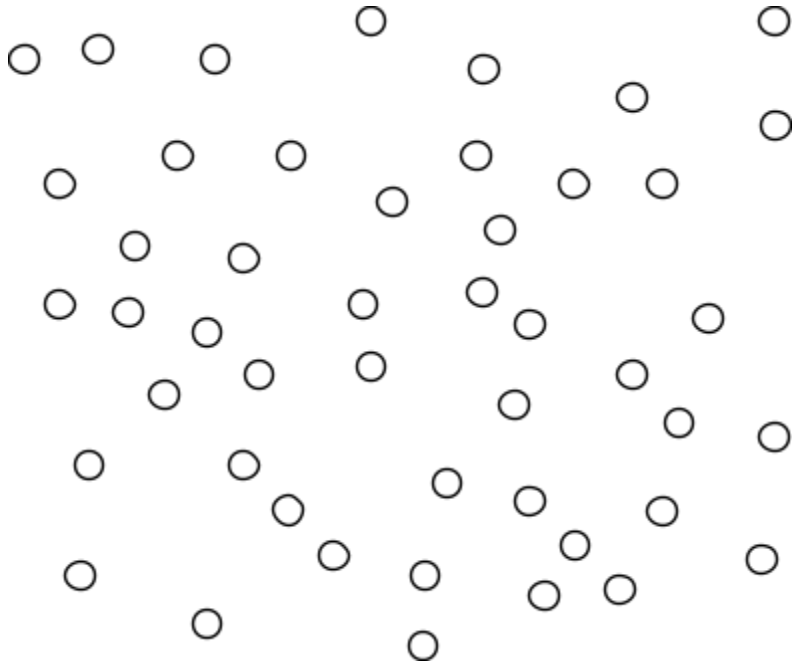
Introduction of Clustering Method and Algorithm



Written By MUHAMMAD ADHIEM WICAKSANA

Supported By Bengkel Koding (**Universitas Dian Nuswantoro**)

Daftar isi

“Klik untuk langsung lompat ke halaman yang ingin dituju”

Pendahuluan

kamu udah tau apa itu **Clustering ?,** Kalo belum tau saya jelasin dulu ya.
Clustering adalah proses pengelompokan data menjadi beberapa cluster atau kelompok, sehingga data di dalam cluster memiliki kemiripan yang maksimal dan data antar cluster memiliki kemiripan yang minimal.

“Untuk memulai modul ini kamu harus nyiapin apa aja ? “

Yang paling pertama kali wajib kamu siapkan adalah kemampuan basic bahasa pemrograman **Python,** karena di modul ini kita bakal belajar menggunakan **Python.**

“Terus, tool yang bakal digunakan apa saja ?”

IDE yang saya pakai untuk melakukan demonstrasi di modul ini adalah **Jupyter Notebook dan** juga **Anaconda** sebagai **environment manager.**

Dan juga **Jupyter Notebook-**nya saya run di dalam **Visual Studio Code,** jadi jangan bingung ya kalo User Interfacenya berbeda dengan **Jupyter Notebook** pada umumnya.

Kamu juga bisa menggunakan platform cloud computing gratis seperti **google colab** atau **Jetbrain Data Lore**,

“kok pakai platform cloud computing, emang machine learning seberat itu ya ? ”

Iya, kalo kamu udah implementasi machine learning ke dalam project mu, performa komputer yang kamu butuhkan untuk menjalankan project mu nggak sedikit, jadi kalo kalian punya komputer dengan spesifikasi yang apa adanya kalian bisa pakai platform cloud computing yang sudah saya sebutkan diatas.
Untuk file codingan yang tersedia di dalam modul ini bisa dilihat di : [madhiemw/Clustering-Algorithm-Module (github.com)]

Metode dan Algoritma

Setelah membahas hal mendasar mengenai ***Clustering*** di bagian ini kita bakal bahas lebih jauh tentang ***clustering***, sesuai namanya ***clustering*** memiliki fungsi untuk meng-cluster atau mengelompokan.

Dalam proses pengelompokan, clustering memiliki 5 macam Metode, dan setiap metode memiliki algoritmanya sendiri-sendiri. Di modul ini kita bakal bahas satu-persatu ya.

1. Partitioning-based Clustering
2. Hierarchical-based Clustering
3. Density-based Clustering
4. Grid-based Clustering
5. Model-based Clustering

Dalam proses pengenalan metode dan algoritma clustering, library yang saya pakai bakal berbeda beda. Jadi jangan lupa untuk install library yang belum kamu punya dengan **“pip install (nama library yang mau diinstal tanpa kurung)”**

Partitioning-based Clustering

Metode pertama yang bakal saya bahas adalah **Partitioning-based clustering**.
Metode ini bekerja dengan cara mempartisi object ke dalam beberapa cluster atau biasa disebut **k-number**, dimana setiap cluster memiliki keunikannya sendiri sendiri dan tidak akan sama dengan cluster lainnya.

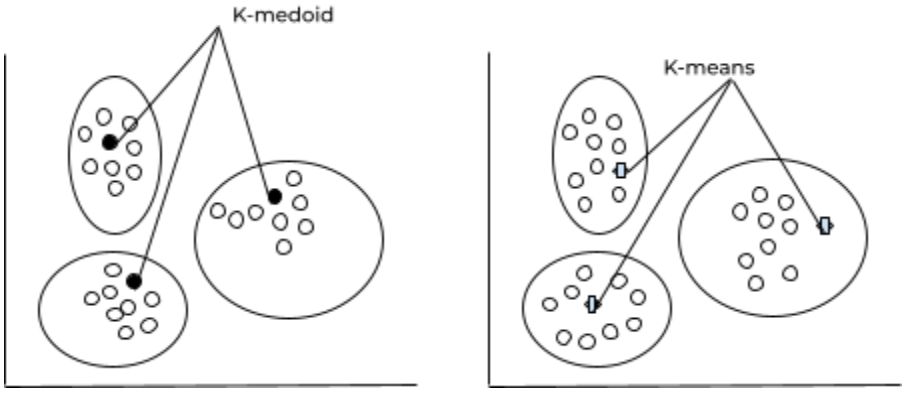
Algoritma **k-means** dan **k-medoid/PAM** adalah beberapa algoritma yang menggunakan metode ini.

“Bedanya k-means sama k-medoid apa ?, kan namanya hampir sama”

Cara kerja mereka berdua memang sangat identik bahkan sama, hanya saja proses penentuan data point untuk acuan setiap cluster yang berbeda.

Data point di setiap cluster di sebut ***Centroid Cluster.***
K-means menjadikan nilai rata-rata sebagai **centroid cluster.**
Cara kerjanya adalah dengan cara menghitung nilai rata rata (mean) dari setiap object yang akan di cluster. Setelah proses penghitungan selesai algoritma akan memasukan setiap object dengan nilai rata-rata yang dekat kedalam 1 cluster yang sama.

Sedangkan **k-medoid** memiliki cara kerja yang lebih spesifik, dimana algoritma langsung menggunakan perwakilan object (medoid) sebagai **centroid cluster**.



Oh iya, Untuk menentukan jumlah cluster (nilai K) yang akan ditetapkan, kamu bisa menggunakan **metode elbow** dan **metode Silhouette**, metode elbow bekerja dengan cara melihat persentase hasil perbandingan antara jumlah cluster yang akan membentuk sudut siku pada suatu titik. Sedangkan metode Silhouette adalah metode yang difungsikan untuk melihat kualitas dan kekuatan cluster, seberapa baik suatu objek ditempatkan dalam suatu cluster.

Selanjutnya adalah tutorial implementasi algoritma ini ke dalam codingan. Saya akan melanjutkan ke sesi coding.

