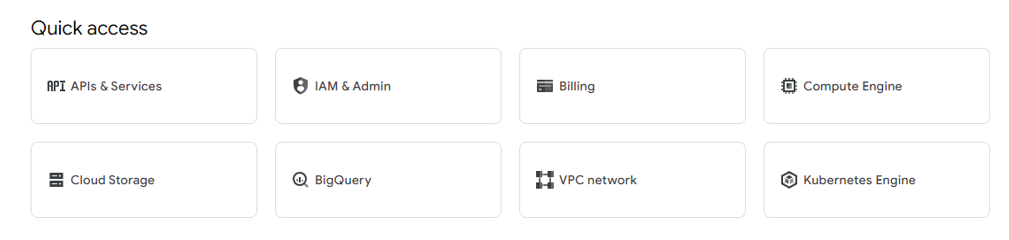


1. **CRAWLING DATA YOUTUBE**
2. **Menyiapkan google cloud function**
3. Lakukan login atau registrasi google cloud function
4. buat New Project

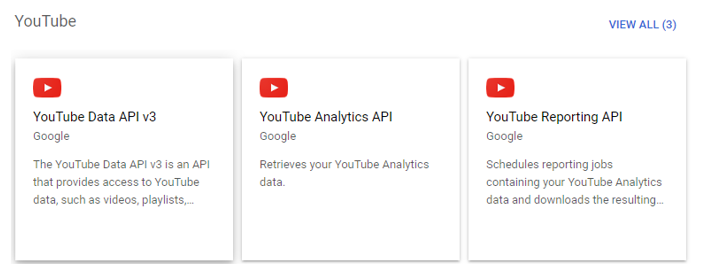
 



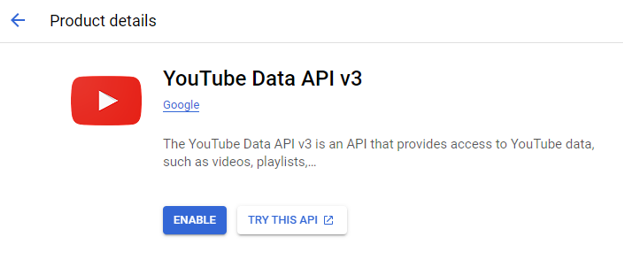
1. Jika sudah berhaasil create new project, masuk dan cari Quick Access. kemudian pilih APIs & Services



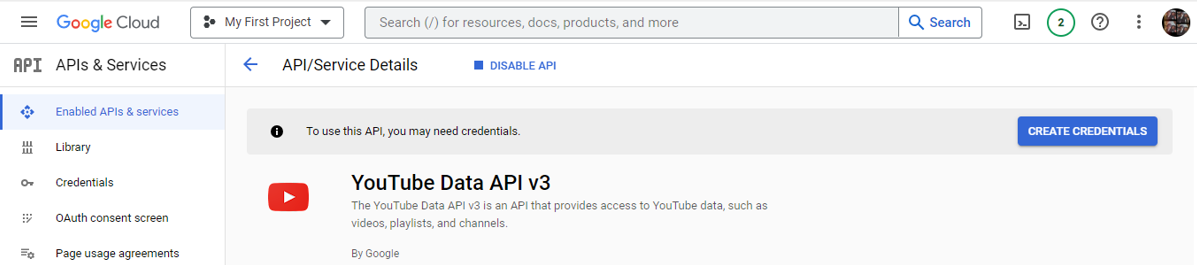
1. scrool ke bawah hingga menemukan Youtube seperti dibawah, kemudian pilih Youtube Data API v3



