Jurnal Sistem Informasi Bisnis (JUNSIBI)
OPEN ACCES
p-ISSN: 2774-3446
e-ISSN: 2774-3454
diterbitkan oleh:
Program Studi Sistem Informasi
Institut Bisnis dan Informatika (IBI) Kosgoro 1957
Vol. 6, No. 2, Oktober 2025, pp. 247-257
DOI: https://doi.org/10.55122/junsibi.v6i2.1684

KLASTERISASI LAGU PADA PLATFORM SPOTIFY BERDASARKAN FITUR AUDIO MENGGUNAKAN ALGORITMA K-MEANS DAN K-MEANS++

PENULIS

[1)]Ramadanti Indah Safitri, [2)]Sari Ningsih

ABSTRAK

Perkembangan teknologi telah mengubah cara pengguna menikmati musik, dengan platform seperti Spotify menyediakan jutaan lagu dari berbagai genre. Namun, dengan jumlah lagu yang sangat besar, pengelompokan lagu berdasarkan karakteristik audio menjadi tantangan tersendiri. Penelitian ini bertujuan untuk mengelompokkan lagu-lagu di Spotify berdasarkan fitur audio menggunakan algoritma K-Means dan K-Means++. Dataset yang digunakan mencakup atribut seperti danceability, energy, valence, tempo, loudness, speechiness, instrumentalness, acousticness, dan liveness. Metode klasterisasi K-Means dan K-Means++ diterapkan untuk membagi lagu ke dalam tiga klaster utama, berdasarkan hasil Elbow Method. Lalu kluster dilabeli dengan Energik, Mellow, dan Ceria berdasarkan analisis hasil rata-rata pada tiap cluster. Evaluasi menggunakan Silhouette Score menunjukkan bahwa K-Means++ menghasilkan klasterisasi yang lebih optimal dibandingkan K-Means, dengan nilai 0.32246467825287023. Klaster Energik menjadi kelompok terbesar (56.5%), diikuti oleh Mellow (24.3%) dan Ceria (19.2%). Visualisasi menunjukkan pemisahan klaster yang cukup jelas, meskipun terdapat sedikit beririsan antara klaster Ceria dan Energik.

Kata Kunci: Klasterisasi Lagu, Fitur Audio, Spotify, K-Means, K-Means++, Perbandingan

AFILIASI

Program Studi: [1)]Informatika, Fakultas Teknologi Komunikasi dan Informatika
[2)]Sistem Informasi, Fakultas Teknologi Komunikasi dan Informatika
Nama Institusi: [1,2)]Universitas Nasional
Alamat Institusi: [1,2)]Jl. Sawo Manila No.61, Pejaten, Pasar Minggu, Jakarta Selatan, DKI Jakarta

KORESPONDENSI

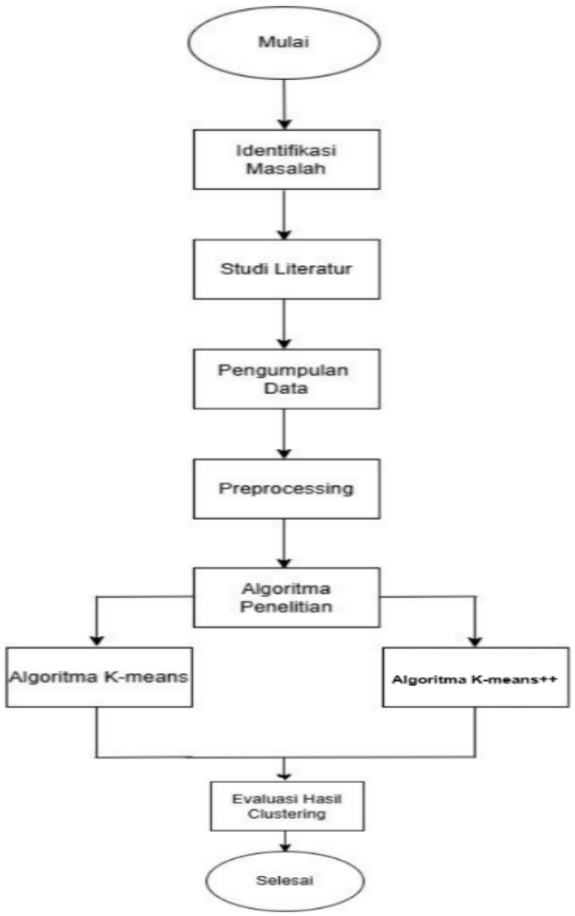
Penulis: Sari Ningsih
Email: lectures.sariningsih@gmail.com