Tutorial penggunaan K-means di python :

```
import pandas as pd
df = pd.read_csv("scraping.csv")
df
```

	Unnamed: 0	text
0	0	Timestamp :\n0:00 opening\n0:25 disclaimer \n1...
1	1	Team speaker Eggel, sudah punya yang Eggel Fit...
2	2	Wah kebetulan lagi nyari Speaker Bluetooth yan...
3	3	Masih QCY Box 2 tetap di hati, gua punya speak...
4	4	This is where local industries start rising. P...
...	...	...
1419	1419	nah
1420	1420	Tod bacod
1421	1421	sori gw dislike, textnya mengganggu gambar
1422	1422	Rinrei
1423	1423	31

1424 rows × 2 columns

Untuk clustering dengan k-means langkah pertama, kamu harus import dataset yang sudah kamu siapkan kedalam python. Dengan menggunakan library pandas.

Setelah kamu upload dataset yang kamu punya, langsung lanjut ke tahap *data cleaning* dimana di tahap ini kamu bakal membersihkan data dari noise-noise yang mengganggu selama proses clustering, contoh : titik dan koma, time stamp, garis miring dll

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Setelah data sudah bersih kita lanjut ke proses vectorization.

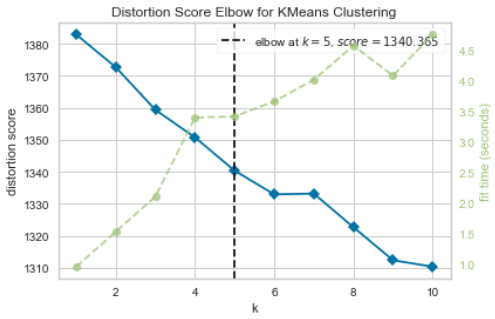
Vectorization adalah proses pengubahan string menjadi bentuk numerical sehingga algoritma bisa melakukan penghitungan guna menentukan **centroid cluster.**

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

Setelah vectorizer tahap selanjutnya adalah menentukan jumlah cluster yang paling optimal untuk data yang akan kita cluster, metode yang akan digunakan adalah **elbow method.** Penggunaan metode penentu cluster tidak wajib dilakukan, karena kamu juga bisa bereksperimen dengan menentukan jumlah cluster sendiri.

```
from sklearn.cluster import KMeans
from yellowbrick.cluster import KElbowVisualizer # cluster visualizer
import matplotlib.pyplot as plt
%matplotlib inline
model = KMeans()
visualizer = KElbowVisualizer(model, k=(1, 11))

visualizer.fit(x)
visualizer.show()
plt.show()
```

✓ 33.3s



Setelah mendapatkan **elbow score,** tahap terakhir adalah melakukan implementasi algoritma clustering ke dalam codingan yang sudah kita buat.

```
from sklearn.cluster import KMeans
true_k = 6
model = KMeans(n_clusters=true_k, init='k-means++', max_iter=1423, n_init=1423)
model.fit(x)
labels=model.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])
print(new_df.sort_values(by=['cluster']))
```

Ku Jelaskan ya, di baris pertama saya pakai **sklearn** sebagai **library** untuk penggunaan algoritma.

variable “true_k” digunakan untuk menjadi variabel integer penentu jumlah cluster.

Setelah menentukan “true_k” lanjut ke baris selanjutnya yaitu **model**, variabel ini adalah konfigurasi untuk algoritma yang akan kita pakai.

- K Means adalah nama algoritma yang sudah kita import melalui library sklearn.
- n_cluster adalah jumlah cluster yang ingin kita buat.
- max_iter adalah jumlah maximal algoritma akan mengacak dan menghitung ulang cluster
- n_init adalah jumlah maksimal pengacakan centroid pada setiap cluster
- Lalu untuk meng-export hasil clustering kamu bisa pakai “to.csv(namaFile.csv)”

Tutorial penggunaan algoritma cara menggunakan K-medoid di python:

Untuk **k-medoid** cara implementasi hampir sama dengan **K-mean,** tapi disini saya pakai library yang berbeda yaitu **scikit-learn-extra.**

“loh, kan sama-sama **scikit-learn** bedanya apa ?”

Scikit-learn-extra adalah library extensi untuk scikit-learn. Di library **Scikit-learn-extra** memiliki 2 algoritma **clustering** yaitu **CNN** dan **K-medoid.**

```
import pandas as pd
df = pd.read_csv("scraping.csv")
df
```

	Unnamed: 0	text
0	0	Timestamp :\n0:00 opening\n0:25 disclaimer \n1...
1	1	Team speaker Eggel, sudah punya yang Eggel Fit...
2	2	Wah kebetulan lagi nyari Speaker Bluetooth yan...
3	3	Masih QCY Box 2 tetap di hati, gua punya speak...
4	4	This is where local industries start rising. P...
...	...	...

Untuk proses input data masih sama dengan tutorial sebelumnya yaitu menggunakan **pandas.**

f

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Setelah kamu upload dataset yang kamu punya, langsung lanjut ke tahap *data cleaning,* dimana di tahap ini kamu bakal membersihkan data dari noise-noise yang mengganggu selama proses clustering, contoh : titik dan koma, time stamp, garis miring dll

Proses dilanjutkan dengan proses **vectorization,** proses vectorizer dilakukan untuk mengubah string menjadi numeric agar algoritma bisa melakukan penghitungan guna menentukan sebuah cluster.

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

Lalu yang terakhir adalah implementasi algoritma ke dalam codingan yang sudah di buat menggunakan library sklearn-extra.

```
from sklearn_extra.cluster import KMedoids
kmedoids = KMedoids(n_clusters=11 ,metric='euclidean',init='k-medoids++', max_iter= 1000000,random_state=1000000).fit(data)
labels= kmedoids.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])
print(new_df.sort_values(by=['cluster']))

new_df.to_csv('result.csv')
```

Library yang di gunakan adalah sklearn

Di baris kedua variable **k medoids** berisi konfigurasi algoritma yang akan dipakai

n_cluster = variabel yang digunakan untuk menentukan jumlah cluster

Metric = variable penentu jarak

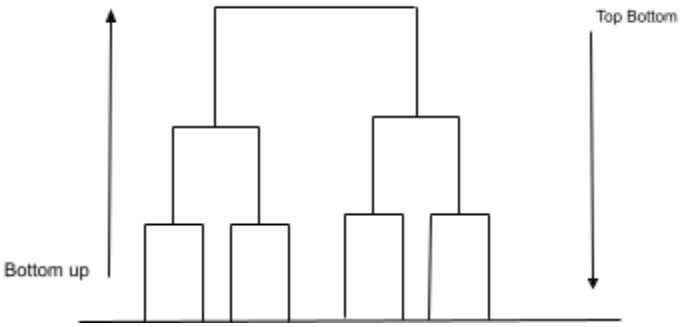
Max_iter = jumlah maksimal pengulangan penentuan cluster

Random_state = variabel yang akan melakukan pengacakan angka medoid

Hierarchical-based Clustering

Setelah saya kasih penjelasan tentang **Partitioning-based clustering** sekarang saya bakal bahas **Hierarchical-based Clustering.** Metode ini bekerja dengan cara mengelompokan setiap objek ke satu atau lebih kelompok lalu dilanjutkan lagi dengan memecah ke setiap kelompok ke kelompok lainnya dan terus sampai algoritma benar benar menemukan cluster yang tepat untuk setiap objek atau text.

Metode ini juga terbagi menjadi 2 cara kerja yaitu **Top bottom dan bottom up approach.**



“Bentar-bentar, memangnya mau belajar matkul manajemen proyek ?”

Hehehe, nggak kok memang metode ini terbagi menjadi 2 cara kerja, **top bottom** dan **bottom up approach**

Beberapa algoritma **Hierarchical-based Clustering** yang akan kita bahas adalah algoritma **Agglomerative Clustering (AGNES),** algoritma ini menggunakan cara kerja **bottom up**. Yang kedua adalah **Divisive Analysis(DIANA)** algoritma ini menggunakan cara kerja **top bottom**.

