Informatika

Innovative & Collaborative

Clustering Data Pengaduan Masyarakat Menggunakan HDBSCAN dan K-Means pada Dataset NYC 311

Kelompok 7

MEMBER

Siti Fadhilah Rahmi
(2311532003)

Kaela Assyura Syadira
(2311531001)

Muhammad Raudhatul
(2311532013)

Fayi Amatullah Azhara
(2311537001)

DATASET

Dataset NY 311 Service Requests merupakan data terbuka yang mencatat seluruh permintaan layanan dan pengaduan non-darurat dari masyarakat di New York City melalui layanan NYC 311. Dataset ini mencakup informasi seperti waktu laporan, jenis keluhan, instansi penanggung jawab, status penanganan, serta lokasi kejadian. Data ini digunakan untuk meningkatkan transparansi, memantau kinerja layanan publik, dan menganalisis pola permasalahan perkotaan, sehingga sangat bermanfaat untuk analisis data, visualisasi, dan pengambilan keputusan berbasis data.

DATASET

```python
import pandas as pd
import heapq

file_path = "311_data/311-service-requests-from-2010-to-present.csv"
chunksize = 200000


top_heap = []
counter = 0

print("Memulai pemrosesan streaming...")

for chunk in pd.read_csv(file_path, chunksize=chunksize, parse_dates=["Created Date"], low_memory=False):


    chunk = chunk.dropna(subset=["Created Date"])


    chunk = chunk.sort_values("Created Date", ascending=False)


    chunk_records = chunk.to_dict("records")

    for row in chunk_records:
        created = row["Created Date"]


        tuple_item = (created, counter, row)
        counter += 1

        if len(top_heap) < 10000:
            heapq.heappush(top_heap, tuple_item)
        else:

            if created > top_heap[0][0]:
                heapq.heapreplace(top_heap, tuple_item)
            else:
                break

print("Mengubah heap jadi DataFrame...")
final_rows = [item[2] for item in top_heap]
df_final = pd.DataFrame(final_rows)

print("Sort final berdasarkan Created Date terbaru...")
df_final = df_final.sort_values("Created Date", ascending=False)

print("Menyimpan ke CSV...")
df_final.to_csv("311_latest_10k.csv", index=False)

print("Selesai! File: 311_latest_10k.csv")
```

Dataset NY 311 Service Requests memiliki skala yang sangat besar dengan jumlah data mencapai jutaan baris dan ukuran file sekitar ±16 GB, sehingga kurang efisien jika digunakan secara keseluruhan dalam satu proyek analisis. Oleh karena itu, pada proyek ini dilakukan cropping data dengan mengambil 10.000 baris data terbaru sebagai sampel utama yang digunakan. Subset ini terdiri dari 46 kolom yang mencakup informasi kunci seperti identitas laporan, waktu pembuatan dan penyelesaian laporan, jenis dan deskripsi keluhan, instansi penanggung jawab, status penanganan, serta detail lokasi kejadian termasuk alamat, borough, dan koordinat geografis. Dengan pembatasan data ini, proses eksplorasi, pembersihan, dan analisis dapat dilakukan secara lebih efisien tanpa mengurangi konteks dan kualitas insight yang dihasilkan.

DATASET

```
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 46 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Unique Key                      10000 non-null  int64
 1   Created Date                    10000 non-null  object
 2   Closed Date                     5199 non-null   object
 3   Agency                          10000 non-null  object
 4   Agency Name                     10000 non-null  object
 5   Complaint Type                  10000 non-null  object
 6   Descriptor                      9871 non-null   object
 7   Location Type                   7592 non-null   object
 8   Incident Zip                    9868 non-null   float64
 9   Incident Address                9725 non-null   object
 10  Street Name                     9725 non-null   object
 11  Cross Street 1                  6691 non-null   object
 12  Cross Street 2                  6682 non-null   object
 13  Intersection Street 1           5981 non-null   object
 14  Intersection Street 2           5978 non-null   object
 15  Address Type                    4228 non-null   object
 16  City                            9486 non-null   object
 17  Landmark                        5361 non-null   object
 18  Facility Type                   148 non-null    object
 19  Status                          10000 non-null  object
 20  Due Date                        0 non-null      float64
 21  Resolution Description          7760 non-null   object
 22  Resolution Action Updated Date  7646 non-null   object
 23  Community Board                 10000 non-null  object
 24  BBL                             9084 non-null   float64
 25  Borough                         10000 non-null  object
 26  X Coordinate (State Plane)      9830 non-null   float64
 27  Y Coordinate (State Plane)      9830 non-null   float64
 28  Open Data Channel Type          10000 non-null  object
 29  Park Facility Name              9988 non-null   object
 30  Park Borough                    10000 non-null  object
 31  Vehicle Type                    4 non-null      object
 32  Taxi Company Borough            10 non-null     object
 33  Taxi Pick Up Location           159 non-null    object
 34  Bridge Highway Name             1 non-null      object
 35  Bridge Highway Direction        5 non-null      object
 36  Road Ramp                       4 non-null      object
 37  Bridge Highway Segment          4 non-null      object
 38  Latitude                        9830 non-null   float64
 39  Longitude                       9830 non-null   float64
 40  Location                        9830 non-null   object
 41  Zip Codes                       5296 non-null   float64
 42  Community Districts             5306 non-null   float64
 43  Borough Boundaries              5306 non-null   float64
 44  City Council Districts          5306 non-null   float64
 45  Police Precincts                5306 non-null   float64
dtypes: float64(12), int64(1), object(33)
```

Subset dataset NY 311 Service Requests yang digunakan dalam proyek ini terdiri dari 10.000 entri dengan 46 kolom, yang merepresentasikan berbagai aspek laporan layanan publik di New York City. Struktur data mencakup kombinasi 33 kolom bertipe kategorikal (object), 12 kolom numerik bertipe float, dan 1 kolom numerik bertipe integer, sehingga memungkinkan analisis deskriptif maupun spasial. Kolom-kolom utama memuat informasi inti seperti identitas laporan, waktu pembuatan dan penutupan laporan, jenis dan deskripsi keluhan, instansi penanggung jawab, serta status penanganan. Selain itu, tersedia pula detail lokasi yang cukup lengkap, mulai dari alamat, borough, hingga koordinat geografis (latitude dan longitude). Beberapa kolom memiliki nilai kosong (missing values), terutama pada atribut opsional seperti fasilitas, detail persimpangan jalan, dan informasi kendaraan, yang mencerminkan karakteristik data dunia nyata dan perlu diperhatikan dalam tahap pembersihan data sebelum analisis lanjutan.

