

Implementasi Algoritma K-Means *Clustering* Berdasarkan Karakteristik Suara untuk Pengelompokan Gender pada Data Audio *Waveform*

Isbatudinia¹, Abdurahman^{2*}, Sukemi³, M. Ali Buchari⁴, Siti Latifah⁵, Wanda Hamidah⁶, Rusdi Efendi⁷
^{1,2,3,4,5,6}Program Studi Sistem Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya, Palembang, Indonesia
⁷Program Studi Manajemen Informatika, Fakultas Ilmu Komputer, Universitas Sriwijaya, Palembang, Indonesia
email : *abdurahman@unsri.ac.id

Abstrak— Penelitian ini mengimplementasikan algoritma K-Means Clustering untuk mengidentifikasi gender berdasarkan karakteristik suara. Metodologi melibatkan pre-processing data audio, ekstraksi fitur menggunakan Mel-Frequency Cepstral Coefficients (MFCC), dan pengelompokan dengan K-Means. Hasil menunjukkan keberhasilan dalam memisahkan suara menjadi dua cluster: 70.2% suara perempuan dan 29.8% suara laki-laki. Kombinasi MFCC dan K-Means terbukti efektif dalam menangkap perbedaan karakteristik frekuensi antar gender. Metode ini menunjukkan potensi signifikan untuk aplikasi pengenalan suara berbasis gender tanpa memerlukan label awal.

Kata Kunci— *Clustering, K-Means, Pengenalan Suara, Identifikasi Gender, MFCC.*

I. PENDAHULUAN

Identifikasi karakteristik suara berdasarkan gender merupakan salah satu implementasi yang semakin relevan dalam pengembangan sistem keamanan, verifikasi identitas berbasis suara. Mengklasifikasikan suara manusia dengan tingkat akurasi dan presisi yang lebih tinggi dengan akurasi yang semakin meningkat, membuka jalan bagi pengembangan sistem yang lebih canggih dalam mendeteksi pola suara [1].

Salah satu tantangan utama dalam klasifikasi suara adalah bagaimana mengekstraksi fitur yang relevan dari sinyal suara yang kompleks. ariasi karakteristik suara individu, serta pengaruh faktor seperti usia, kesehatan, latar belakang budaya, dan bahasa, membuat proses ini semakin sulit [2].

Untuk itu, penggunaan Mel-Frequency Cepstral Coefficients (MFCC) telah menjadi metode yang paling banyak digunakan. MFCC dikenal efektif dalam menangkap frekuensi suara sesuai dengan cara manusia mendengar, menjadikannya sangat berguna dalam pengenalan ucapan dan klasifikasi suara.

Penelitian ini bertujuan untuk mengatasi tantangan tersebut dengan menerapkan teknik pengelompokan menggunakan algoritma K-Means Clustering pada data suara yang diperoleh. K-Means dipilih karena kemampuannya dalam memisahkan data suara berdasarkan kemiripan fitur tanpa memerlukan label awal.

Dalam penelitian ini, penggunaan MFCC dan K-Means diharapkan dapat mengidentifikasi pola pengelompokan suara berdasarkan gender secara efektif. Dengan mengoptimalkan tahap pre-processing seperti normalisasi dan trimming menggunakan Librosa, penelitian ini berupaya menjaga kualitas data suara dan meminimalkan distorsi serta noise yang tidak relevan.

II. PENELITIAN TERKAIT

Asma, Mansour, dan Zied Lachiri mereka menekankan pandangan sistematis untuk mengidentifikasi penguat di bawah berbagai kondisi emosional berdasarkan buku kerja multi-vektor mesin dukungan (SVM). Penguatan kinerja proses pengenalan pembicara emosional telah menerima perhatian yang meningkat dalam beberapa tahun terakhir.