Agglomerative Clustering (AGNES) mengawali proses kerja dengan menjadikan setiap data sebagai 1 cluster. Jadi kalo kamu punya 1400 data untuk di cluster, algoritma ini akan mengawali proses clustering dengan membuat 1400 cluster, proses dilanjutkan dengan menggabungkan 2 cluster yang memiliki kesamaan, proses akan terus dilanjutkan sampai proses pembagian cluster selesai.

Divisive Analysis(DIANA) bekerja dengan cara yang terbalik dengan **Agglomerative Clustering (AGNES).** Dimana proses kerja diawali dengan menjadikan semua data sebagai 1 cluster yang besar, dilanjutkan dengan membagi menjadi 2 cluster dan

terus dilakukan hingga proses clustering sampai di tahap menyisakan 1 data di setiap clusternya.

Selanjutnya adalah tutorial penggunaan algoritma ini ke dalam codingan.

Tutorial penggunaan algoritma Agglomerative Clustering (AGNES) di python :

```
import pandas as pd
df = pd.read_csv("scraping.csv")
df
```

	Unnamed: 0	text
0	0	Timestamp :\n0:00 opening\n0:25 disclaimer \n1...
1	1	Team speaker Eggel, sudah punya yang Eggel Fit...
2	2	Wah kebetulan lagi nyari Speaker Bluetooth yan...
3	3	Masih QCY Box 2 tetap di hati, gua punya speak...
4	4	This is where local industries start rising. P...
...	...	...

Langkah pertama adalah dengan import dataset yang ingin di cluster kedalam python dengan menggunakan library pandas.

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Setelah kamu upload dataset yang kamu punya, langsung lanjut ke tahap *data cleaning* dimana di tahap ini kamu bakal membersihkan data dari noise-noise yang mengganggu selama proses clustering, contoh : titik dan koma, time stamp, garis miring dll

Setelah data sudah bersih kita lanjut ke proses **vectorization**.

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

Vectorization adalah proses pengubahan string menjadi bentuk numerical sehingga algoritma bisa membaca kata apa yang akan di cluster.
Setelah **Vectorization,** tahap selanjutnya adalah melakukan implementasi algoritma clustering ke dalam codingan yang sudah kita buat.

```
from scipy.cluster.hierarchy import ward, dendrogram, fcluster, single, complete
from sklearn.cluster import AgglomerativeClustering
import matplotlib.pyplot as plt
%matplotlib inline

model = AgglomerativeClustering(distance_threshold=2, n_clusters=None).fit(x)
labels= model.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])



print(new_df.sort_values(by=['cluster']))
new_df.to_csv('hierarchial.csv')
```

saya pakai **scipy** sebagai **library** untuk otak dari visualisasi proses kerja algoritma.

Sklearn digunakan untuk implementasi algoritma clustering.

Matplotlib digunakan untuk melakukan visualisasi data yang nanti akan dihubungkan dengan **Scipy.**

Variable **"model"** digunakan untuk melakukan tuning untuk algoritma yang akan di pakai.

Variable "**new_df"** difungsikan untuk apply hasil clustering ke dalam dataset sekaligus convert numerical kembali menjadi string.

Density-based Clustering

Salah satu algoritma yang menggunakan metode ini adalah **Density-Based Spatial Clustering of Application with Noise (DBSCAN)** dan **Ordering Points To Identify The Clustering Structure (OPTICS).**

Density-Based Spatial Clustering of Application with Noise (DBSCAN) adalah algoritma yang mengelompokan data berdasarkan kepadatan yang terkoneksi.
Dalam proses clustering. **DBSCAN** memerlukan 2 parameter yaitu **epsilon (eps)** dan **minimum points (minPts). Epsilon** adalah jarak maksimal antara dua data dalam 1 cluster yang diizinkan dan **minimum points (minPts)** adalah banyaknya data dalam jarak **epsilon** agar terbentuk suatu cluster. DBSCAN adalah salah satu algoritma yang efisien untuk mengklaster data dengan **outliers atau jumlah noise yang banyak.**

“Outliers itu apa ? ”

Dalam ilmu statistika, outliers adalah data dengan karakteristik yang berbeda jauh dari data atau observasi observasi lainnya. Hal ini juga yang membuat outliers sebagai musuh utama dalam proses clustering karena dapat mempengaruhi hasil akhir clustering.

Ordering Points To Identify The Clustering Structure (OPTICS) adalah algoritma penyempurna dari **Density-Based Spatial Clustering of Application with Noise (DBSCAN),** perbedaanya yang paling mencolok adalah **OPTICS** tidak memerlukan konsistensi kepadatan dalam sebuah data untuk melakukan clusterisasi, ini menutup kekurangan algoritma **DBSCAN** dalam melakukan clustering, dimana **DBSCAN** tidak bisa meng-cluster data dengan jarak yang tidak konsisten.

Selanjutnya adalah tutorial implementasi algoritma ini ke dalam codingan.

Tutorial penggunaan DBSCAN di python :

```
import pandas as pd
df = pd.read_csv("scraping.csv")
df
```

	Unnamed: 0	text
0	0	Timestamp :\n0:00 opening\n0:25 disclaimer \n1...
1	1	Team speaker Eggel, sudah punya yang Eggel Fit...
2	2	Wah kebetulan lagi nyari Speaker Bluetooth yan...
3	3	Masih QCY Box 2 tetap di hati, gua punya speak...
4	4	This is where local industries start rising. P...
...	...	...
1419	1419	nah
1420	1420	Tod bacod
1421	1421	sori gw dislike, textnya mengganggu gambar
1422	1422	Rinrei
1423	1423	31

1424 rows × 2 columns

Langkah pertama adalah dengan import dataset yang ingin di cluster kedalam python dengan menggunakan library pandas.

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Setelah kamu upload dataset yang kamu punya, langsung lanjut ke tahap *data cleaning* dimana di tahap ini kamu bakal membersihkan data dari noise-noise yang mengganggu selama proses clustering, contoh : titik dan koma, time stamp, garis miring dll

Setelah data sudah bersih kita lanjut ke proses **vectorization**.

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

Vectorization adalah proses pengubahan string menjadi bentuk numerical sehingga algoritma bisa membaca kata apa yang akan di cluster.

Setelah **Vectorization,** tahap selanjutnya adalah melakukan implementasi algoritma clustering ke dalam codingan yang sudah kita buat.

```
from sklearn.cluster import DBSCAN
model =DBSCAN(eps=1, min_samples=2)
model.fit(x)
labels=model.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])
print(new_df.sort_values(by=['cluster']))
new_df.to_csv('new.csv')
```

Library yang di pakai untuk implementasi algoritma adalah **Sklearn.**

Variable **model** berisikan konfigurasi algoritma :

Eps adalah variabel yang digunakan untuk menentukan jarak antar data **(epsilon)** yang akan dipakai algoritma.

Min_samples adalah variabel yang digunakan untuk menentukan jumlah objek yang dijadikan core point.

Tutorial implementasi OPTICS di python :

```
import pandas as pd
df = pd.read_csv("scraping.csv")
df
```

	Unnamed: 0	text
0	0	Timestamp :\n0:00 opening\n0:25 disclaimer \n1...
1	1	Team speaker Eggel, sudah punya yang Eggel Fit...
2	2	Wah kebetulan lagi nyari Speaker Bluetooth yan...
3	3	Masih QCY Box 2 tetap di hati, gua punya speak...
4	4	This is where local industries start rising. P...
...	...	...

Langkah pertama adalah dengan import dataset yang ingin di cluster kedalam python dengan menggunakan library pandas.