I. PENDAHULUAN

Spotify telah menjadi salah satu platform streaming musik terbesar di dunia dengan lebih dari 500 juta pengguna aktif, memberikan akses ke jutaan lagu dari berbagai genre. Kemudahan akses terhadap musik ini membawa tantangan dalam hal pengelompokan dan rekomendasi lagu yang sesuai dengan preferensi pengguna [2]. Salah satu cara untuk mengatasi tantangan ini adalah dengan memanfaatkan fitur audio sebagai parameter utama dalam analisis musik. Studi sebelumnya menunjukkan bahwa pengelompokan lagu berdasarkan fitur audio dapat menghasilkan klaster yang mencerminkan suasana hati tertentu, seperti bahagia, sedih, atau santai. Algoritma K-Means sering digunakan dalam analisis data musik karena kesederhanaan dan efektivitasnya dalam mengelompokkan lagu berdasarkan kesamaan fitur audio [4][5]. Namun, kelemahan utama algoritma ini terletak pada pemilihan pusat klaster yang dilakukan secara acak, yang dapat menyebabkan hasil klasterisasi yang tidak stabil. Untuk mengatasi permasalahan ini, K-Means++ diperkenalkan dengan proses inisialisasi pusat klaster yang lebih baik, memungkinkan konvergensi yang lebih cepat dan hasil klasterisasi yang lebih optimal [3][6]. Dengan semakin banyaknya data musik yang tersedia di platform streaming seperti Spotify, penerapan metode machine learning dalam analisis musik menjadi semakin penting. Spotify menyediakan API yang memungkinkan peneliti untuk mengakses berbagai atribut audio yang relevan untuk klasterisasi lagu. Hasil klasterisasi ini tidak hanya membantu pengguna dalam menemukan lagu yang sesuai dengan selera mereka tetapi juga dapat digunakan oleh Spotify untuk mengembangkan sistem rekomendasi yang lebih canggih. Dengan memahami pola dalam data audio dan interaksi pengguna terhadap lagu-lagu tertentu, sistem rekomendasi yang dihasilkan dapat lebih personal dan relevan [1][7][8][9].

II. METODE PENELITIAN

Penelitian ini dilakukan dengan menggunakan dataset lagu dari Spotify yang mencakup fitur audio seperti *danceability, energy, valence, tempo, loudness, speechiness, instrumentalness, acousticness,* dan *liveness.* Gambar 1 menggambarkan tahapan penelitian.



Gambar 1. Tahapan Penelitian

2.1 Identifikasi Masalah

Salah satu kendala utama adalah kurangnya pemahaman tentang fitur audio yang paling berpengaruh dalam menarik minat pengguna. Fitur seperti *danceability, energy, valence, loudness, tempo, speechiness, acouticness, instrumentalness, dan liveness* berperan penting dalam menentukan apakah lagu disukai pendengar. Penelitian ini diharapkan dapat menyempurnakan algoritma rekomendasi agar lebih sesuai dengan preferensi pengguna.

2.2 Studi Literatur

Tahapan studi literatur dalam penelitian ini memanfaatkan referensi dari penelitian sebelumnya sebagai landasan teori serta sebagai bahan rujukan untuk mempermudah penyusunan laporan penelitian. Jurnal-jurnal yang dijadikan acuan pada penelitian ini memiliki rentang waktu antara tahun 2019 hingga 2024.

2.3 Pengumpulan Data

Proses pengumpulan data dilakukan melalui web scraping dan akses ke Spotify Web API. Tahapan awal meliputi pembuatan akun, login di developer.spotify.com, serta pembuatan dashboard untuk memperoleh client ID dan client secret guna mengakses data. Data yang dikumpulkan mencakup fitur audio lagu di Spotify untuk analisis lebih lanjut.

Dataset penelitian ini berisi 2.972 entri dengan 16 kolom, termasuk fitur seperti danceability, energy, valence, tempo, loudness, speechiness, instrumentalness, acousticness, dan liveness. Detail dataset dapat dilihat pada Gambar 4.1 dan informasi lebih lanjut tersedia di Tabel 1.