Penulis membandingkan dua metode untuk mengekstrak fitur, untuk mendapatkan fitur yang paling akurat digunakan untuk menyajikan urutan bicara psikologis. Mereka menggunakan dua metode, metode pertama adalah MFCC dan metode lain adalah SDC, keduanya digabungkan dengan MFCC (SDC-MFCC). Dua metode ini diproses oleh atribut rata-rata dan varian.

Eksperimen dilakukan pada database EMOCAP menggunakan dua pendekatan multi-lapisan SFM satu melawan semua (OAA) dan satu melawan satu (OAO). Hasil yang diperoleh menunjukkan bahwa SDC-MFCC lebih unggul daripada performa sekurup biasa.

III. PUSTAKA SOFTWARE

Python adalah sebuah bahasa pemrograman yang fleksibel dan powerful, yang menawarkan kemampuan interpretasi, interaktivitas, dan orientasi objek [3]. Python dikenal karena kemudahannya untuk dipelajari serta statusnya sebagai perangkat lunak sumber terbuka. Bahasa ini menggabungkan kekuatan yang luar biasa dengan sintaksis yang sangat jelas.

OS (*Operating System*) adalah modul bawaan Python yang menyediakan cara untuk berinteraksi dengan sistem operasi. *Library* ini memungkinkan kode Python untuk melakukan berbagai operasi yang terkait dengan sistem operasi secara portabel, artinya kode yang sama dapat berjalan di berbagai sistem operasi seperti Windows, macOS, atau Linux [4].

NumPy adalah perpustakaan komputasi ilmiah *open-source* yang kuat dan efisien untuk Python. Pustaka ini menyediakan berbagai kemampuan yang kuat seperti objek array N-dimensi, fungsi-fungsi canggih (*broadcasting*), alat yang kuat untuk mengintegrasikan kode C/C++ dan *Fortran* dengan Python, serta fungsionalitas aljabar linier, transformasi Fourier, dan pembangkitan bilangan acak yang berguna [5].

Librosa adalah pustaka Python yang dirancang khusus untuk analisis musik dan audio. Pustaka ini menyediakan berbagai fungsi dan alat yang diperlukan untuk membangun sistem temu kembali informasi musik (*Music Information Retrieval* - MIR). Librosa memungkinkan pengembang untuk

melakukan berbagai tugas pemrosesan sinyal audio, termasuk ekstraksi fitur seperti *Mel-frequency Cepstral Coefficients* (MFCCs), *chroma*, *onset detection*, *beat tracking*, dan spektrogram. Pustaka ini juga menyediakan fungsi untuk visualisasi audio, yang sangat berguna dalam analisis dan pemahaman karakteristik suara [6].

Pandas adalah sebuah perangkat lunak analisis dan manipulasi data *open-source* yang handal, efisien, dan mudah digunakan, yang dibangun di atas bahasa pemrograman Python [7].

Scikit-learn adalah pustaka *machine learning open-source* untuk bahasa pemrograman Python. Pustaka ini menyediakan berbagai algoritma untuk klasifikasi, regresi, dan klusterisasi. Scikit-learn dirancang untuk kemudahan penggunaan dan efisiensi, sehingga banyak digunakan oleh peneliti dan praktisi di berbagai bidang. Selain itu, Scikit-learn terintegrasi dengan pustaka lain seperti NumPy, Pandas, dan Matplotlib, memungkinkan pengguna untuk dengan mudah mengelola data, melakukan analisis, dan memvisualisasikan hasil [8].

Mel-Frequency Cepstral Coefficients (MFCC) adalah representasi koefisien dari sinyal suara dalam domain frekuensi, yang sangat efektif untuk digunakan dalam tugas-tugas seperti pengenalan suara dan klasifikasi suara. MFCC meniru bagaimana telinga manusia merespons suara, menjadikannya fitur yang sangat relevan dalam analisis audio berbasis *machine learning* [9].