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Setelah kamu upload dataset yang kamu punya, langsung lanjut ke tahap *data cleaning* dimana di tahap ini kamu bakal membersihkan data dari noise-noise yang mengganggu selama proses clustering, contoh : titik dan koma, time stamp, garis miring dll

Setelah data sudah bersih kita lanjut ke proses **vectorization**.

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

Vectorization adalah proses pengubahan string menjadi bentuk numerical sehingga algoritma bisa membaca kata apa yang akan di cluster.

Setelah **Vectorization,** tahap selanjutnya adalah melakukan konfigurasi algoritma clustering.

```
from sklearn.cluster import OPTICS
model = OPTICS(min_samples=10)
model.fit(x)
labels=model.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])
print(new_df.sort_values(by=['cluster']))
new_df.to_csv('new.csv')
```

Library yang di pakai untuk implementasi algoritma adalah **Sklearn.**

Variable **model** berisikan konfigurasi algoritma :

min_samples = minimal sample untuk setiap cluster

Grid-based Clustering

Salah satu contoh algoritma yang menggunakan metode ini adalah **statistical information grid-based approach (STING).**

statistical information grid-based approach (STING) adalah algoritma yang bekerja dengan menggunakan struktur multilayer grid lalu melakukan penghitungan untuk menentukan grid dan kisi yang akan ditetapkan.

Clique adalah algoritma yang menggunakan 2 metode sekaligus. **Grid-based method** dan **Density-based method**. Algoritma ini bekerja dengan mengambil density sebagai syarat terbentuknya cluster dan grid sebagai parameter penentu data di setiap cluster.

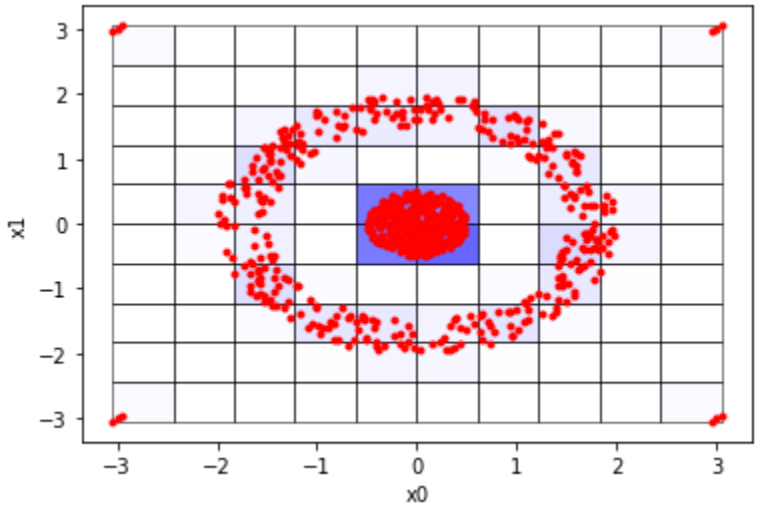
Selanjutnya saya bakal melampirkan contoh codingan **Clique.**

Kodingan di bawah menggunakan library **pyclustering**.

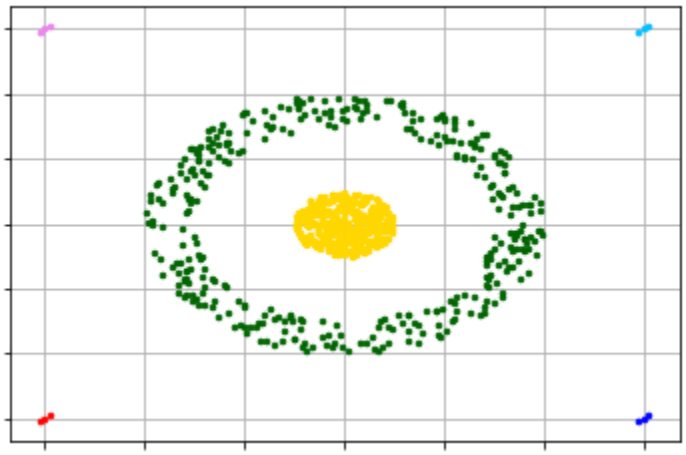
```
from pyclustering.cluster.clique import clique, clique_visualizer
from pyclustering.utils import read_sample
from pyclustering.samples.definitions import FCPS_SAMPLES
# baca two-dimendional data 'Target'
data = read_sample(FCPS_SAMPLES.SAMPLE_TARGET)
# gunakan comand di bawah untuk implementasi algoritma CLIQUE
intervals = 10  #menentukan berapa banyak cells di dalam grid
threshold = 0   # lets consider each point as non-outlier
clique_instance = clique(data, intervals, threshold)
# gunakan command di bawah untuk memulai proses clusterisasi
clique_instance.process()
clusters = clique_instance.get_clusters()  # alokasi cluster
noise = clique_instance.get_noise()     # points yang menentukan outliers
cells = clique_instance.get_cells()     # CLIQUE blocks
print("Amount of clusters:", len(clusters))
# visualisasi data clustering
clique_visualizer.show_grid(cells, data)    # menunjukan proses pembagian cell
clique_visualizer.show_clusters(data, clusters, noise)  # show clustering results
```

✓ 4.3s

Gambar di bawah menunjukan proses pembagian grid



Hasil akhir dari algoritma **CLIQUE**



Model-based clustering

Model-based clustering adalah metode clustering yang bertujuan untuk mengoptimalkan kemiripan antara individu dengan menggunakan pendekatan model probabilistik.

“Model probabilistik itu apa ? ”

Model probabilistik adalah model yang menghasilkan hasil akhir tidak pasti.

Salah satu contoh algoritma yang bekerja dengan metode ini adalah **gaussian mixture model,** sesuai namanya algoritma ini bekerja bekerja dengan implementasi lebih dari 1 algoritma lain seperti **Expectation Maximization (EM)** sebagai penentu centroid setiap cluster dan **Bayes Information Criterion (BIC)** sebagai penentu jumlah cluster yang paling baik untuk data yang akan di cluster.

Contoh codingan Gaussian Mixture Model (GMM):

Langkah pertama adalah import library yang dibutuhkan.

```
from sklearn.mixture import GaussianMixture
from sklearn.datasets import make_blobs
import matplotlib.pyplot as plt
from numpy import random
```

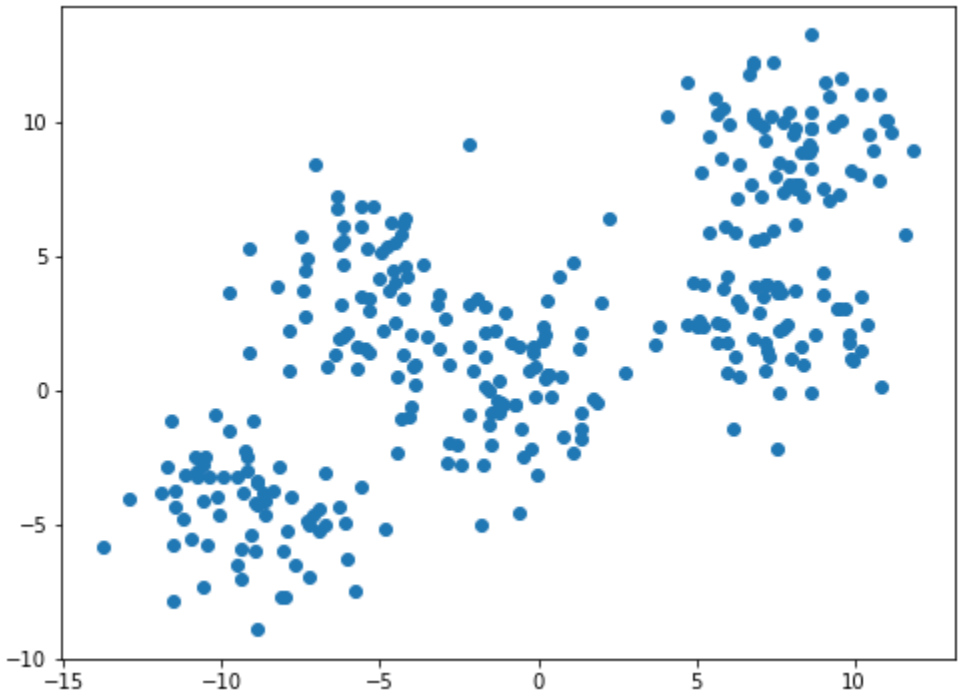
✓ 0.6s Python

Kedua buat random sample dengan “make blobs”.

```
random.seed(234)
x, _ = make_blobs(n_samples=330, centers=5, cluster_std=1.84)
plt.figure(figsize=(8, 6))
plt.scatter(x[:,0], x[:,1])
plt.show()
```