Tabel 1. Atribut Data Spotify

Fitur	Informasi
name	*Name* adalah judul lagu dari data yang diambil.
album	*Album* adalah nama album tempat lagu tersebut dirilis.
artist	*Artist* adalah nama penyanyi atau band yang membawakan lagu tersebut.
release_date	*Release date* adalah tanggal ketika lagu dirilis.
length	*Length* adalah durasi lagu dalam satuan waktu (detik).
popularity	*Popularity* adalah ukuran popularitas sebuah lagu.
danceability	*Danceability* mengukur kecocokan sebuah lagu untuk ditarikan berdasarkan tempo, ritme, dan keteraturannya.Fitur ini memiliki rentan nilai dari 0.0 (tidak cocok untuk ditarikan) hingga 1.0 (cocok untuk ditarikan).
energy	*Energy* merupakan pengukuran tingkat intensitas dan aktivitas suatu lagu, di mana lagu dengan energi tinggi cenderung terdengar cepat, keras, dan ramai. Fitur ini memiliki rentan nilai dari 0.0 (kurang enerjik) hingga 1.0 (enerjik).
valence	*Valence* adalah pengukur tingkat kepositifan dan kenegatifan sebuah lagu. Fitur ini memiliki rentan nilai dari 0.0 (sedih, tertekan, marah) hingga 1.0 (gembira, ceria, euforia).
tempo	*Tempo* adalah mengacu pada tingkat kecepatan keseluruhan lagu yang diukur dalam satuan ketukan per menit (BPM). Tempo tidak memiliki rentan nilai karena bervariasi, tergantung lagu.
loudness	*Loudness* adalah keseluruhan track dalam desibel (dB). Fitur ini memiliki rentan nilai dari -60 hingga 0.
speechiness	*Speechiness* merupakan pengukuran jumlah kata yang diucapkan dalam suatu lagu. Fitur ini memiliki rentan nilai dari 0.33 hingga 0.66. Semakin tinggi nilai, semakin banyak kata yang diucapkan dalam lagu.
instrumentalness	*Instrumentalness* adalah prediksi yang digunakan untuk mengidentifikasi apakah suatu lagu memiliki vokal atau tidak. Fitur ini memiliki rentan nilai dari 0 hingga 1. Semakin tinggi nilai, menujukkan besar kemungkinan lagu tidak memiliki konten vokal.
acousticness	*Acousticness* adalah pengukur apakah sebuah lagu adalah lagu akustik atau bukan. Fitur ini memiliki rentan nilai dari 0.0 hingga 1.0. Semakin tinggi nilai menujukkan bahwa lagu bersifat akustik.
liveness	*Liveness* adalah mendeteksi apakah msuik direkam pada saat konser belangsung atau tidak. Jika nilai 0,8 menunjukkan bahwa lagu direkam pada saat konser berlangsung.

2.4 Prepocessing

Tahapan preprocessing data dilakukan menggunakan Python. Langkah pertama adalah pengecekan duplikasi, namun tidak ditemukan duplikasi dalam dataset. Selanjutnya, dilakukan pengecekan missing values, dan hasilnya tidak ada data yang hilang. Kemudian, data dinormalisasi menggunakan Min-Max Scaling untuk menyamakan skala fitur numerik dalam rentang 0-1. Setelah itu, outlier dideteksi menggunakan Z-Score dan dihapus untuk mencegah gangguan pada analisis. Setelah pembersihan data termasuk pengecekan duplikasi, missing values, dan outlier—jumlah data yang tersisa menjadi 2.537 entri, memastikan dataset lebih representatif dan berkualitas untuk analisis.

2.5 Pengolahan Data

Tahapan pengolahan data dalam penelitian ini mencakup langkah-langkah sistematis yang dilakukan untuk mempersiapkan, mengolah, dan menganalisis data menggunakan algoritma *K-means* dan *K-Means++*.

1) Algoritma K-Means

Algoritma K-Means mengelompokkan data berdasarkan kesamaan fitur dengan memilih sejumlah centroid secara acak sesuai jumlah klaster. Berikut adalah langkah-langkah perhitungannya beserta penjelasan setiap rumusnya:

a) Inisialisasi centroid secara acak

Data awal dipilih secara acak dari dataset untuk menetapkan posisi awal centroid.

b) Hitung jarak data ke setiap centroid

$$d(x_i, C_k) = \sqrt{\sum_{j=1}^{n} \left(x_{ij} - C_{kj}\right)^2} \quad (1)$$

Penjelasan :

x_i = Data ke-i dalam dataset.

C_k = Centroid ke-k

$x_{ij} dan\ C_{kj}$ = Komponen ke j dari data x_i dan centroid C_k

c) Penugasan Klaster

$$Cluster(x_i) = arg_k\ min\ d(x_i, C_k) \quad (2)$$

Penjelasan :

Data x_i akan masuk ke klaster dengan k di mana jaraknya $d(x_i, C_k)$ paling kecil.

arg_k = Operator untuk menemukan indeks k dengan nilai minimum.

d) Memperbarui Posisi Centroid

$$C_k^{(t+1)} = \frac{1}{N_k} \sum_{x_i \in Cluster_k} x^1 \quad (3)$$

Penjelasan :

$C_k^{(t+1)}$ = Centroid baru pada iterasi t + 1.

N_k = Jumlah data dalam klaster ke-k.

$x_i \in Cluster_k$ = Semua data yang termasuk dalam klaster ke-k.

e) Konvergensi

Algortima berhenti ketika perubahan posisi centroid sangat kecil. Jika posisi centroid tidak lagi berubah (konvergen), iterasi dihentikan.