Clustering adalah metode dalam Data Mining yang digunakan untuk mengelompokkan objek-objek data yang memiliki kemiripan ke dalam kelompok atau kluster. Tujuan utama dari teknik ini adalah untuk mengungkap pola atau struktur tersembunyi dalam data yang belum diberi label atau klasifikasi sebelumnya. Teknik ini sering diterapkan dalam berbagai bidang seperti pengenalan pola, analisis pasar, dan bioinformatika, di mana klasifikasi manual tidak memungkinkan atau tidak praktis. [10].

K-Means adalah algoritma *clustering* yang tidak terawasi, yang berfungsi untuk mengelompokkan data ke dalam sejumlah *cluster* berdasarkan kedekatan atau kesamaan karakteristik. Dalam konteks pengenalan suara, K-Means dapat digunakan untuk mengelompokkan data audio ke dalam kategori tertentu (misalnya, gender) tanpa adanya label awal [11].

Waveform adalah representasi grafis dari variasi suatu sinyal dalam waktu. Biasanya, *waveform* digunakan dalam konteks sinyal listrik, suara, atau gelombang lainnya. Dalam audio, *waveform* menunjukkan amplitudo sinyal suara terhadap waktu, yang memungkinkan analisis karakteristik suara seperti frekuensi dan durasi. *Waveform* dapat ditampilkan dalam berbagai bentuk, seperti gelombang sinusoidal, persegi, atau segitiga, tergantung pada sifat sinyal yang dianalisis [12].

CSV (*Comma-Separated Values*) adalah format file yang digunakan untuk menyimpan data tabular, di mana setiap baris dalam file mewakili satu entri data dan setiap nilai dipisahkan oleh koma. Format ini sangat populer karena kesederhanaannya dan kemudahan untuk dibaca oleh manusia serta diolah oleh berbagai program perangkat lunak, termasuk spreadsheet seperti Microsoft Excel dan Google Sheets [13].

IV. METODOLOGI PENELITIAN

Penelitian ini menggunakan algoritma *clustering* dengan tujuan mengidentifikasi gender berdasarkan karakteristik suara pada data audio *waveform*. Pengembangan algoritma ini melibatkan rangkaian tahapan seperti *flowchart* dibawah ini:



Gambar 1. Flowchart Tahap Penelitian

Berikut adalah penjelasan rinci dari setiap tahap yang dilakukan:

A. Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh dari platform Kaggle, yang merupakan salah satu sumber data terbuka yang terkenal di kalangan peneliti dan praktisi data. Dalam dataset ini, terdapat sejumlah total 228 rekaman suara yang diambil, dengan durasi setiap rekaman berkisar antara 2 hingga 5 detik. Semua rekaman suara ini menggunakan Bahasa Inggris sebagai bahasa utama. Penting untuk dicatat bahwa rekaman suara tersebut mencakup kontribusi dari berbagai gender, dengan jumlah rekaman yang mewakili baik suara laki-laki maupun suara perempuan. Semua rekaman tersebut disimpan dalam format file WAV, yang merupakan salah satu format audio standar yang umum digunakan dalam pengolahan suara. Dengan menggunakan dataset ini, penelitian bertujuan untuk menganalisis dan mengeksplorasi karakteristik serta pola yang terdapat dalam pengucapan dan intonasi suara berdasarkan gender.

B. Pre-Processing

Data suara yang telah terkumpul kemudian melalui proses *preprocessing* untuk mempersiapkan analisis lebih lanjut. Proses ini dimulai dengan menggunakan paket Librosa, yang dikenal luas dalam komunitas pemrosesan audio. Librosa digunakan untuk memuat file audio dalam format WAV dan untuk mengekstrak data sinyal audio beserta sample rate-nya. Pemilihan Librosa didasarkan pada kemudahan penggunaannya, fitur-fitur yang komprehensif, serta integrasi yang baik dengan ekosistem Python. Alat ini sangat ideal untuk ekstraksi fitur audio, seperti *Mel-frequency cepstral coefficients* (MFCC), dalam konteks pemrosesan sinyal audio.