Preprocessing

Load Data

```
import pandas as pd
df = pd.read_csv('/content/311_latest_10k.csv', engine='python', on_bad_lines='skip')

print("First 5 rows of the DataFrame:")
print(df.head())

print("\nDataFrame Information:")
df.info()
```

```
[5 rows x 46 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
```

```
   Unique Key         Created Date Closed Date Agency  \
0    45050101  2019-12-01 02:04:01         NaN    DOT
1    45054936  2019-12-01 01:59:41         NaN   NYPD
2    45049329  2019-12-01 01:59:08         NaN   NYPD
3    45052046  2019-12-01 01:58:23         NaN   NYPD
4    45054999  2019-12-01 01:58:07         NaN   NYPD

                      Agency Name       Complaint Type  \
0     Department of Transportation     Street Condition
1  New York City Police Department   Noise - Commercial
2  New York City Police Department  Noise - Residential
3  New York City Police Department  Noise - Residential
4  New York City Police Department      Illegal Parking

                   Descriptor             Location Type  Incident Zip  \
0                     Pothole                       NaN       10001.0
1           Loud Music/Party         Club/Bar/Restaurant       11223.0
2           Loud Music/Party  Residential Building/House       11207.0
3           Loud Music/Party  Residential Building/House       11358.0
4  Commercial Overnight Parking            Street/Sidewalk       11426.0

     Incident Address  ... Road Ramp Bridge Highway Segment   Latitude  \
0    WEST   29 STREET  ...                      NaN  40.745668
1        178 AVENUE U  ...                      NaN  40.596475
2  807 SCHENCK AVENUE  ...                      NaN  40.660605
3    42-41 159 STREET  ...                      NaN  40.759994
4    88-49 237 STREET  ...                      NaN  40.729499
```

```
   Longitude                                          Location Zip Codes  \
0 -73.987719  {'longitude': '-73.9877188309367', 'latitude':...       NaN
1 -73.977721  {'longitude': '-73.97772147626671', 'latitude'...       NaN
2 -73.883508  {'longitude': '-73.8835082736363', 'latitude':...       NaN
3 -73.806856  {'longitude': '-73.80685560533585', 'latitude'...       NaN
4 -73.729998  {'longitude': '-73.72999847973517', 'latitude'...       NaN

  Community Districts Borough Boundaries City Council Districts  \
0                 NaN                 NaN                   NaN
1                 NaN                 NaN                   NaN
2                 NaN                 NaN                   NaN
3                 NaN                 NaN                   NaN
4                 NaN                 NaN                   NaN

  Police Precincts
0              NaN
1              NaN
2              NaN
3              NaN
4              NaN
```

```
 45  Police Precincts                          5306
dtypes: float64(12), int64(1), object(33)
memory usage: 3.5+ MB
None
```

Data Loading: Dataset 311_latest_10k.csv berhasil dimuat.
Struktur Awal:

- Total Data: 10.000 baris (entries).
- Total Kolom: 46 kolom.
- Temuan Awal: Terdapat banyak nilai kosong (null) pada kolom seperti Closed Date, Facility Type, dan Latitude/Longitude.

Handling Missing Values

```
print(f"Baris sebelum drop lokasi kosong: {len(df)}")
df = df.dropna(subset=[col for col in location_cols if col in df.columns])
print(f"Baris setelah drop lokasi kosong: {len(df)}")

#Kolom ID Wilayah & Koordinat Sekunder
#Kolom ini lebih ke 'label' wilayah. Kalau tidak tahu, isi 0 (sebagai tanda 'Unknown').
other_numerical_cols = [
    'BBL', 'X Coordinate (State Plane)', 'Y Coordinate (State Plane)',
    'Zip Codes', 'Community Districts', 'Borough Boundaries',
    'City Council Districts', 'Police Precincts'
]

for col in other_numerical_cols:
    if col in df.columns:
        df[col] = df[col].fillna(0)

print("\nSisa missing values di kolom numerik (seharusnya 0):")
all_num_cols = [c for c in location_cols + other_numerical_cols if c in df.columns]
print(df[all_num_cols].isnull().sum())
```

```
Baris sebelum drop lokasi kosong: 10000
Baris setelah drop lokasi kosong: 9830

Sisa missing values di kolom numerik (seharusnya 0):
Incident Zip                0
Latitude                    0
Longitude                   0
BBL                         0
X Coordinate (State Plane)  0
Y Coordinate (State Plane)  0
Zip Codes                   0
Community Districts         0
Borough Boundaries          0
City Council Districts      0
Police Precincts            0
dtype: int64
```

- Penghapusan Kolom: Menghapus 9 kolom yang memiliki data kosong >90% (contoh: Facility Type, Due Date, Vehicle Type).
- Penyaringan Baris: Menghapus baris yang tidak memiliki informasi lokasi krusial (Latitude, Longitude, Incident Zip).
- Hasil Akhir Cleaning:
  - Jumlah baris berkurang dari 10.000 menjadi 9.830 baris.
  - Data yang tersisa dijamin memiliki koordinat lokasi yang lengkap.