2) Algoritma K-Means++

a) Inisialisasi Centroid Pertama Secara Acak

Centroid pertama dipilih secara acak dari data dalam dataset.

b) Hitung Jarak Kuadrat ke Centroid Terpilih

$$D(x_i) = min_{j=1,\ldots,k} \sum_{m=1}^{n} (x_{im} - C_{jm})^2 \quad (4)$$

Penjelasan:
$D(x_i)$ = Jarak kuadrat minimum dari x_i ke centroid terdekat.
x_{im} = Komponen ke-m dari data x_j
C_{jm} = Komponen ke-m dari data C_j
k = Jumlah centroid yang telah terpilih.

c) Probabilitas Pemilihan Centroid Baru

$$P(x_i) = \frac{D(x_i)}{\sum_{x \in X} D(x)} \quad (5)$$

Penjelasan :
$P(x_i)$ = Probabilitas data x_i untuk dipilih sebagai centroid berikutnya.
$\sum_{x \in X} D(x)$ = Total jarak kuadrat dari semua data ke centroid terdekat.

d) Pemilihan Centroid Baru Secara Probabilistik
e) Melanjutkan Algoritma K-Means

2.6 Evaluasi Hasil Klastering

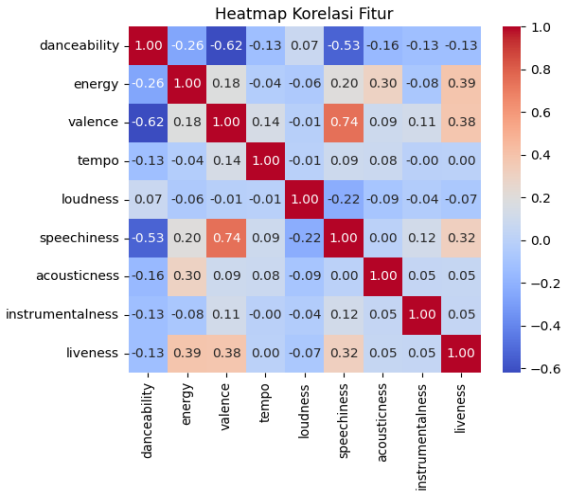
Evaluasi klastering menggunakan Silhouette Score untuk menilai homogenitas dalam klaster dan heterogenitas antar klaster. Dengan nilai -1 hingga 1, skor mendekati 1 menunjukkan klasterisasi yang baik. Evaluasi ini memastikan K-Means dan K-Means++ menghasilkan klaster yang relevan dan representatif sesuai pola data fitur audio Spotify[10].

III. HASIL DAN PEMBAHASAN

3.1 Implementasi Algoritma K-Means

Proses implementasi pada algoritma K-Means menggunakan python untuk proses klasterisasi. Tahapan implementasi meliputi:

1) Analisis Korelasi Antar Fitur Audio

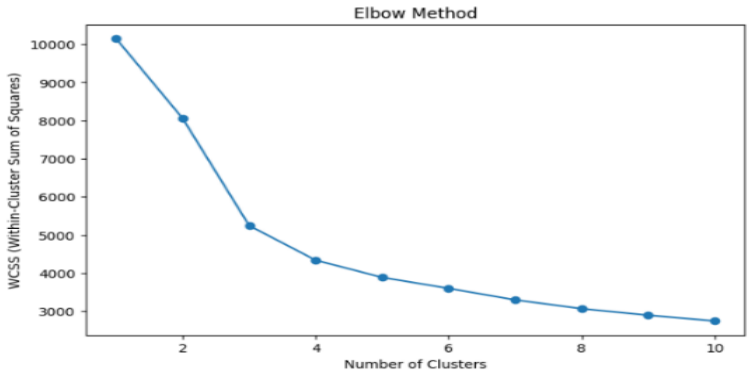


Gambar 2. Heatmap Korelasi Fitur

Analisis korelasi fitur audio menunjukkan beberapa hubungan kuat. Danceability berkorelasi negatif dengan valence (-0.62) dan speechiness (-0.53), menunjukkan bahwa lagu yang mudah ditarikan cenderung lebih melankolis dan minim vokal. Loudness berkorelasi positif dengan energy (0.74), mengindikasikan lagu dengan volume tinggi lebih energik. Speechiness dan valence juga memiliki korelasi positif (0.74), menunjukkan lagu dengan banyak elemen vokal cenderung ceria. Fitur lain memiliki korelasi lebih lemah, seperti tempo yang tidak signifikan kecuali sedikit hubungan dengan loudness (0.25), serta instrumentalness yang tidak terkait dengan energy atau speechiness. Liveness berkorelasi positif dengan energy (0.39), menunjukkan lagu live performance lebih energik.

2) Penentuan Jumlah Cluster

Elbow Method digunakan untuk menentukan jumlah klaster optimal berdasarkan *Within-Cluster Sum of Squares* (WCSS), yang mengukur total jarak data ke centroid klaster. Metode ini memplot WCSS terhadap jumlah klaster (K) dan mengidentifikasi "titik siku", di mana penurunan WCSS mulai melambat, menunjukkan jumlah klaster optimal.