Dengan demikian, Librosa menawarkan efisiensi tinggi, memungkinkan analisis audio dan pengembangan aplikasi

pemrosesan sinyal tanpa memerlukan langkah-langkah manual yang berlebihan. Setelah memuat data audio, langkah selanjutnya adalah melakukan normalisasi terhadap sinyal audio agar nilainya berada dalam rentang yang diinginkan. Proses normalisasi ini dilakukan dengan membagi semua nilai sinyal audio dengan nilai absolut maksimum dari dataset tersebut. Tujuan dari normalisasi adalah untuk memastikan bahwa amplitudo sinyal audio terletak dalam rentang $[-1, 1]$, yang merupakan standar umum dalam pemrosesan sinyal.

Pendekatan menggunakan nilai absolut maksimum untuk normalisasi dianggap tepat dalam konteks pemrosesan sinyal audio karena mampu menjaga amplitudo sinyal tanpa mengakibatkan distorsi atau amplifikasi yang berlebihan. Meskipun metode lain seperti normalisasi *mean*, *root mean square*, atau min-max juga tersedia, pendekatan ini lebih unggul, terutama ketika berhadapan dengan sinyal yang mengandung nilai negatif, noise, atau variasi amplitudo yang signifikan.

Selanjutnya, tahap *preprocessing* melibatkan penghilangan bagian dari audio yang dianggap sebagai keheningan atau noise melalui proses *trimming*. Metode *trimming* digunakan untuk memangkas bagian awal dan akhir sinyal audio yang tidak relevan, dengan tujuan membersihkan sinyal dari data yang tidak diperlukan. Librosa dipilih untuk melakukan *trimming* karena efisiensinya dalam menghapus keheningan berdasarkan amplitudo dengan cara yang cepat dan mudah, serta integrasinya yang baik dengan analisis fitur audio lainnya.

Proses ini memungkinkan pengguna untuk melakukan pemrosesan audio secara otomatis dan konsisten, serta memberikan fleksibilitas dalam menyesuaikan *threshold* atau menambahkan efek lainnya sesuai kebutuhan. Meskipun terdapat library lain seperti PyDub atau Sox yang dapat digunakan untuk tujuan tertentu, Librosa tetap menjadi pilihan ideal bagi aplikasi pemrosesan sinyal audio yang terfokus pada analisis fitur, seperti dalam konteks pengenalan suara atau pengenalan musik.

C. Ekstraksi Fitur

Fitur akustik dari sinyal suara diekstraksi untuk merepresentasikan karakteristik utama dari suara tersebut. Proses ekstraksi ini menggunakan metode *Mel Frequency Cepstral Coefficients* (MFCC), yang dipilih karena keunggulannya dalam meniru cara telinga manusia memproses suara, terutama dalam menangkap frekuensi-frekuensi yang relevan untuk ucapan dan suara. Dengan memetakan frekuensi ke skala Mel yang mencerminkan persepsi manusia, MFCC mampu menangkap fitur suara yang paling signifikan secara efisien dan tahan terhadap noise, menjadikannya sangat cocok untuk berbagai aplikasi, seperti pengenalan suara, *speech recognition*, dan pengklasifikasian suara.

Keunggulan lain dari MFCC adalah kemampuannya untuk mengompresi informasi menjadi sejumlah koefisien yang relatif kecil, sehingga membuatnya efisien dalam hal penyimpanan dan pemrosesan tanpa kehilangan esensi informasi akustik yang penting. Selain itu, MFCC telah terbukti efektif dalam berbagai aplikasi real-time berkat proses ekstraksi yang cepat dan stabil.