Data Type Conversion & Filling

```python
# 1. MUAT ULANG DATAFRAME dan KONVERSI TIPE DATA
try:
    df = pd.read_csv('311_prepocessed.csv')

    # Konversi kolom tanggal kembali ke tipe datetime
    df['Created Date'] = pd.to_datetime(df['Created Date'])
    df['Closed Date'] = pd.to_datetime(df['Closed Date'])
    df['Resolution Action Updated Date'] = pd.to_datetime(df['Resolution Action Updated Date'])

    # Konversi kolom 'Complaint Duration' kembali ke timedelta
    df['Complaint Duration'] = pd.to_timedelta(df['Complaint Duration'])

    print("✅ DataFrame 'df' berhasil dimuat ulang dan tipe data dikonversi.")
```

```python
# Hitung panjang (jumlah karakter) dari 'Resolution Description'
temp_desc = df['Resolution Description'].replace('Unknown', None)
df['Resolution Description Length'] = temp_desc.str.len().fillna(0).astype(int)

# Hitung panjang (jumlah karakter) dari 'Complaint Type'
df['Complaint Type Length'] = df['Complaint Type'].str.len().fillna(0).astype(int)
```

```python
# Hitung Durasi Keluhan dalam Jam (dari timedelta)
df['Complaint Duration Hours'] = df['Complaint Duration'].dt.total_seconds() / 3600
```

Konversi Tanggal: Kolom Created Date dan Closed Date diubah dari teks menjadi format datetime.
Pengisian Nilai Kosong (Imputasi):

- Kolom Kategorikal (misal Descriptor, City) diisi dengan label 'Unknown'.
- Kolom Numerik area (misal BBL, Zip Codes) diisi dengan angka 0.

Hasil: Tidak ada lagi missing values pada dataset (kecuali kolom tanggal penutupan untuk kasus yang masih aktif).

Feature Engineering

```
✅ DataFrame 'df' berhasil dimuat ulang dan tipe data dikonversi.

--- 1. Ekstraksi Fitur Temporal ---
First 5 rows of new temporal features:
   Created Day of Week  Created Month  Complaint Duration Hours
0                    6             12                       NaN
1                    6             12                       NaN
2                    6             12                       NaN
3                    6             12                       NaN
4                    6             12                       NaN

--- 2. Normalisasi Fitur Lokasi ---
Borough unik setelah normalisasi: ['MANHATTAN' 'BROOKLYN' 'QUEENS' 'BRONX' 'STATEN ISLAND']

--- 3. Ekstraksi Fitur Teks Sederhana ---
First 5 rows of new text features:
   Resolution Description Length  Complaint Type Length
0                            104                     16
1                              0                     18
2                              0                     19
3                              0                     19
4                              0                     15

✅ DataFrame akhir berhasil disimpan ke '311_prepocessed_and_featured.csv'.
```

Tujuan: Menciptakan variabel baru yang lebih informatif untuk model machine learning.

Fitur Temporal (Waktu):

- Created Day of Week: Mengubah tanggal menjadi angka hari (0=Senin, 6=Minggu).
- Created Month: Mengambil bulan untuk melihat pola musiman.
- Complaint Duration Hours: Menghitung selisih waktu (Closed - Created) dalam satuan jam.

Fitur Teks & Lokasi:

- Borough: Normalisasi teks menjadi huruf kapital (standarisasi).
- Description Length: Menghitung panjang keluhan (jumlah karakter) untuk melihat seberapa detail pelapor.

Standardization

```python
from sklearn.preprocessing import StandardScaler

# Handle NaN values in 'Complaint Duration Hours' with the median
median_complaint_duration_hours = df['Complaint Duration Hours'].median()
df['Complaint Duration Hours'].fillna(median_complaint_duration_hours, inplace=True)

print(f"Missing values in 'Complaint Duration Hours' after median imputation: {df['Complaint Duration Hours'].isnull().sum()}")

# Initialize StandardScaler
scaler = StandardScaler()

# Apply StandardScaler to the selected numerical columns
df[numerical_cols_to_standardize] = scaler.fit_transform(df[numerical_cols_to_standardize])

print("\nFirst 5 rows of standardized numerical columns:")
print(df[numerical_cols_to_standardize].head())

print("\nDescriptive statistics of standardized numerical columns:")
print(df[numerical_cols_to_standardize].describe())
```

Imputasi Data:

- Output: Missing values in 'Complaint Duration Hours' after median imputation: 0.
- Nilai kosong pada durasi keluhan telah diisi dengan nilai tengah (median) agar tidak merusak proses perhitungan.

Tampilan Data Terstandarisasi (head):

- Kolom seperti Incident Zip, Latitude, Created Hour kini memiliki nilai berskala kecil (contoh: -1.47, 0.78).
- Tidak ada lagi angka ribuan atau jutaan yang mendominasi.

Verifikasi Statistik (describe):

- Mean (Rata-rata): Semua kolom mendekati 0 (terlihat angka seperti 1.57e-15).
- Std (Standar Deviasi): Semua kolom bernilai 1.0.
- Ini mengonfirmasi bahwa data sudah terdistribusi normal standar.

Model K-Means

```
n_clusters = 10
kmeans = KMeans(n_clusters=n_clusters, init='k-means++', random_state=42)
```