Gambar 3. Grafik Elbow Method

Pada penelitian ini, grafik Elbow Method (Gambar 3) menunjukkan bahwa jumlah klaster optimal adalah 3, yang ditandai dengan perubahan gradien WCSS yang signifikan sebelum mencapai titik tersebut.

3) Proses Klasterisasi

```
KMeans
KMeans(n_clusters=3, n_init=10, random_state=42)
```

Gambar 4. Proses Klasterisasi K-Means

Gambar 4.3 menunjukkan konfigurasi K-Means dengan n_clusters=3, sesuai hasil pada (Gambar 3). Parameter n_init=10 memastikan algoritma berjalan 10 kali dengan centroid awal berbeda, lalu dipilih hasil terbaik. Random_state=42 digunakan untuk menjaga konsistensi hasil klasterisasi, memastikan stabilitas dan optimalisasi klaster.

4) Analisis dan Penamaan Kluster

```
Rata-rata fitur tiap cluster:
         danceability    energy   valence     tempo
Cluster
0            0.159685  0.693351  0.681047  0.119520
1            0.652854  0.548160  0.393462  0.136832
2            0.146130  0.653016  0.710593  0.344749
```

Gambar 5. Rata-rata fitur tiap klaster K-Means

Berdasarkan hasil analisis, Cluster 0 diberi nama Energik karena memiliki energy tinggi (0.693) dan valence positif (0.681), namun dengan tempo lambat (0.119) dan danceability rendah (0.159). Cluster 1 disebut Mellow karena memiliki danceability cukup tinggi (0.652) dan tempo lambat (0.136), tetapi dengan valence rendah (0.393. Terakhir, Cluster 2 dinamakan Ceria karena memiliki energy tinggi (0.653), valence positif (0.710), dan tempo cepat (0.344), meskipun danceability-nya rendah (0.146).

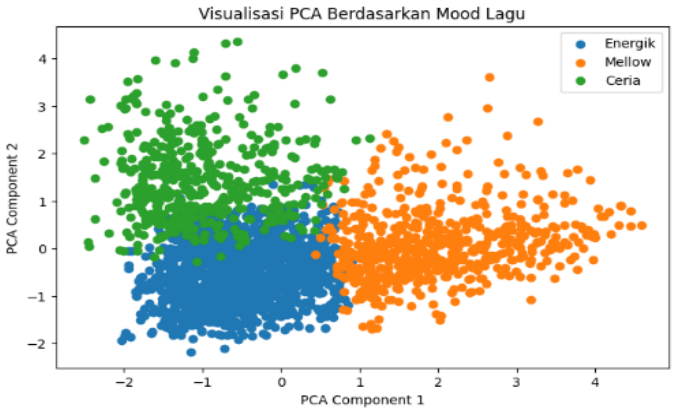
5) Evaluasi

```
Silhouette Score untuk 3 clusters: 0.3223947365342133
```

Gambar 6. Hasil Silhoutte Score K-Means

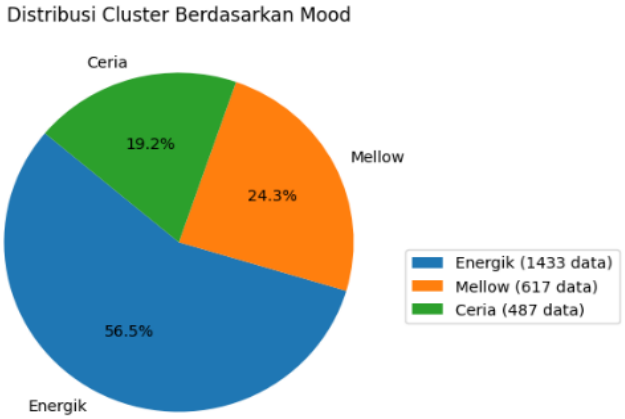
Nilai Silhouette Score sebesar 0.3224 (jika dibulatkan kedalam 4 desimal) pada algoritma K-Means menandakan bahwa kualitas klasterisasi masih belum optimal. Nilai ini menunjukkan bahwa beberapa data belum sepenuhnya terkelompok dengan baik dan masih ada tumpang tindih antara cluster yang satu dengan yang lain.

6) Hasil Klasterisasi



Gambar 7. Visualisasi PCA Berdasarkan Mood

Visualisasi Principal Component Analysis (PCA) pada (Gambar 7) menunjukkan pemisahan tiga kategori mood: Energik, Mellow, dan Ceria. Lagu Energik terdistribusi di kiri bawah plot, Mellow mendominasi bagian kanan dengan penyebaran luas, sementara Ceria berada di tengah hingga atas. Pemisahan ini menunjukkan fitur audio mampu mengidentifikasi karakteristik unik tiap mood, meskipun ada sedikit tumpang tindih antara Ceria dan Energik, kemungkinan karena kombinasi danceability, valence, dan energy yang serupa. Sebaliknya, Mellow lebih terisolasi, menunjukkan ciri khasnya seperti tempo yang lebih lambat.