Di sisi lain, terdapat alternatif lain seperti *chroma features*, *spectral features*, dan *Linear Predictive Coding* (LPC), yang cenderung lebih sesuai untuk konteks-konteks

tertentu, seperti analisis musik atau rekonstruksi suara. Namun, alternatif-alternatif ini kurang efektif dalam pengenalan ucapan atau pengolahan suara manusia. Oleh karena itu, MFCC tetap menjadi standar de facto dalam pemrosesan suara, berkat keseimbangan antara akurasi, ketahanan, dan efisiensi komputasi, menjadikannya pilihan yang paling andal untuk tugas-tugas yang berhubungan dengan analisis ucapan dan suara manusia.

Proses ekstraksi fitur ini melibatkan beberapa langkah sebagai berikut:

- Transformasi ke Domain Frekuensi: Sinyal suara dipecah menjadi beberapa frame, yang kemudian diubah ke dalam domain frekuensi menggunakan Transformasi Fourier Cepat (FFT). Proses ini dinyatakan dengan persamaan berikut:

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j(2\pi kn/N)} \quad (1)$$

di mana $X(k)$ adalah hasil dari Transformasi Fourier dari sinyal $x(n)$, dan N merupakan panjang frame.

- Mel-scale Filterbank: Frekuensi yang dihasilkan dari proses FFT kemudian diterapkan pada filter Mel-scale, yang berfungsi untuk mengonversi frekuensi nyata ke skala yang lebih sesuai dengan persepsi manusia. Rumus yang digunakan dalam konversi ini adalah:

$$M(f) = 2595 \cdot \log_{10} \left(1 + \frac{f}{700} \right) \quad (2)$$

Skala Mel ini memberikan representasi yang lebih linear pada rentang frekuensi rendah dan logaritmik pada frekuensi tinggi, mencerminkan cara manusia mendengar suara.

- Logaritmik Amplitudo dan Transformasi Cosinus Diskrit (DCT): Setelah sinyal melewati filter Mel, amplitudo logaritmik dihasilkan dan kemudian diterapkan Transformasi Cosinus Diskrit (DCT) untuk menghasilkan 13 koefisien MFCC. Koefisien ini berfungsi sebagai representasi kompak dari frekuensi yang diambil dari suara. Fitur ini sering digunakan dalam berbagai aplikasi karena kemampuannya untuk menangkap karakteristik penting dalam klasifikasi suara, seperti pengenalan gender.

Selanjutnya, dilakukan penghitungan rata-rata (mean) dari setiap koefisien MFCC untuk menyederhanakan representasi fitur. Proses ini melibatkan transposisi matriks MFCC menggunakan perintah `mfccs.T`, sehingga koefisien MFCC untuk setiap frame audio berada dalam baris yang terpisah. Kemudian, fungsi `np.mean(mfccs.T, axis=0)` digunakan untuk menghitung rata-rata dari setiap koefisien MFCC di sepanjang semua frame. Hasil dari proses ini adalah satu vektor yang terdiri dari 13 nilai rata-rata MFCC, yang menyederhanakan fitur audio menjadi satu set representasi yang lebih ringkas.

Penggunaan 13 koefisien MFCC dalam ekstraksi fitur suara merupakan hasil dari keseimbangan yang optimal antara akurasi representasi dan efisiensi komputasi. Dalam praktiknya, 12 hingga 13 koefisien MFCC sudah cukup untuk menangkap informasi penting dari spektrum suara yang relevan bagi pengenalan ucapan dan analisis audio. Koefisien pertama umumnya mewakili energi total sinyal, sementara

koefisien berikutnya mencerminkan karakteristik frekuensi dari formant-formant penting yang terkait dengan vokal dan konsonan. Dengan menggunakan 13 koefisien, kami dapat merepresentasikan perubahan dalam suara dengan detail yang memadai, tanpa menghasilkan data yang berlebihan yang dapat menyebabkan *overfitting* jika dilakukannya modeling.