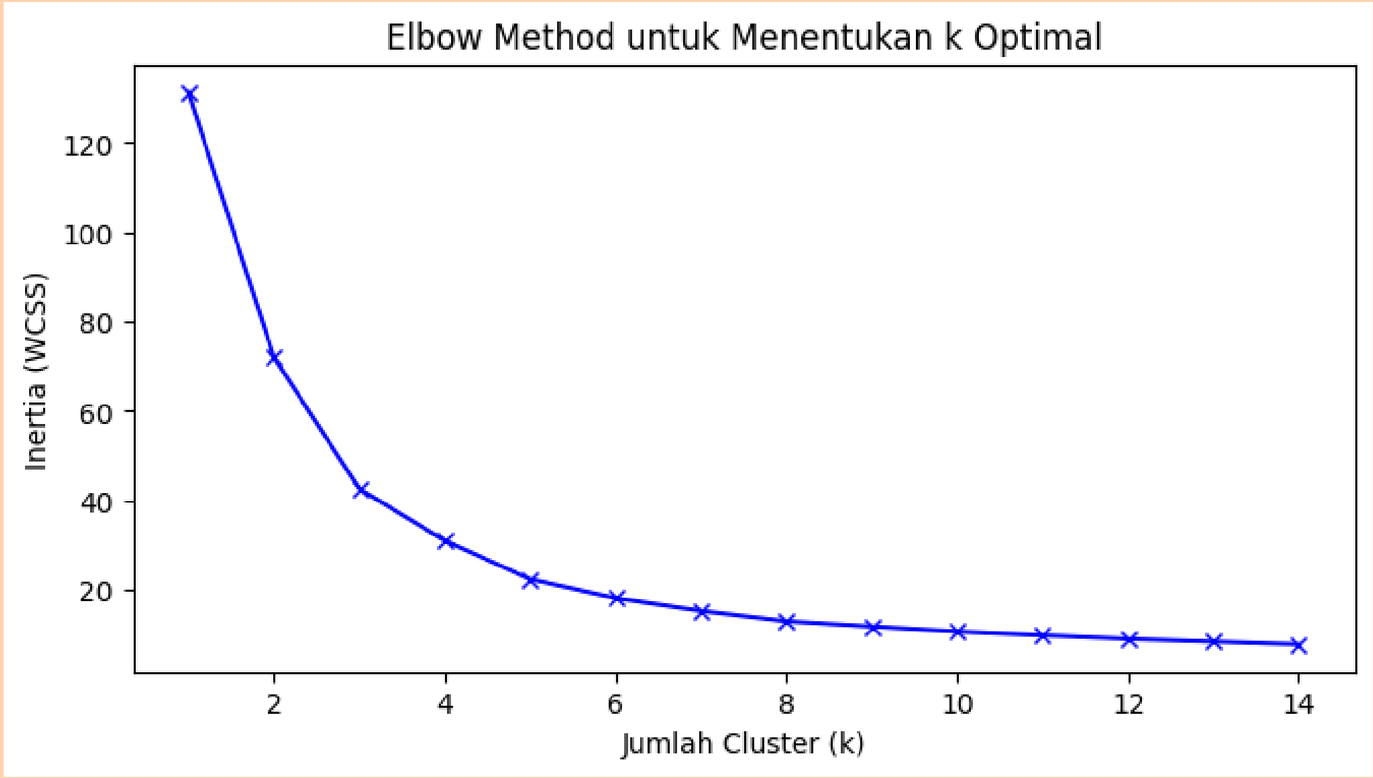
- n_clusters = 10, artinya membagi data menjadi 10 kelompok (wilayah) berbeda
- init='k-means++' artinya untuk memilih lokasi awal pusat kelompok (centroid) secara berjauhan. Tujuannya agar proses pengelompokan lebih cepat akurat dan tidak terjebak pada hasil yang buruk.
- random_state = 42
- K-Means sangat populer untuk aplikasi web karena kecepatannya dan hasilnya yang pasti (setiap keluhan akan masuk ke dalam satu wilayah/cluster tertentu, tanpa ada yang dianggap sebagai noise) karena K-Means akan memaksa setiap titik masuk ke dalam kelompok, tidak ada data yang dibuang sebagai "noise".

Model K-Means

```
df['cluster_kmeans'] = kmeans.fit_predict(coords)

centroids = kmeans.cluster_centers_
print("Pusat Cluster (Latitude, Longitude):")
print(centroids)
```

```
Pusat Cluster (Latitude, Longitude): [
[ 40.84978699 -73.91258425]
[ 40.67548063 -73.92088262]
[ 40.58907049 -74.13926398]
[ 40.72858462 -73.98287594]
[ 40.74329579 -73.88097734]
[ 40.67284106 -73.80626142]
[ 40.85774341 -73.8632636 ]
[ 40.62339606 -73.97681683]
[ 40.79801219 -73.94778125]
[ 40.73673117 -73.78081669]]
```