✓ 0.3s Python

Berikut adalah penampakan sampel random yang belum di cluster



Lalu konfigurasi model **GMM** dengan menggunakan library **sklearn** untuk melakukan proses clustering.

```
gm = GaussianMixture(n_components=5).fit(x)
centers = gm.means_

gm.get_params()
```

✓ 0.7s Python

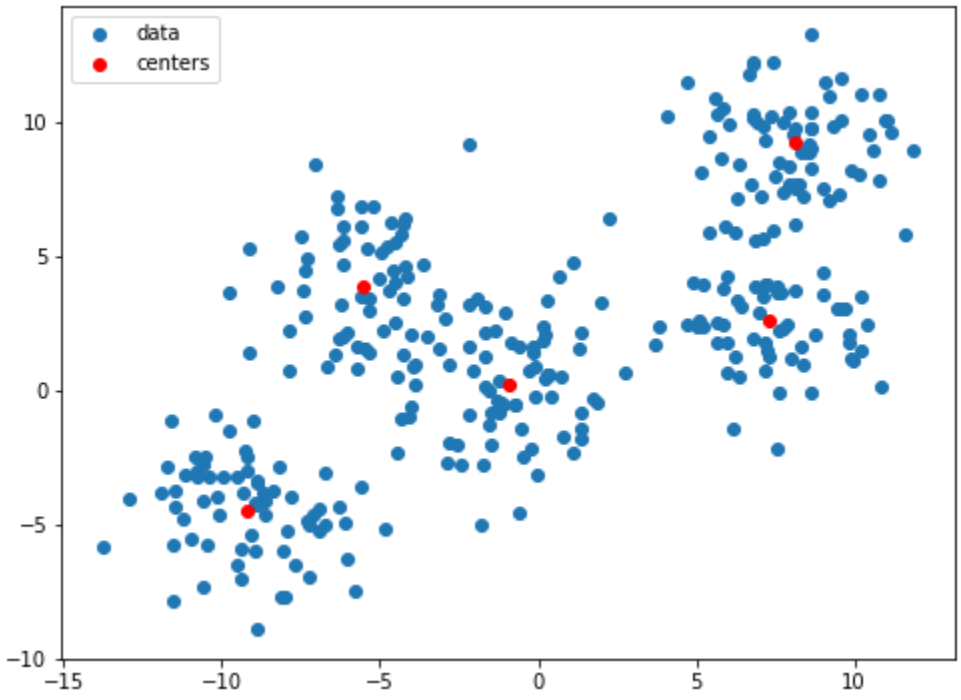
N_component adalah variabel yang dibuat untuk menentukan jumlah cluster.

Setelah selesai, lsayakan konfigurasi data guna memvisualisasi hasil akhir cluster.

```
plt.figure(figsize=(8, 6))
plt.scatter(x[:,0], x[:,1], label="data")
plt.scatter(centers[:,0], centers[:,1],c='r', label="centers")
plt.legend()
plt.show()
```

✓ 0.4s Python

Berikut adalah hasil dari kinerja **gaussian mixture modul**



Project

Setelah membahas metode dan algoritma sekarang saya bakal membahas project proses pembuatan dataset dengan metode **clustering** guna mendapatkan dataset untuk kritik dan saran**.** Jadi di project ini saya bakal memberikan tutorial **clustering** dengan case **sentiment analysis.**

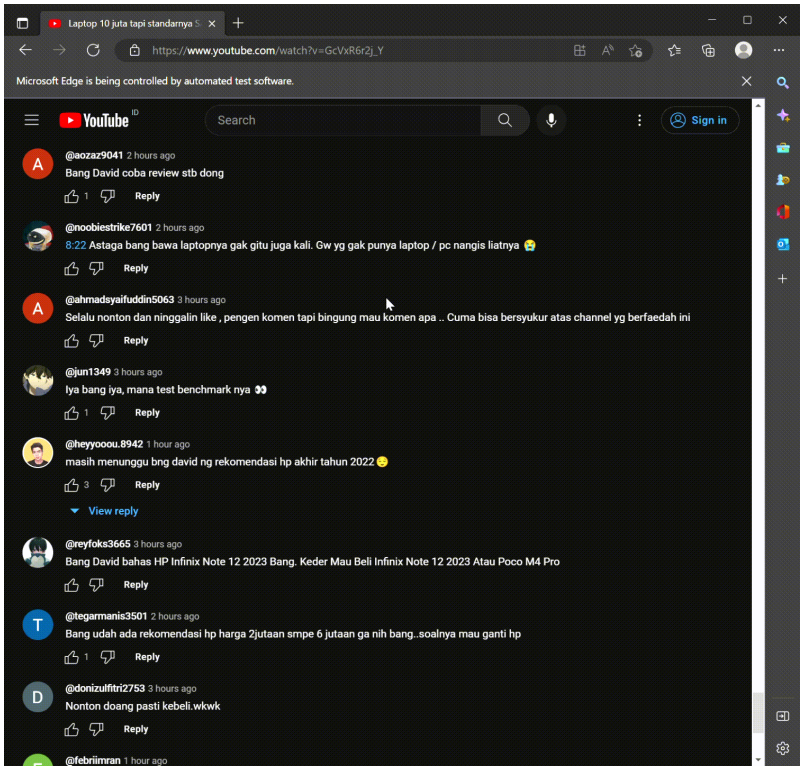
Sentiment analysis adalah proses menganalisis teks digital untuk menentukan arti dan maksud dari sebuah kata dan kalimat.

Berikut langkah langkah yang harus dilakukan.

1. Proses awal dari project ini adalah menentukan platform apa yang akan dijadikan sebagai sumber dari dataset yang akan dibuat. Dalam kasus ini saya pakai platform kolom komentar youtube sebagai sumber dataset.
2. Langkah kedua adalah melakukan **data crawling** dengan cara **scraping** ke web atau platform yang akan di crawling, target yang sudah ditentukan yaitu Youtube.
3. Membuat program untuk scrapping. pembuatan program scrapping harus disesuaikan dengan platform apa yang akan di scrapping.