Selain itu, penggunaan jumlah koefisien ini telah terbukti efektif melalui berbagai eksperimen dalam bidang pengenalan suara, sehingga menjadikannya standar yang diakui secara luas untuk analisis ucapan dan suara. Dengan kata lain, 13 koefisien MFCC memberikan gambaran menyeluruh mengenai karakteristik suara, sambil tetap mempertahankan efisiensi dalam pemrosesan data. Hal ini menjadikannya ideal untuk aplikasi berbasis suara yang membutuhkan kecepatan dan akurasi tinggi. Berikut penjelasan tiap koefisiennya:

1. MFCC-1 (Koefisien energi): Intensitas atau kekuatan sinyal suara.
2. MFCC-2 hingga MFCC-3: Karakteristik frekuensi rendah, mencerminkan formant pertama dan kedua, yang penting untuk identifikasi vokal dan karakteristik dasar suara.
3. MFCC-4 hingga MFCC-5: Berhubungan dengan formant ketiga dan beberapa variasi dalam vokal, membantu dalam membedakan jenis ucapan dan suara.
4. MFCC-6 hingga MFCC-13: Menangkap detail frekuensi yang lebih tinggi, berhubungan dengan karakteristik suara yang lebih halus seperti nada, aksen, dan nuansa suara.

Ilustrasi dalam Konteks Ucapan:

- MFCC-2 hingga MFCC-4 mencerminkan informasi yang penting untuk membedakan antara vokal seperti /a/, /i/, dan /u/, karena formant-formant ini bervariasi berdasarkan posisi lidah dan bentuk saluran vokal.
- MFCC-6 hingga MFCC-13 dapat mencerminkan informasi yang lebih halus, seperti nasalitas atau konsonan frikatif (seperti /s/ atau /f/), yang bergantung pada komponen frekuensi tinggi.

D. Pengelompokan Suara Menggunakan K-Means

Setelah proses ekstraksi fitur suara selesai, langkah selanjutnya adalah melakukan pengelompokan (*clustering*) untuk memisahkan data suara menjadi dua kelompok utama, yaitu suara perempuan dan suara laki-laki. Pengelompokan ini dilakukan menggunakan algoritma *K-Means Clustering*, yang mengelompokkan data berdasarkan kemiripan fitur yang telah diekstraksi sebelumnya.

Dalam penelitian ini, nilai *K* pada algoritma K-Means ditentukan menjadi dua, sesuai dengan jumlah kelompok target (0 untuk perempuan dan 1 untuk laki-laki). Dengan menggunakan *centroid* sebagai titik pusat, algoritma K-Means memposisikan setiap data suara ke dalam salah satu dari kedua kategori tersebut. Hasil dari proses pengelompokan ini berfungsi untuk memvalidasi konsistensi label gender yang diberikan serta mendukung proses analisis lebih lanjut pada model klasifikasi.

Pemilihan K-Means dalam penelitian ini didasarkan pada kombinasi kesederhanaan, efisiensi, dan kemampuan

algoritma ini dalam mengelompokkan data numerik, menjadikannya pilihan yang sangat baik. Kelebihanannya dalam memproses data berdimensi tinggi, kecepatan eksekusi, serta kemudahan dalam interpretasi hasil sangat sesuai untuk analisis audio yang kompleks. Dengan menggunakan K-Means, data audio dapat dikelompokkan dengan cara yang informatif dan mendalam, memungkinkan analisis yang lebih lanjut serta pengembangan aplikasi berbasis audio yang lebih baik.

Ada juga langkah-langkah *clustering* menggunakan K-Means:

1. Menentukan Jumlah *Cluster* (*k*): Tentukan jumlah *cluster* yang diinginkan. Ini bisa dilakukan berdasarkan pengetahuan domain atau menggunakan teknik seperti *Elbow Method* atau *Silhouette Score* untuk mencari jumlah *cluster* yang optimal.
2. Inisialisasi *Centroid*: Pilih *k* titik data secara acak dari dataset sebagai *centroid* awal. Alternatifnya, metode K-Means++ dapat digunakan untuk pemilihan *centroid* yang lebih baik, yang dapat meningkatkan konvergensi algoritma.
3. Menghitung Jarak: Hitung jarak antara setiap titik data dan *centroid* yang ada menggunakan jarak *Euclidean*. Rumus jarak *Euclidean* untuk dua titik X_i dan C_j adalah:

$$d(X_i, C_j) = \sqrt{\sum_{m=1}^p (X_{im} - C_{jm})^2} \quad (3)$$

Di mana p adalah jumlah fitur.