Hasil Grafik Elbow

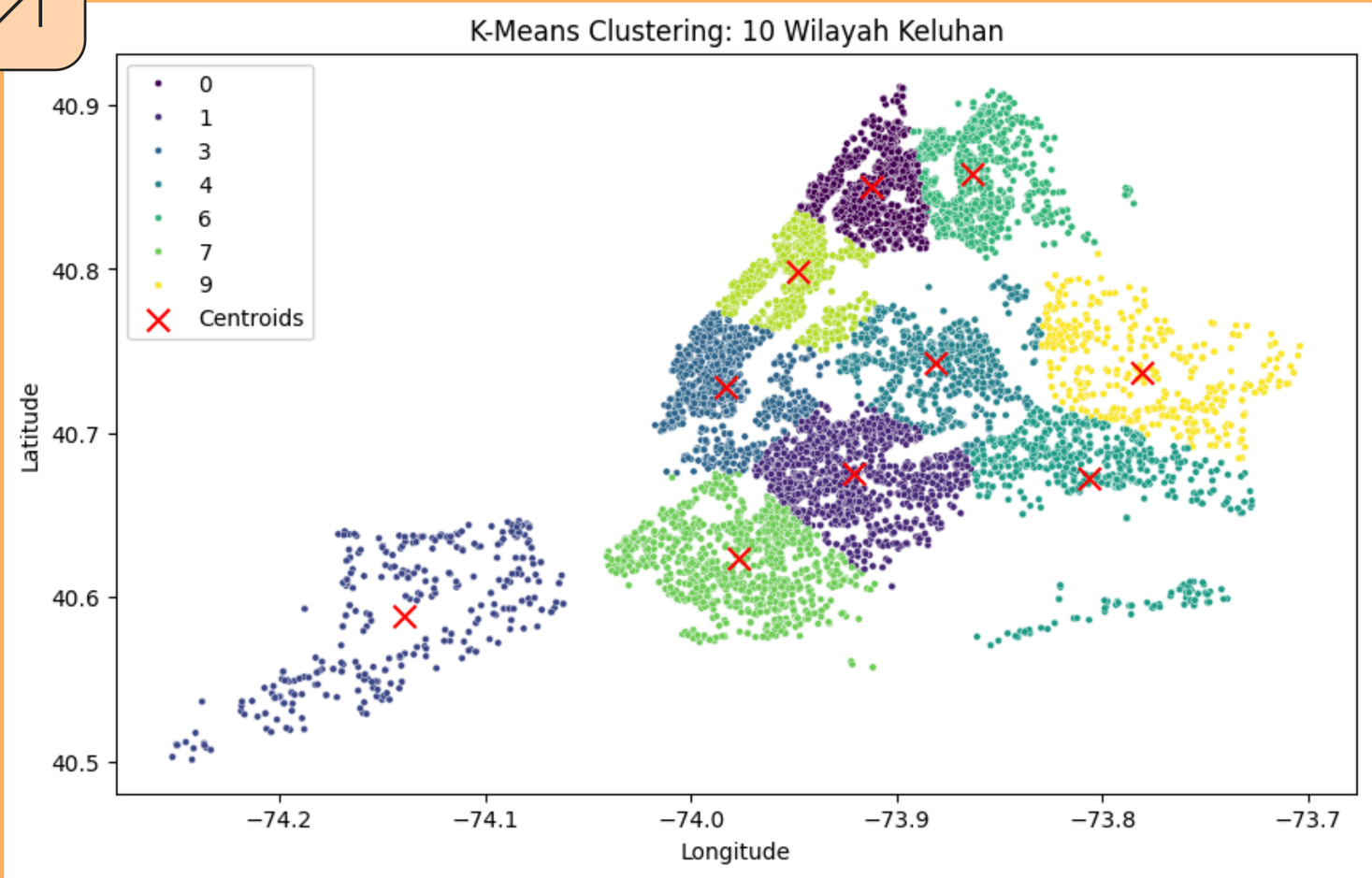


Model K-Means

```
print("\nJumlah keluhan per cluster:")
print(df['cluster_kmeans'].value_counts().sort_index())
```

Jumlah keluhan per cluster:

- 0 = 1597
- 1 = 1388
- 2 = 327
- 3 = 1129
- 4 = 772
- 5 = 809
- 6 = 1055
- 7 = 1241
- 8 = 1026
- 9 = 486



Hasil Clustering K-Means

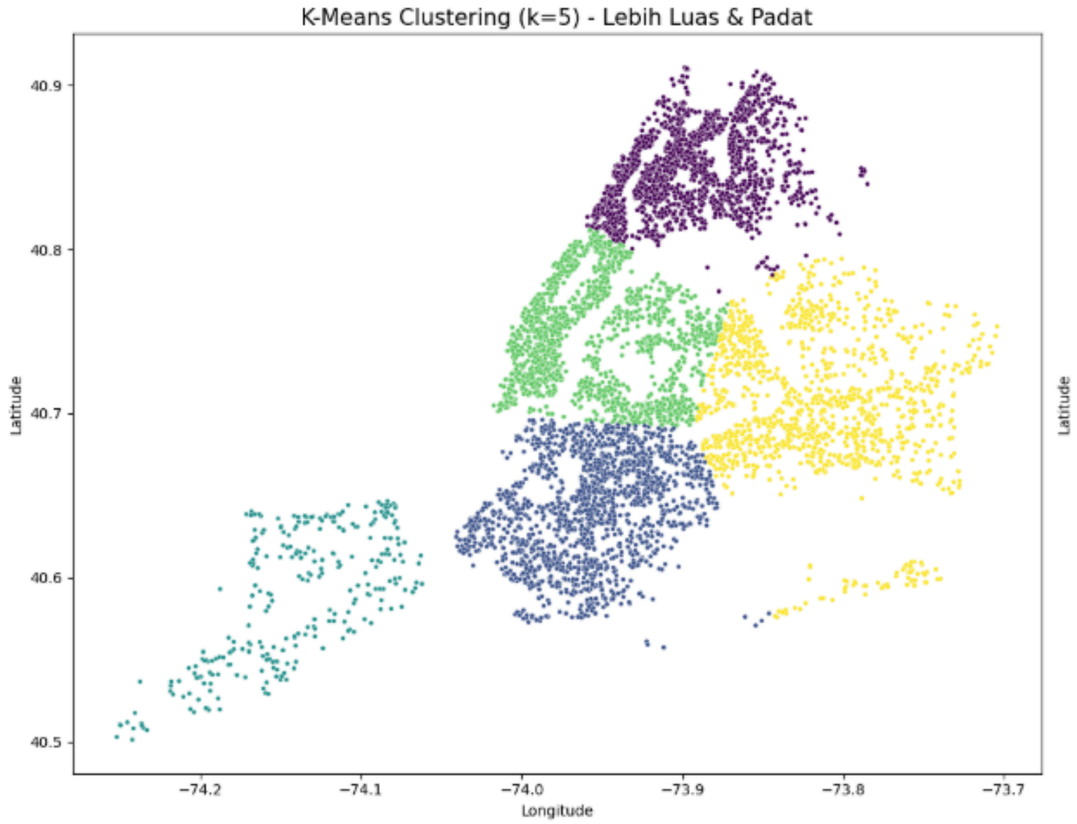
3 Keluhan Teratas dari Semua Cluster

3 Keluhan Terbanyak di Tiap Wilayah (Cluster):

	cluster_kmeans	Complaint Type	count
23	0	HEAT/HOT WATER	642
36	0	Noise - Residential	458
28	0	Illegal Parking	77
89	1	HEAT/HOT WATER	341
104	1	Noise - Residential	257
64	1	Blocked Driveway	159
153	2	Illegal Parking	46
161	2	Noise - Residential	40
158	2	Missed Collection (All Materials)	37
222	3	Noise - Residential	173
206	3	HEAT/HOT WATER	133
210	3	Illegal Parking	66
291	4	Noise - Residential	135
254	4	Blocked Driveway	128
280	4	Illegal Parking	128
368	5	Sewer	158
325	5	Blocked Driveway	135
348	5	Illegal Parking	91
407	6	HEAT/HOT WATER	335
419	6	Noise - Residential	256
387	6	Blocked Driveway	115
476	7	HEAT/HOT WATER	256
480	7	Illegal Parking	190
491	7	Noise - Residential	175
562	8	Noise - Residential	274
544	8	HEAT/HOT WATER	270
548	8	Illegal Parking	49
589	9	Blocked Driveway	73
615	9	Noise - Residential	68
606	9	Illegal Parking	67