Gambar 8. Visualisasi Distribusi Cluster

Berdasarkan (Gambar 8) , dapat dilihat bahwa Cluster Energik mendominasi hasil klasterisasi dengan jumlah data sebesar 1.433 lagu (56.5%). Cluster Mellow menempati urutan kedua dengan 617 lagu (24.3%), yang memiliki karakteristik mood yang mellow atau melankolis. Sedangkan, Cluster ceria memiliki jumlah data sebanyak 487 lagu (19.2%), yang mencakup lagu-lagu dengan nuansa ceria.

7) Hasil Lagu Pada Tiap Klaster

Berikut adalah 5 contoh lagu pada tiap Klaster berdasarkan hasil klasterisasi menggunakan algoritma K-Means.

Tabel 2. Hasil Lagu Pada Cluster 0 (Energik)

Cluster 0 (Energik)	
Name	***Artist***
Snooze	SZA
Nonsense	Sabrina Carpenter
End of Beginning	*Djo*
Kill Bill	SZA
Oklahoma Smokeshow	Zach Bryan

Tabel 3. Hasil Lagu Pada Cluster 1 (Mellow)

Cluster 1 (Mellow)	
Name	***Artist***
Something in the Orange	Zach Bryan
Stick Season	Noah Kahan
Sun to Me	Zach Bryan
Wondering Why	The Red Clay Strays
Burn, Burn, Burn	Zach Bryan

Tabel 4. Hasil Lagu Pada Cluster 2 (Ceria)

Cluster 2 (Ceria)	
Name	***Artist***
Romantic Homicide	d4vd
Bad Habit	Steve Lacy
As It Was	Harry Styles
Broadway Girls (feat. Morgan Wallen)	Lil Durk
I'm Good (Blue)	David Guetta

3.2 Implementasi Algoritma K-Means++

1) Proses Pemilihan Centroid

```
Centroids awal dari KMeans++:
[[-0.59243828  0.54288149 -0.16547212 -0.65123938]
 [-0.77235288 -3.27804797  1.50117991  2.03232212]
 [ 1.26189617 -1.60975482  0.01194568 -0.54343901]]
```

Gambar 9. Proses Pemilihan Centroid K-Means++

Algoritma K-Means++ dimulai dengan pemilihan centroid secara optimal dibandingkan inisialisasi acak. Gambar 9 menampilkan hasil inisialisasi centroids yang diperoleh dari algoritma K-Means++. Terdapat tiga pusat klaster awal dengan masing-masing nilai yang menunjukkan koordinat dalam ruang fitur.

2) Proses Klasterisasi

```
# Implementasi KMeans dengan centroid yang sudah diinisialisasi
from scipy.spatial.distance import cdist

for _ in range(100):
    # Menghitung jarak setiap titik ke centroid
    distances = cdist(data_scaled, initial_centroids, 'euclidean')
    labels = np.argmin(distances, axis=1)

    # Memperbarui centroid
    new_centroids = np.array([data_scaled[labels == cluster].mean(axis=0) for cluster in range(n_clusters_optimal)])

    # Periksa konvergensi
    if np.all(initial_centroids == new_centroids):
        break

    initial_centroids = new_centroids

# Menambahkan label cluster ke dataset
data['Cluster'] = labels
```

Gambar 10. Proses Klasterisasi K-Means++

Pada Gambar 10. pertama-tama dihitung jarak Euclidean antara setiap titik data yang telah dinormalisasi dengan centroid yang ada. Kemudian, setiap titik data diberikan label klaster berdasarkan centroid terdekat. Setelah itu, centroid diperbarui dengan menghitung rata-rata dari semua titik data dalam masing-masing klaster. Proses ini berulang hingga centroid tidak mengalami perubahan, yang menandakan bahwa algoritma telah mencapai konvergensi.

3) Analisis dan Penamaan Klaster

```
Rata-rata fitur tiap cluster:
         danceability    energy   valence     tempo
Cluster
0            0.159644  0.693494  0.680977  0.119616
1            0.146223  0.652512  0.710862  0.344930
2            0.652854  0.548160  0.393462  0.136832
```

Gambar 11. Rata-rata fitur tiap klaster pada K-Means++

Gambar 11. menunjukkan rata-rata fitur audio setiap klaster hasil K-Means++. Cluster 0 (Energik) memiliki valence positif (0.680977) dan energy tinggi (0.693494), tetapi danceability rendah (0.159644) dan tempo lambat (0.119616). Cluster 1 (Ceria) memiliki valence tertinggi (0.710862), energy tinggi (0.652512), dan tempo cepat (0.344930), meskipun danceability rendah (0.146223). Cluster 2 (Mellow) memiliki danceability tinggi (0.652854), valence rendah (0.393462), energy sedang (0.548160), dan tempo lambat (0.136832).