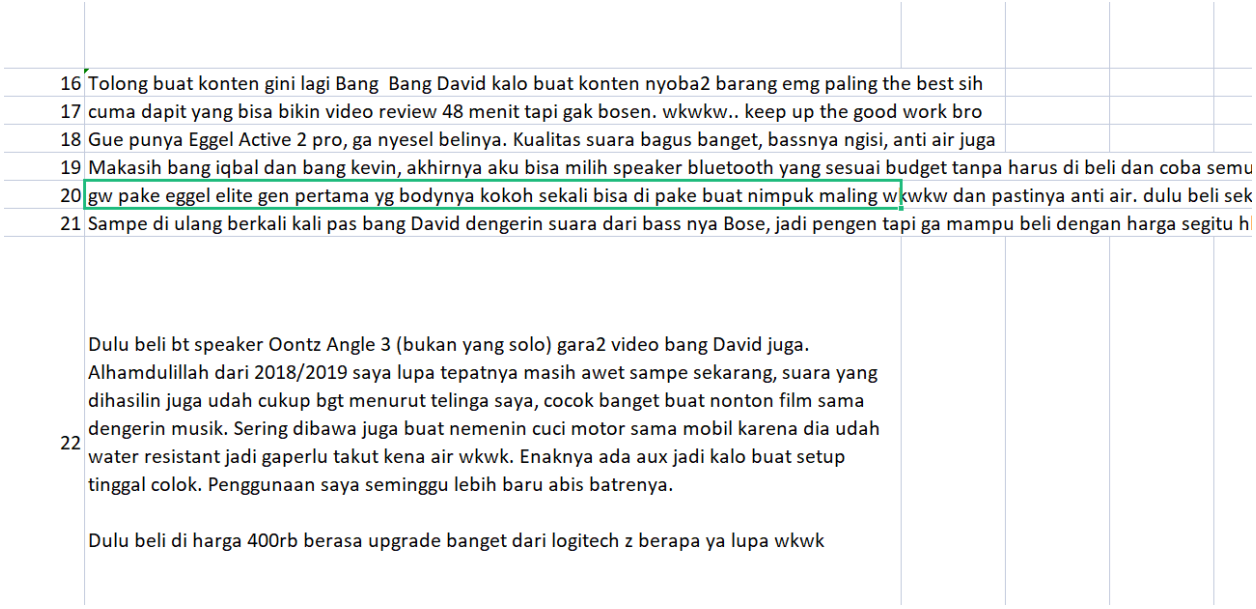
   Di kasus ini saya mengcustom program scrapping yang dibuat, dimana program hanya akan menyaring data dengan ukuran font yang spesifik hanya untuk isi kolom komentar video youtube. Program tidak kubuat dari awal melainkan didapatkan dari platform github dan di modifikasi guna memudahkan proses input link video yang akan kita ambil data komentarnya.

```python
import pandas as pd
from selenium import webdriver
from selenium.webdriver.edge.service import Service as EdgeService
from webdriver_manager.microsoft import EdgeChromiumDriverManager
from bs4 import BeautifulSoup
import time

def  ScrapComment(url):
    option = webdriver.EdgeOptions()
    option.add_argument("--headless")
    driver = webdriver.Edge(service=EdgeService(EdgeChromiumDriverManager().install()))
    driver.get(url)
    prev_h = 0
    while True:
        height = driver.execute_script("""
                function getActualHeight() {
                    return Math.max(
                        Math.max(document.body.scrollHeight, document.documentElement.scrollHeight),
                        Math.max(document.body.offsetHeight, document.documentElement.offsetHeight),
                        Math.max(document.body.clientHeight, document.documentElement.clientHeight)
                    );
                }
                return getActualHeight();
            """)
        driver.execute_script(f"window.scrollTo({prev_h},{prev_h + 200})")
        # kondisikan kecepatan scrolling web dengan kecepatan internet
        time.sleep(0.3)
        prev_h +=200
        if prev_h >= height:
            break
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    driver.quit()
    title_text_div = soup.select_one('#container h1')
    title = title_text_div and title_text_div.text
    comment_div = soup.select("#content #content-text")
    comment_list = [x.text for x in comment_div]
    df = pd.DataFrame(comment_list, columns=['text'])
    df

if __name__ == "__main__":

    a = input("Masukan Link Video Youtube = ")
    print("wait a few minute program need a few minute to start")
    ScrapComment(a)
```



Diatas adalah aktivitas Setelah program di run dan melakukan input link.

Program akan secara otomatis melakukan scrolling dan scrapping semua kolom komentar sampai halaman web tidak bisa di scrolling dan diakhiri dengan menyajikan data hasil scraping dalam bentuk csv.



16	Tolong buat konten gini lagi Bang Bang David kalo buat konten nyoba2 barang emg paling the best sih
17	cuma dapit yang bisa bikin video review 48 menit tapi gak bosen. wkwkw.. keep up the good work bro
18	Gue punya Eggel Active 2 pro, ga nyesel belinya. Kualitas suara bagus banget, bassnya ngisi, anti air juga
19	Makasih bang iqbal dan bang kevin, akhirnya aku bisa milih speaker bluetooth yang sesuai budget tanpa harus di beli dan coba semu
20	gw pake eggel elite gen pertama yg bodynya kokoh sekali bisa di pake buat nimpuk maling wkwkw dan pastinya anti air. dulu beli sek
21	Sampe di ulang berkali kali pas bang David dengerin suara dari bass nya Bose, jadi pengen tapi ga mampu beli dengan harga segitu h
22	Dulu beli bt speaker Oontz Angle 3 (bukan yang solo) gara2 video bang David juga. Alhamdulillah dari 2018/2019 saya lupa tepatnya masih awet sampe sekarang, suara yang dihasilin juga udah cukup bgt menurut telinga saya, cocok banget buat nonton film sama dengerin musik. Sering dibawa juga buat nemenin cuci motor sama mobil karena dia udah water resistant jadi gaperlu takut kena air wkwk. Enaknya ada aux jadi kalo buat setup tinggal colok. Penggunaan saya seminggu lebih baru abis batrenya. Dulu beli di harga 400rb berasa upgrade banget dari logitech z berapa ya lupa wkwk

4. Setelah proses scrapping selesai kita akan lanjut ke proses **clustering**. Sebelum melakukan clustering, hal yang harus dilsayakan terlebih dahulu adalah **data cleaning** dan **case folding**. proses ini adalah proses pembersihan data dari noise noise yang akan mengganggu proses clustering seperti koma, tanda kutip, angka dll dan juga meratakan semua huruf menjadi **lowercase.**

```
def clean(a,b):
    df['text']= df['text'].str.replace(a,b)
df['text'] = df['text'].astype(str).str.lower()
df.text = df.text.str.replace('\d+', '')
clean('.','')
clean(',','')
clean('-','')
clean('/','')
clean(':','')
df
```

Sebelum proses data cleaning dan case folding

Speaker pertama sy mifa f7 ukuranya kecil, enak banget suaranya bass bulet, cma udh drop baterainya.
Skrang pindah ke harman kardon aura 2 udh hmpir 2thun pakai, puas banget.. suaranya jernih, bass manjain kuping.
Detail2 suara di lagu/film yg ga kdengeran di earphone/ speaker murah di aura 2 ini bisa kedengeran. Jadi berasa nonton di bioskop.
Harga emng ga bohong sih, wort it banget pakai dikamar buat beberpa tahun kedepan.

Setelah proses data cleaning dan case folding

speaker pertama sy mifa f ukuranya kecil enak banget suaranya bass bulet cma udh drop baterainyaskrang pindah ke harman kardon aura udh hmpir thun pakai puas banget suaranya jernih bass manjain kupingdetail suara di lagufilm yg ga kdengeran di earphone speaker murah di aura ini bisa kedengeran jadi berasa nonton di bioskop harga emng ga bohong sih wort it banget pakai dikamar buat beberpa tahun kedepan

5. Setelah melakukan proses data cleaning dan case folding, lanjutkan proses ke vectorizer. Vectorizer berfungsi untuk melakukan konversi dari string ke numeric, karena algoritma bekerja dengan cara menghitung ini alasan kenapa proses vectorization di butuhkan.