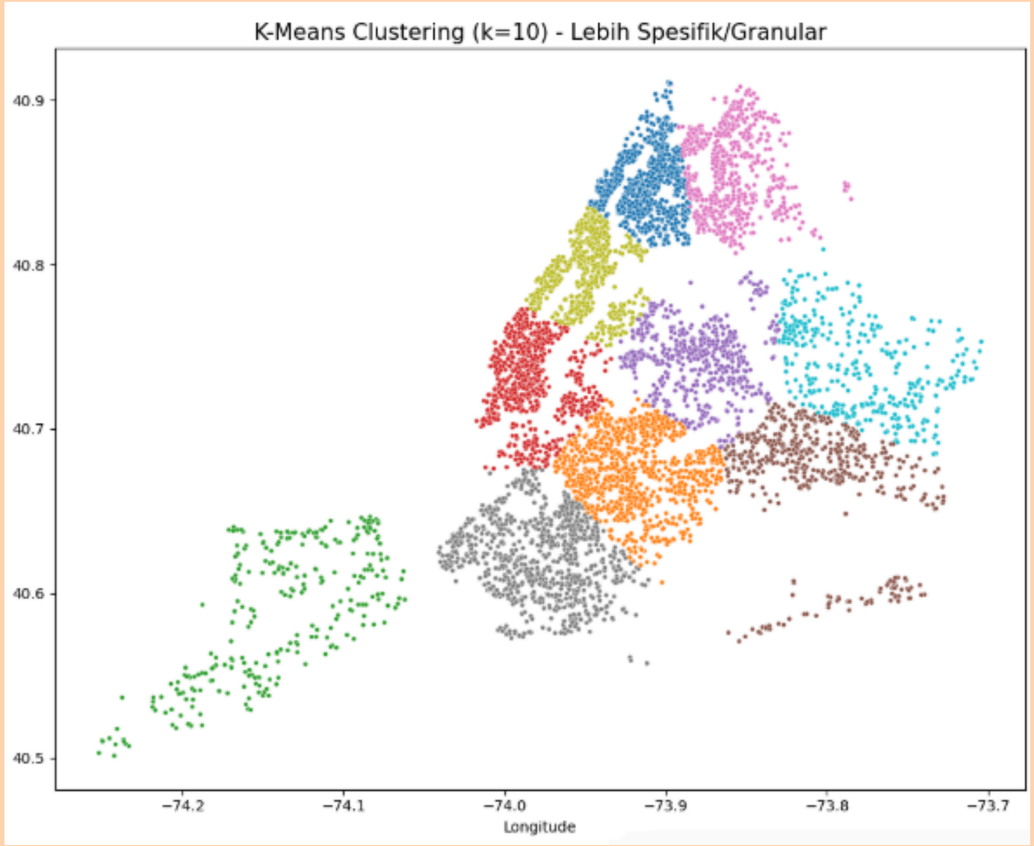
Hasil Silhouette Score

Silhouette Score k=5 : 0.4661



Hasil Silhouette Score

Silhouette Score k=10: 0.3909



Pemilihan K-Means

Model K-Means Clustering dipilih karena

- Kemampuannya membagi wilayah secara cepat dan tegas sehingga sangat responsif saat digunakan di aplikasi Streamlit.
- Model ini memastikan setiap keluhan warga terpetakan ke dalam zona operasional tertentu tanpa ada data yang terbuang.
- Dengan hasil Elbow Method yang stabil dan skor Silhouette mencapai 0.4661, K-Means terbukti efektif dalam mengelompokkan lokasi keluhan
- Mempermudah identifikasi masalah dominan di setiap wilayah.

Content Strategy

- Social Media Optimization
- Search Engine Optimization
- Paid Advertising Optimization
- Marketing using Electronic Mail

To captivate potential customers, we shall provide them with engaging and practical materials such as informative articles, user-friendly infographics, and captivating video content.

Target & Projection

Increased Brand Awareness

Target

Develop and enhance brand recognition among pertinent and prospective clientele.

Projection

In the first 3 months, we estimate brand awareness will increase by 30% from your product's website and social media traffic.

Increased Site Visitor

Target

Generate greater recognition for your brand among pertinent and prospective clientele.

Projection

In the first three months, we think that traffic to your product's website and social media pages will raise brand recognition by 30%.