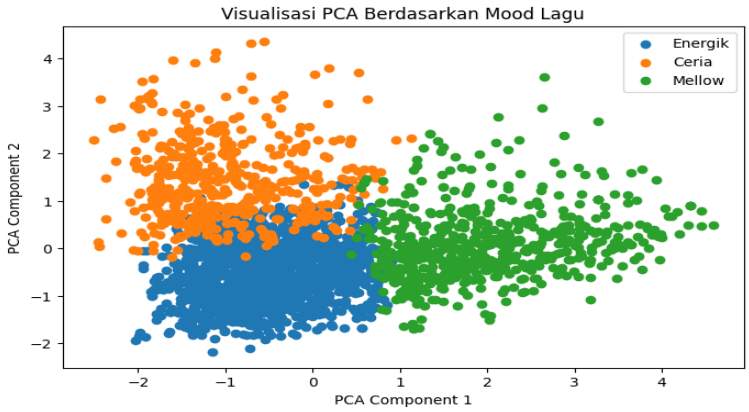
4) Evaluasi

```
Silhouette Score untuk 3 clusters: 0.32246467825287023
```

Gambar 12. Hasil Silhoutte Score K-Means++

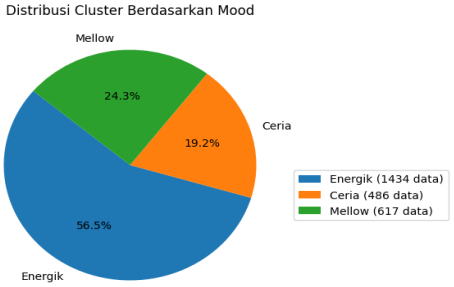
Nilai Silhouette Score sebesar 0.3225 pada K-Means++ menunjukkan bahwa kualitas klasterisasi masih belum optimal, dengan beberapa data belum terkelompok dengan baik dan masih terdapat tumpang tindih antar klaster.

5) Hasil Klaster



Gambar 13. Visualisasi PCA Berdasarkan Mood

Visualisasi PCA pada Gambar 13 menunjukkan pemisahan tiga mood: Ceria, Mellow, dan Energik. Lagu Ceria terdistribusi di kiri bawah, Mellow mendominasi kanan dengan penyebaran luas, dan Energik terkonsentrasi di tengah-atas. Meskipun fitur audio mampu mengidentifikasi karakteristik unik tiap mood, terdapat tumpang tindih antara Ceria dan Energik, kemungkinan karena kesamaan danceability, valence, dan energy. Sebaliknya, Mellow lebih terisolasi, menunjukkan ciri khas seperti tempo lambat dan energi lebih rendah.



Gambar 14. Visualisasi Distribusi Cluster

Berdasarkan Gambar 14, Cluster Energik mendominasi dengan 1.434 lagu (56.5%), menunjukkan mayoritas lagu memiliki energi tinggi. Cluster Mellow berisi 617 lagu (24.3%) dengan karakteristik melankolis, sementara Cluster Ceria terdiri dari 486 lagu (19.2%), mencerminkan mood ceria.

6) Hasil Lagu Pada Tiap Klaster

Berikut adalah 5 contoh lagu pada tiap Klaster berdasarkan hasil klasterisasi menggunakan algoritma K-Means.

Tabel 5. Hasil Lagu Pada Cluster 0 (Energik)

Cluster 0 (Energik)	
Name	***Artist***
Snooze	SZA
Nonsense	Sabrina Carpenter
End of Beginning	*Djo*
Kill Bill	SZA
Oklahoma Smokeshow	Zach Bryan

Tabel 6. Hasil Lagu Pada Cluster 2 (Ceria)

Cluster 2 (Ceria)	
Name	***Artist***
Romantic Homicide	d4vd
Bad Habit	Steve Lacy
As It Was	Harry Styles
Broadway Girls (feat. Morgan Wallen)	Lil Durk
I'm Good (Blue)	David Guetta

Tabel 7. Hasil Lagu Pada Cluster 1 (Mellow)

Cluster 1 (Mellow)	
Name	***Artist***
Something in the Orange	Zach Bryan
Stick Season	Noah Kahan
Sun to Me	Zach Bryan
Wondering Why	The Red Clay Strays
Burn, Burn, Burn	Zach Bryan

3.3 Hasil Perbandingan

Hasil evaluasi Silhouette Score menunjukkan bahwa K-Means memiliki skor 0.3224, sementara K-Means++ sedikit lebih tinggi, yaitu 0.3225. Meskipun perbedaannya kecil, K-Means++ lebih unggul karena strategi pemilihan centroid awal yang lebih optimal dibandingkan pemilihan acak pada K-Means. Meskipun peningkatan performa tidak signifikan, K-Means++ tetap lebih direkomendasikan untuk dataset kompleks atau sensitif terhadap inisialisasi centroid.