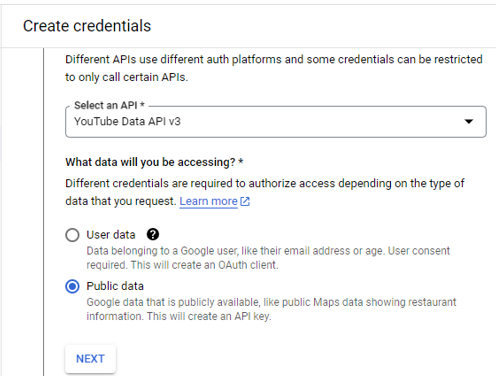
1. Pilih ENABLE



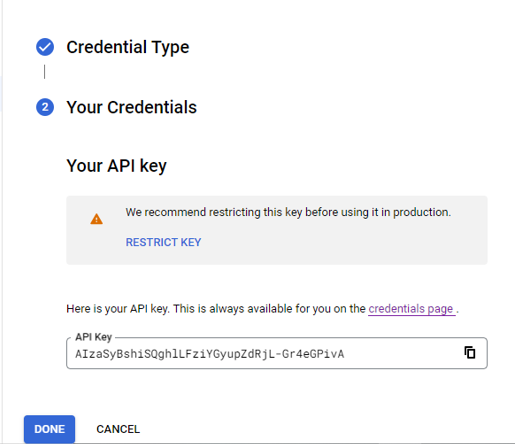
1. Lakukan Create credentials, Ikuti prosesnya, serta isilah apa yang diminta sesuai dengan data kamu.



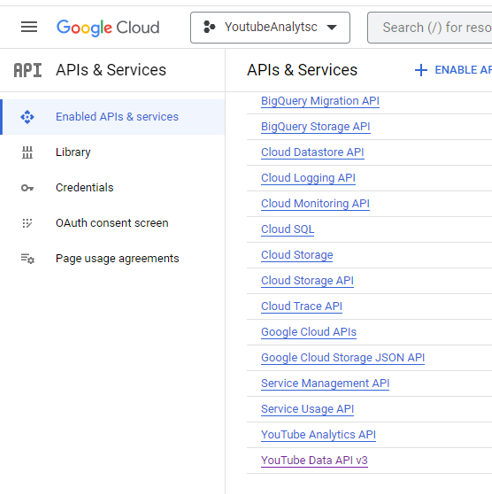
Pilih Next,



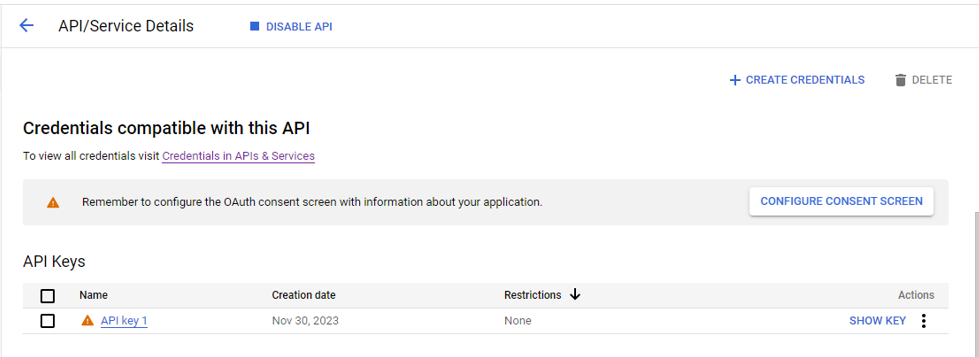
Api Key diatas yang nantinya akan anda gunakan dalam proses crawling data. jangan lupa pilih button Done.



1. Jika kredensial yang kamu buat sudah benar maka akan muncul pada APIs & SERVICES seperti dibawah ini.



Double klik pada kredensial yang kamu buat maka akan diarahkan ke halaman API Keys. lalu bisa pilih show Key untuk melihat API Key kamu. API ini yang nanti akan digunakan dalam proses crawling data komentar youtube



1. **Proses Crawling Data**
2. Mengimport Library pandas dan Koneksi ke layanan API google, menggunakan kode seperti berikut:

import pandas as pd

from googleapiclient.discovery import build

1. Kode berikut ini merupakan kode Python yang gunanya untuk mengambil komentar serta balasan komentar dari video Youtube yang dipilih menggunakan Youtube API v3 tadi.

def video\_comments(video\_id):

# empty list for storing reply

replies = []

# creating youtube resource object

youtube = build('youtube', 'v3', developerKey=api\_key)

# retrieve youtube video results

video\_response = youtube.commentThreads().list(part='snippet,replies', videoId=video\_id).execute()

# iterate video response

while video\_response:

# extracting required info

# from each result object

for item in video\_response['items']:

# Extracting comments ()

published = item['snippet']['topLevelComment']['snippet']['publishedAt']

user = item['snippet']['topLevelComment']['snippet']['authorDisplayName']