Model Clustering HDBSCAN

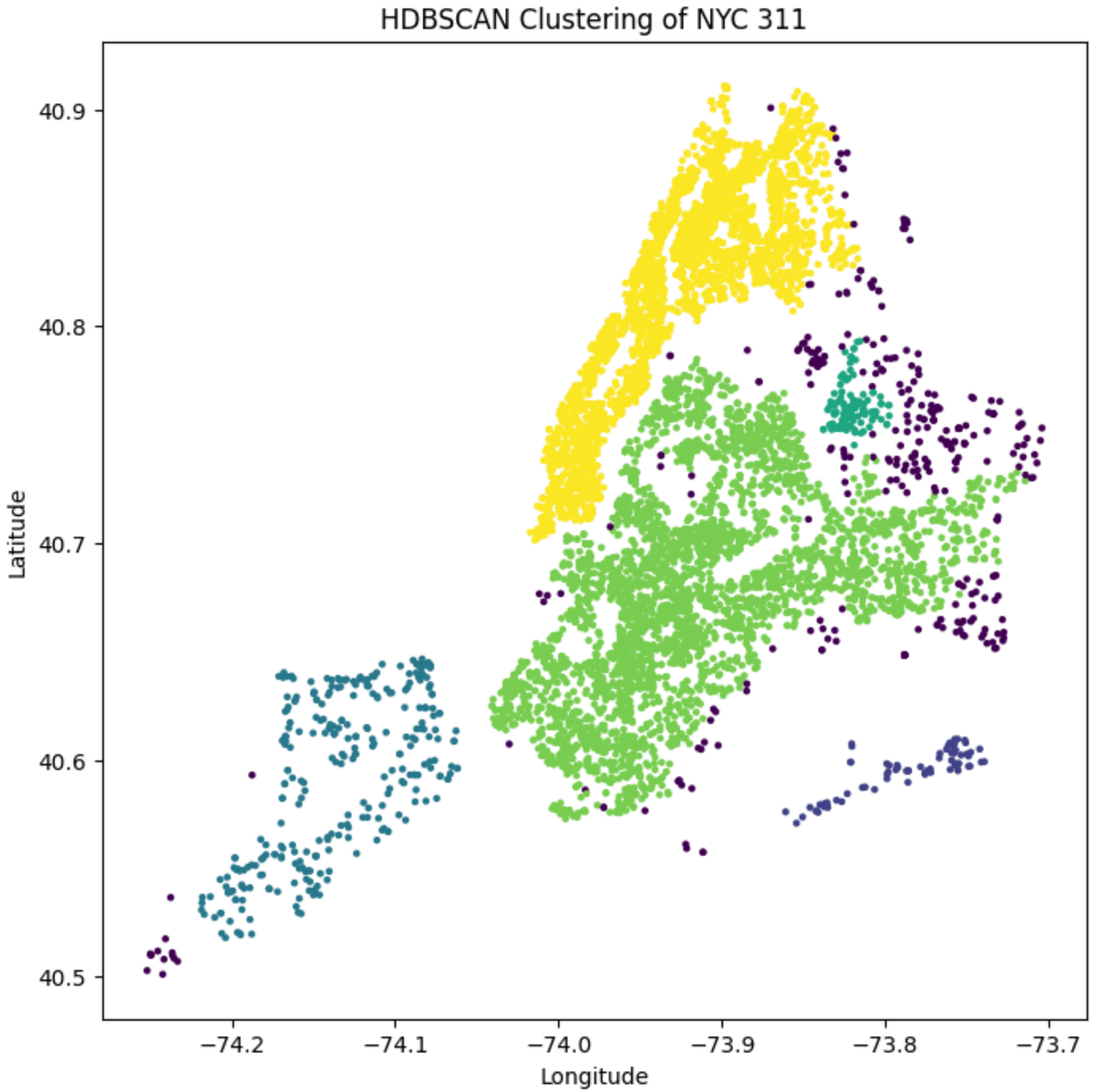
Parameter yang digunakan:

- min_cluster_size = 70
- min_samples = 10
- metric = euclidean
- cluster_selection_method = 'eom'
- HDBSCAN dipilih karena:
  - Tidak perlu menentukan jumlah cluster di awal.
  - Mampu menangani noise dan kepadatan data yang tidak seragam.

```
import hdbscan

clusterer = hdbscan.HDBSCAN(
    min_cluster_size=70,
    min_samples=10,
    metric='euclidean',
    cluster_selection_method='eom'
)

df['cluster'] = clusterer.fit_predict(coords)

df['cluster'].value_counts()

df.to_csv("df_with_clusters.csv", index=False)
```

Visualisasi Hasil Cluster

```
====== HDBSCAN Cluster Evaluation ======
Silhouette Score          : 0.2409
Davies-Bouldin Index      : 0.7456
Calinski-Harabasz Score : 3343.6057
```