IV. KESIMPULAN

Penelitian ini menunjukkan bahwa klasterisasi lagu pada Spotify menggunakan algoritma K-Means dan K-Means++ dapat mengelompokkan lagu ke dalam tiga klaster utama, yaitu Ceria, Mellow, dan Energik, berdasarkan fitur audio seperti danceability, energy, valence, tempo, dan loudness. Hasil evaluasi menunjukkan bahwa K-Means++ lebih optimal dibandingkan K-Means, dengan Silhouette Score sebesar 0.32246467825287023. Klaster Energik (56.5%) menjadi yang paling dominan, diikuti Mellow (24.3%) dan Ceria (19.2%). Visualisasi menunjukkan pemisahan klaster yang cukup jelas, meskipun terdapat sedikit tumpang tindih antara klaster Ceria dan Energik.

REFERENSI

[1]. Marlia, S., Setiawan, K., Juliane, C., Bisnis, S. I., Informasi, S., Likmi, S., Jalan, B., Juanda, I. H., & 96, N. (2024). Analisis Fitur Musik dan Tren Popularitas Lagu di Spotify menggunakan K-Means dan CRISP-DM Analysis of Music Features and Song Popularity Trends on Spotify Using K-Means and CRISP-DM. http://sistemasi.ftik.unisi.ac.id

[2]. Nuriska, D., Irawan, B., Bahtiar, A., & Rinaldi Dikananda, A. (2023). KLASTERISASI DATA LAGU TERPOPULER SPOTIFY 2023 BERDASARKAN SUASANA HATI MENGGUNAKAN ALGORITMA K-MEANS. In Jurnal Mahasiswa Teknik Informatika (Vol. 7, Issue 6).

[3]. Nugroho, N., & Adhinata, F. D. (2022). Penggunaan Metode K-Means dan K- Means++ Sebagai Clustering Data Covid-19 di Pulau Jawa. Teknika, 11(3), 170–179. https://doi.org/10.34148/teknika.v11i3.502

[4]. Nurhalimah, L., Hermanto, T. I., & Kaniawulan, I. (2022). Analisis PrediksiMood Genre Musik Pop Menggunakan Algoritma K-Means dan C4.5.JURIKOM (Jurnal Riset Komputer), 9(4), 1006. https://doi.org/10.30865/jurikom.v9i4.4597

[5]. Odelia, F., & Sembiring, S. (2024). Analisis Kepuasan Pengguna AplikasiSpotify Dengan Menggunakan Metode UTAUT. Jurnal Media Akademik (JMA), 2(2), 3031–5220. https://doi.org/10.62281

[6]. Putra Nugraha, R., Laxmi, G. F., & Riana, F. (2024). PENERAPAN K-MEANS++ UNTUK PENGELOMPOKAN MAHASISWA BERPOTENSI DROP OUT (STUDI KASUS: UNIVERSITAS IBN KHALDUN BOGOR). In Jurnal Mahasiswa Teknik Informatika (Vol. 8, Issue 3).

[7]. Reyhan Jarsi Yoga, Basuki Rahmat, & Eka Prakarsa Mandyartha. (2024). Analisis Klaster dan Klasifikasi Emosi Dalam Musik K-Pop dengan K-Means dan Algoritma C 4.5. Neptunus: Jurnal Ilmu Komputer Dan Teknologi Informasi, 2(3), 147–160. https://doi.org/10.61132/neptunus.v2i3.228

[8]. Ramdani, C., & Safadila, N. (2022). LEDGER: Journal Informatic andInformation Technology Analisis Data Akademis dengan Menerapkan 62 Algoritme K-Means dan K-Means++. OPEN ACCESS LEDGER, 1(4). https://doi.org/10.20895/LEDGER.V1I4.918

[9]. Riziq sirfatullah Alfarizi, M., Zidan Al-farish, M., Taufiqurrahman, M.,Ardiansah, G., & Elgar, M. (2023). PENGGUNAAN PYTHONSEBAGAI BAHASA PEMROGRAMAN UNTUK MACHINE LEARNING DAN DEEP LEARNING. In Karimah Tauhid (Vol. 2, Issue1).

[10]. Rohman, N., & Wibowo, A. (2024). CLUSTERING OF POPULAR SPOTIFYSONGS IN 2023 USING K-MEANS METHOD AND SILHOUETTE COEFFICIENT. Jurnal Pilar Nusa Mandiri, 20(1), 18–24. https://doi.org/10.33480/pilar.v20i1.4937