# Extracting comments

comment = item['snippet']['topLevelComment']['snippet']['textDisplay']

likeCount = item['snippet']['topLevelComment']['snippet']['likeCount']

replies.append([published, user, comment, likeCount])

# counting number of reply of comment

replycount = item['snippet']['totalReplyCount']

# if reply is there

if replycount>0:

# iterate through all reply

for reply in item['replies']['comments']:

# Extract reply

published = reply['snippet']['publishedAt']

user = reply['snippet']['authorDisplayName']

repl = reply['snippet']['textDisplay']

likeCount = reply['snippet']['likeCount']

# Store reply is list

#replies.append(reply)

replies.append([published, user, repl, likeCount])

# print comment with list of reply

#print(comment, replies, end = '\n\n')

# empty reply list

#replies = []

# Again repeat

if 'nextPageToken' in video\_response:

video\_response = youtube.commentThreads().list(

part = 'snippet,replies',

pageToken = video\_response['nextPageToken'],

videoId = video\_id

).execute()

else:

break

#endwhile

return replies

1. Berikut ini merupakan proses memfungsikan kode yang telah dideklarasikan pada kode sebelumnya. Lebih spesifiknya mengambil komentar youtube menggunakan API sesuai dengan video yang dipilih. Berikut kode yang di maksud:

# isikan dengan api key Anda

api\_key = '\*\*\*\*\*\*\*\*'

# Enter video id

# contoh url video =https://www.youtube.com/watch?v=2iZdW\_\_BR5w

video\_id = "2iZdW\_\_BR5w" #isikan dengan kode / ID video

# Call function

comments = video\_comments(video\_id)

comments

1. Selanjutnya adalah memfungsikan pandas, agar komen youtube yang sudah kita ambil tadi bisa lebih rapi dan mudah dimengerti, seperti berikut:

df = pd.DataFrame(comments, columns=['publishedAt', 'authorDisplayName', 'textDisplay', 'likeCount'])

df

1. Yang terakhir adalah mengekspor data frame ke dalam bentuk csv. Serta menghindari menyimpan index data frame ke dalam file csv.

df.to\_csv('youtube-comments.csv', index=False)

1. **CLEANING DATA TWITTER DAN YOUTUBE**
2. **Siapkan Data Crawling dalam bentuk csv**
3. **Proses Cleaning Data**
4. Lakukan impor modul dengan menggunakan kode berikut:

import sys

sys. Version

1. Lakukan instalasi beberapa paket python

!pip install emoji

!pip install Sastrawi

!pip install gensim

!pip3 install swifter

1. Lakukan instalasi paket TextBlob

pip install textblob

1. Lakukan instalasi atau upgrade paket deep-translator

!pip install -U deep-translator

1. Lakukan instalasi imblearn

pip install imblearn

1. Lakukaan instalasi paket emoji

pip install emoji --upgrade

1. Lakukan instalasi wordcloud

!pip install wordcloud

1. Lakukan download nltk dan import

import nltk

from nltk.tokenize import word\_tokenize

# UNCOMMENT "nltk.download()" untuk  pertamakali run, untuk run selanjutnya comment kembali

# nltk.download()

1. Lakukan text preprocessing dan machine learning tasks.

import pandas as pd

import numpy as np

import re

import string

# import emoji

# import Sastrawi package

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

# from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory

from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory, StopWordRemover, ArrayDictionary

import swifter

from tqdm import tqdm

# from nltk.tokenize import word\_tokenize

# import nltk

# nltk.download()

import ast

import gensim

from gensim import corpora, models

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import confusion\_matrix,classification\_report, accuracy\_score

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.naive\_bayes import MultinomialNB

import matplotlib.pyplot as plt

import seaborn as sns

from wordcloud import WordCloud

1. Setelah selesai dan berhasil dengan kodingan di atas, selanjutnya adalah mengiport google translator

from deep\_translator import GoogleTranslator

1. Jika sudah lakukan pemanggilan dari file yang akan di cleaning, pemanggilan ini sesuaikan dengan letak serta nama dari file tersebut, disini dimisaalkan:

dfc = pd.read\_csv("prabowogibran.csv", delimiter= ";")

1. Berikut adalah kodingaan yang berguna menampilkan data yang sudah di kelompokkan.

dfc.head()