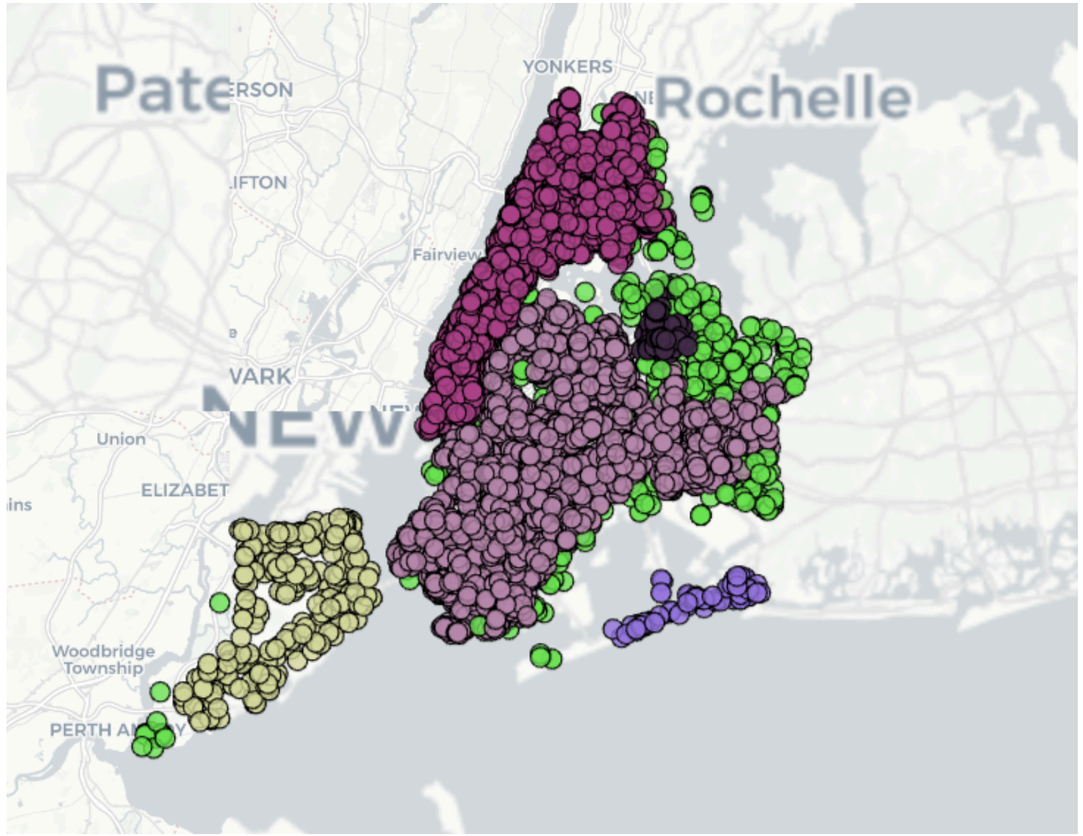
Pengaduan Dominan

- Data dikelompokkan berdasarkan cluster hasil HDBSCAN.
- Untuk setiap cluster, diambil 5 jenis pengaduan terbanyak (Top-5 Complaint Type).
- Tujuan: Memahami karakteristik utama setiap cluster.

cluster	Complaint Type	
-1	Illegal Parking	47
	Noise - Residential	32
	Blocked Driveway	26
	Water System	20
	Abandoned Vehicle	18
0	Illegal Parking	15
	HEAT/HOT WATER	13
	Noise - Residential	11
	Blocked Driveway	9
	Sewer	4
1	Illegal Parking	42
	Noise - Residential	39
	Missed Collection (All Materials)	33
	Blocked Driveway	20
	Abandoned Vehicle	16
2	Blocked Driveway	29
	Noise - Residential	21
	HEAT/HOT WATER	14
	Abandoned Vehicle	7
	Illegal Parking	7
3	HEAT/HOT WATER	845
	Noise - Residential	756
	Blocked Driveway	592
	Illegal Parking	581
	Sewer	191
4	HEAT/HOT WATER	1303
	Noise - Residential	1058
	Blocked Driveway	201
	Illegal Parking	199
	Noise - Street/Sidewalk	129

Name: count, dtype: int64

Peta Interaktif Sebaran Cluster




```
=== Cluster -1 - Borough Distribution ===
Borough
QUEENS           257
BRONX             32
BROOKLYN          25
STATEN ISLAND     14
MANHATTAN          2
Name: count, dtype: int64

=== Cluster 0 - Borough Distribution ===
Borough
QUEENS    80
Name: count, dtype: int64

=== Cluster 1 - Borough Distribution ===
Borough
STATEN ISLAND    313
Name: count, dtype: int64

=== Cluster 2 - Borough Distribution ===
Borough
QUEENS    122
Name: count, dtype: int64

=== Cluster 3 - Borough Distribution ===
Borough
BROOKLYN    2770
QUEENS      1897
Name: count, dtype: int64

=== Cluster 4 - Borough Distribution ===
Borough
BRONX        2250
MANHATTAN    2068
Name: count, dtype: int64
```

Prediksi Jenis Pengaduan

- Data yang memiliki nilai Descriptor dan Complaint Type dipilih.
- Fokus pada 20 jenis pengaduan paling sering muncul.
- Tujuan:
  - Mengurangi ketidakseimbangan kelas.
  - Meningkatkan stabilitas model klasifikasi.

- Menggunakan pipeline:
- TF-IDF Vectorizer
- Maksimum 5000 fitur
- Logistic Regression dengan max_iter = 2000 untuk konvergensi optimal
- Model ini dipilih karena:
  - Efisien untuk data teks skala besar.
  - Mudah diinterpretasikan.
  - Cocok untuk baseline klasifikasi teks.

```
=== Complaint Type Prediction Report ===
                                precision    recall  f1-score   support

               Abandoned Vehicle       1.00      1.00      1.00        34
               Blocked Driveway        1.00      1.00      1.00       176
             Broken Parking Meter      0.95      1.00      0.98        20
    General Construction/Plumbing      1.00      0.88      0.93        24
                  HEAT/HOT WATER       1.00      1.00      1.00       442
                 Illegal Parking       0.99      0.99      0.99       178
Missed Collection (All Materials)      1.00      1.00      1.00        22
                           Noise       1.00      1.00      1.00        28
             Noise - Commercial        0.00      0.00      0.00        47
             Noise - Helicopter        0.83      1.00      0.91        15
             Noise - Residential       0.82      1.00      0.90       384
         Noise - Street/Sidewalk       0.00      0.00      0.00        39
             Noise - Vehicle           0.96      1.00      0.98        23
                    PLUMBING           1.00      1.00      1.00        19
                      Rodent           1.00      1.00      1.00        16
                       Sewer           1.00      1.00      1.00        45
           Sidewalk Condition          1.00      0.90      0.95        30
           Street Condition            0.89      1.00      0.94        24
         UNSANITARY CONDITION          1.00      0.92      0.96        26
                Water System           1.00      1.00      1.00        36

                    accuracy                               0.94      1628
                   macro avg           0.87      0.88      0.88      1628
                weighted avg           0.90      0.94      0.92      1628
```

Prediksi Borough

- Data yang memiliki Latitude, Longitude, dan Borough dipilih.
- Borough dikonversi menjadi nilai numerik menggunakan Label Encoding.
- Encoding diperlukan agar data dapat diproses oleh model machine learning.

- Model KNN Berbasis Jarak Geografis
- Menggunakan K-Nearest Neighbors (KNN).
- Metrik jarak:
  - Haversine Distance, sesuai untuk data koordinat bumi.
  - Koordinat diubah ke satuan radian sebelum pelatihan.

```
===== Cluster Model Evaluation =====
0.9951179820992677
              precision    recall  f1-score   support

          -1       0.95      0.90      0.93        83
           0       1.00      1.00      1.00        20
           1       1.00      1.00      1.00        78
           2       0.94      1.00      0.97        30
           3       1.00      1.00      1.00      1167
           4       1.00      1.00      1.00      1080

    accuracy                           1.00      2458
   macro avg       0.98      0.98      0.98      2458
weighted avg       1.00      1.00      1.00      2458
```

Thank You