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer()
x = vectorizer.fit_transform(df['text'].values.astype('U'))
```

```
Output exceeds the size limit. Open the full output data in a text editor
  (0, 594)      0.10288308248245431
  (0, 1683)     0.10718779443322865
  (0, 2167)     0.09450930809450947
  (0, 3680)     0.11325494516535423
  (0, 1295)     0.10288308248245431
  (0, 932)      0.18901861618901894
  (0, 2163)     0.07487449254244594
  (0, 2928)     0.11325494516535423
  (0, 3790)     0.09954408848851501
  (0, 3777)     0.10718779443322865
  (0, 1173)     0.06312887357687928
  (0, 1445)     0.05288884817728097
  (0, 12)       0.15522645567541352
  (0, 1705)     0.0891722258056151
  (0, 2178)     0.10288308248245431
  (0, 946)      0.08037691833530322
  (0, 1714)     0.0831050750734895
  (0, 2788)     0.11429897035522107
  (0, 2895)     0.14001011130480664
  (0, 3145)     0.09681593175032872
  (0, 2175)     0.0891722258056151
  (0, 3694)     0.09681593175032872
  (0, 3665)     0.08123220453734968
  (0, 173)      0.08123220453734968
  (0, 3654)     0.07176749443375455
```

Diatas adalah hasil dari proses penghitungan vectorizer.

6. Setelah proses vectorizer, langsung lanjut ke proses clustering dengan menggunakan library atau package. Di project ini **elbow method** tidak begitu dibutuhkan dikarenakan kita harus bereksperimen sendiri untuk mendapatkan hasil cluster yang baik.

```
from sklearn.cluster import KMeans
true_k = 5
model = KMeans(n_clusters=true_k, init='k-means++', max_iter=1423, n_init=1423)
model.fit(x)
labels=model.labels_
new_df=pd.DataFrame(list(zip(df['text'],labels)),columns=['title','cluster'])
print(new_df.sort_values(by=['cluster']))
```

Untuk proses awal awal saya pakai bakal buat 5 cluster kalau hasil tidak memuaskan jumlah cluster akan terus ditambah sampai setiap cluster memiliki secara optimal memiliki tema yang berbeda dengan cluster lain.

Lalu export hasil clustering dengan **“df.to_csv(‘nama file’.csv)”**

FYI : dalam proses clustering user tidak bisa menentukan tema setiap cluster, jadi user harus mengoptimalkan konfigurasi algoritma dan terus bereksperimen agar hasil akhir clustering memuaskan.

7. Sekarang kita bisa langsung lihat hasil dari proses clusteringnya
   Cluster 1 berisikan saran untuk video

coba unboksing yg merk im bang kecil barang nya tp cabe rawitbentuk nya bulet segede tutup galon air mineral	1
mending rakit sendiri bang buat boombox	1
bang david ijin request next konten bikin rekomendasi setup box tv digital bang harga dibawah k kalo bisa mayoritas warga kampung masih pake tv	1
coba bang borong set box tv buat referensi beli karna tv analog mau ganti jadi digital biar ga salah beli jadi harus dicoba dulu sama bang david	1
coba compare merk mifan bang produknya tidak pernah mengecewakan	1
next coba review headset gaming dibawah ribu yang paling laris di toko online bang	1
biji itu bongkar semua bang karena kabel dll didalamnya ngaruh ke tahanannya	1
bang di jawa timur ada lomba sound system gitu coba jadiin content bang kayanya seru	1
bang lu harus buat lagu sendiri menurut gua biar pas lagi test audio gini bisa muterin lagu ncr lagi	1
tolong battlein gimbal brica dan dji yah bang david	1
review speaker jbl original bang lain kali kalo lagi gabut	1
karna bang davit itu speaker eggle sold out astaga wkwk	1
harusnya lagu koplo bang biar bisa dinilai bass nya	1
di tunggu acome dan egle nya d jual bang	1
entah kenapa gue selalu suka dan nontonin konten gabut lo bang wkwkwkkw	1
bang coba riviuw polytron pma dan sejenisnya bang dari dulu mau nyari spiker sekelas itu tapi bingung	1

Cluster 2 sama seperti cluster 1, berisi saran konten tapi dengan pola kata kata yang berbeda.

coba bluetooth speaker high end dong bang biar tau worth apa ngga hehe	2
review headphone bluetooth borongan kayak gini dong bang saya lebih tertarik sama headphone daripada tws ato speaker	2
bang bandingkan spiker bluetooth jbl charge sama charge dong	2
next video rekomendasi blender mini terbaik dong min untuk anak kosan nih	2
request review microphone buat karaoke dibawah rb dong bang sekalian juga review speaker aktif buat karaoke rate harga dibawah jt	2
mau nanya dong seputar speaker kalau advance tbt itu gimana ya kualitas nya? subwoofer seharga rb an	2
bang david review smartphone baru dr nothing phone dong bang soalnya design nya unik jadi penasaran	2
bang cobain advance vsbt dong menurut saya lumayan bagus itu	2
bangbikin konten rakit pc jutaan buat tahun ini dong	2
bang davidreview proyektor portabel dong yg bajet jt kebawah	2
next bahas keyboard dong bang membran mechanical	2
gadgetin review speaker seperti phantom boombox atau bose selevel dong	2

Cluster 3 berisikan review penonton terhadap produk dalam video.

overall lebih rekomended acome a atau eggel fit ya bang?
pasukan team eggel
team eggel saya pake terra + dan benar benar mengejutkan
saya dengerin video ini pakai eggel fit the best lah diharga ribuan latency nya bagus buat nonton yutub atau film cocok
speaker eggel sama simbadda bagus mana ya?
udah nebak si eggel fit kek punya gw gw dirumah bandingin ama jbl temen gw dan hasilnya bass dari eggel fit lebih kerasa tapi claryty hampir mirip
saya nonton pake speaker anker soundcore motion+ yg baru kebeli hari yg lalu rela jual eggel terra + demi nambahin fitur audio hi•res & aptx
eggel terbaik pokoknya barang saya semua dari eggel
eggel active plus suara ok battery kurang dipake baru lagu volume full dah minta di carger
eggel tera harganya luar biasa untuk suaranya
bang bikin konten kaya gni lagi ya soalnya keabisan eggel fit '))
gw nnton ini pakai ipad air + eggel active mantaap pokokna
eggel bagus batre tahan lam di rumah sya punya complit bbrapa kali jatoh suaranya masih bagus
saya nonton pake eggel fit
ada speaker bluetooth eggel rupanya dirumah ada biji sih eggel terra + enak ada fitur tws nyadibawah jt merk eggel salah satu yg rekomended sih
wah kurang eggel tera series dan active series bang itu juara juaranya

Cluster 4 berisi noise dan data yang tidak terpakai.

hehhh daviiddd eggel fit langsung naik harganya gara gara looo
eh eh yang kalian cari wkwk
buat yg nyari eh eh doang
sound yg lagi viral eh eh
sound eh eh rame di tiktok bg wkwk
eh eh ful
sound eh eh
nyari "eh eh"?
cuma liat sound eh eh
eh eh
eh ehh
polosan david eh eh
heh eh

Cluster 5 berisi pertanyaan penonton seputar konten video youtube.

speaker apa yg cocok buat muter lagu "sikok bagi duo" ngab?
judul lagu waktu test apa bang share dong hehehe
judul instrumental buat test speaker apa bang david
tes soundnya pakai lagu apa aja itu bang?
info lagu yang di pake bang david judulnya apaan
itu yg buat tes pake lagu apa sih bang?
lagu buat ngetest yang acoustic cewe nyayi itu judul nya apaan ya ??
judul lagu yang nomer itu apa ya?yang i wanna be
judul lagi yg di pake bg david itu apa ya?

Cluster 6 berisikan review penonton terhadap produk yang di review, berupa cerita pribadi konsumen..

acome a ku juga udah tahan tahunan dan emang bagus sih suaranya ngalahin mi pocket speaker yg pernah gw beli	6
speaker pertama sy mifa f ukuranya kecil enak banget suaranya bass bulet cma udh drop baterainyaskrang pindah ke harman kardon aura udh hmpir thun pakai puas banget suaranya jernih	6
acome a yang di review ini gue udah pake hampir tahun kalo buat pemakaian harian kayak gue yg juga wfh dan naro speaker bluetooth di meja buat hiburan musik kala kerja sangat worth i	6
fit emang bagus banget harga kualitas semuanya sebagai guru yang butuh speaker buat audio listening harga murah ringkas ini salah satu pilihan terbaik	6
pas banget lagi cari speaker bluetooth murah buat bertani	6
seru banget nge test speaker nya ikut senang liat koh david senang wkwkwk	6
wkwk gasalah beli gua thn yg lalu pilih acome alarm jam soalnya dulu pengen punya speaker yg sekalian ada fitur jamnya buat pajngan juga sempet bingung banget pertimbanginnya karna b	6
wah kebetulan banget nih saya lagi nyari speaker bluetooth jdi dri video ini bisa jdi referensi buat saya thanks banget bang david	6
saya termasuk orang yang sering atau bahkan setiap hari pake speaker bluetooth di ponsel saya menurut saya robot bagus suaranya (bulet dan bass) speaker satu lagi juga bagus yang ada m	6
kalo untuk simbada sih blutut speaker nya emang kurang nendang tapi kalo simbada yang subwoper boleh di adu hahah	6
wah kebetulan banget nih! gw lagi nyari speaker bluetooth yang murah tapi lumayan enak separasi audionya lanjutkan bang david	6
kak ad speaker bluetooth yg saling connect ga si kak? yang kecil harga jg terjangkau	6
bt speaker bagus murah banyak tapi durabilitas yg mahal percaya deh jbl flip sama mifa a suaranya sama sama mantep sama sama beli di jbl masih utuh suara masih mantap mifa a modul	6
kebetulan banget lagi nyari review bluetooth speaker yang ada jamnya ternyata di review juga siap harga naik nih	6
speaker simbadda nya harus di letakin di meja bang kalau di angkat gitu emang kurang berasa bass nya karna subwoofer nya ngadap bawah saya punya soalnya simbadda itu cuman yaa untu	6
akhirnyaaa udah dari lama cari bluetooth speaker selain jebeel yg ga nguras kantong rasanya kurang yakin kalo belum di review sama gadgetin	6
pas banget dari kemarin bingung mau cari bluetooth speaker	6
menurut saya speaker bluetooth terbaik merk gmc bsuara stereo dan bass nya nendangfitur oke bangetlah	6
kontennya kocak bang david mulai dari speaker belah sampe ada yang desainnya jiplek wkwkwkwk	6
awal maret aku beli acome a di salah satu aplikasi marketplace aku beli ini bukan karena speakernya malah tapi lebih ke jam alarmnya suaranya sih enak dan lantang malah saat dibawa nge	6
senang nya speaker yg ada lampu led sama bass yg mantab	6

Cluster 7 berisikan verifikasi penonton terhadap originalitas barang yang di Review.

jbl go nya udah % saya pastikan kw itu om david jadi gak fair kalau dijadikan acuan toko yang jual jbl go ditokopedia itu udah terkenal jual barang kw dengan dalih kualitas oem jbl go tidak	7
speaker jbl terlihat tidak meyakinkan semoga bukan barang kw bang soalnya punya jbl go yg pertama (bentuk kotak) lebih enak suaranya beli rb oktober	7
itu jbl aslinya rb sekarang harga yg ori jbl go rb hati meek jbl banyak yg palsu dan yg di review adalah barang palsunya aslinya suaranya lembut banget	7
bang david and team untuk jbl go nya sepertinya itu yang versi kw saya sudah beberapa tahun terakhir memakai yang jbl go dan nada openingnya nggak seperti itu (hanya satu nada blub aja	7
agak keget sih pas denger suara jbl go tapi pas denger harga k kemungkinan kw tuh bangyg ori kalo gk salah harga skitar ngomong" aku pakai yg go dulu beli rb dan suaraya menurutku lebih	7
jbl go rb mantep pakek banget! suaranya jernih walau volume sangat keras	7
yang jbl kayanya kw deh jbl go harganya setauku an dan yang di video pas masih di dalem bungkus masa posisinya kebalik wkwk	7
jbl go masih menemani di kamar kostsudah tahun	7
masih punya jbl go versi awal dan masih awet sampai sekarang tapi sekarang punya yang go dan suara bassnya mantap!	7
sedang menggunakan jbl kw harga rb yang suaranya ga kalah sama yang harga rban	7
yg jbl go itu yakin kawe saya klik linknya harga baru diskon rb dan nyantumin gambarnya gadgetin saat review ini di kolom diskusi sellernya gak tegas jawab ori atau enggaksaya cek jbl go o	7
user jbl go pertama nyimak masih bagus dan awet sampai sekarang	7
kw itu bang harga di official storenya jbl sekitar klo ga salah	7
yang jbl go agak diragukan originalitasnya yaa soalnya saya punya yg jbl go generasi pertama aja harganya diatas itu dan suaranya bagus	7
btw untuk jbl nya kw karna tombol jbl go itu tidak timbul sedangkan di video tombolnya terlihat timbul thanks	7
jbl itu di official storenya yg go bukannya an yaa bang?	7
jbl go nya sepertinya kw di official store jbl'nya saja harganya masih rb'an	7
jbl go harga dibawah k fix kw yang ori di official storenya ribuan	7

Conclusion

“Setelah semua penjelasan dan project diatas, Mana metode dan algoritma clustering yang paling sempurna untuk dipakai ?”

Jawabannya, tidak ada. Karena setiap metode dan algoritma mempunyai keuntungan dan kelebihan di setiap kasus.

Kamu bisa melakukan *research* lebih lanjut dan terus ber-eksperimen untuk menemukan metode dan algoritma yang suitable dengan kasus mu.

Tutorial penggunaan algoritma yang saya berikan tidak bisa berlaku untuk semua kasus karena setiap kasus juga punya proses dan konfigurasi yang berbeda.

Untuk file codingan yang tersedia di dalam modul ini bisa dilihat di : [madhiemw/Clustering-Algorithm-Module (github.com)]

Referensi

CLIQUE Algorithm in Data Mining - GeeksforGeeks

Clustering Menggunakan Algoritma DBSCAN (Density-Based Spatial Clustering of Applications with Noise) untuk Data Hasil Produksi Potensi Pertanian Studi kasus : Kabupaten Gresik (ugm.ac.id)

What is the difference between K-Means and DBSCAN? (tutorialspoint.com)

DBSCAN Clustering (algotech.netlify.app)

sklearn.cluster.AgglomerativeClustering — scikit-learn 1.1.3 documentation

2.3. Clustering — scikit-learn 1.1.3 documentation

2.3. Clustering — scikit-learn 1.1.3 documentation

DBSCAN Algorithm Clustering in Python | Engineering Education (EngEd) Program | Section

K-Medoids/Partitioning Around Medoids (PAM) — Non Hierarchical Clustering with R | by Tri Binty N. | Medium

panagiotisanagnostou/HiPart: Hierarchical divisive clustering algorithm execution, visualization and Interactive visualization. (github.com)

Text Clustering (devopedia.org)

5 Clustering Methods in Machine Learning | Clustering Applications (analyticssteps.com)

repository.fe.unj.ac.id/3303/5/Chapter3.pdf

Understanding OPTICS and Implementation with Python | by Yufeng | Towards Data Science

Penerapan metode penggerombolan berdasarkan gaussian mixture models dengan menggunakan algoritma expectation maximization (ipb.ac.id)

Elbow Method vs Silhouette Score - Which is Better? - Data Analytics (vitalflux.com)