4. Menetapkan Label *Cluster*: Tetapkan setiap titik data ke *cluster* terdekat berdasarkan jarak yang dihitung. Titik data X_i akan diberikan label *cluster* j jika jaraknya ke *centroid* C_j adalah yang terdekat.
5. Memperbarui *Centroid*: Setelah semua titik data ditetapkan ke *cluster*, perbarui posisi *centroid* dengan menghitung rata-rata dari semua titik data dalam setiap *cluster*. Rumus untuk menghitung *centroid* baru adalah:

$$C_j^{new} = \frac{1}{n_j} \sum_{i=1}^{n_j} X_i \quad (4)$$

Di mana n_{j_nj} adalah jumlah titik data dalam *cluster* j .

6. Mengulangi Proses: Ulangi langkah 4 sampai 6 (menghitung jarak, menetapkan label, memperbarui *centroid*) hingga posisi *centroid* tidak lagi berubah secara signifikan atau hingga konvergensi tercapai (misalnya, mencapai batas maksimum iterasi).
7. Menghasilkan Output: Setelah konvergensi tercapai, hasil *clustering* dapat ditampilkan, termasuk *centroid* akhir dari setiap *cluster* dan label *cluster* untuk setiap titik data.

Persamaan dasar K-Means:

Pada intinya, K-Means berusaha meminimalkan variansi dalam setiap *cluster* dan memaksimalkan jarak antar *cluster*. Persamaan yang mendasari tujuan K-Means dapat dituliskan sebagai:

$$J(C) = \sum_{j=1}^k \sum_{x_i \in C_j} \|x_i - c_j\|^2 \quad (5)$$

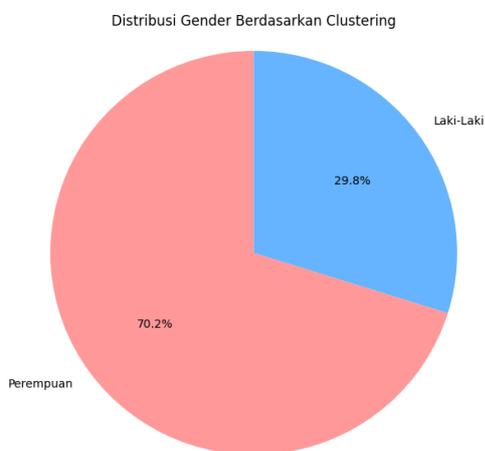
Di mana:

- $J(C)$ adalah fungsi objektif yang ingin diminimalkan (*total squared error*).
- C_j adalah *centroid* dari *cluster* j .
- x_i adalah titik data yang termasuk dalam *cluster* j .
- k adalah jumlah *cluster*.

Proses K-Means terdiri dari serangkaian langkah yang melibatkan pemilihan *centroid*, pengukuran jarak, dan pembaruan posisi *centroid* berdasarkan titik data yang telah dikelompokkan. Dengan mengulangi proses ini sampai konvergensi, K-Means dapat menghasilkan pengelompokan yang efisien dari dataset. Fungsi objektif yang diminimalkan dalam K-Means, yang dinyatakan dengan persamaan di atas, menggambarkan tujuan utama dari algoritma ini, yaitu mengurangi jarak total antara titik data dan *centroid* mereka.

V. HASIL DAN PEMBAHASAN

Penelitian ini berhasil mengimplementasikan algoritma K-Means *Clustering* untuk mengelompokkan data suara berdasarkan karakteristik gender, dengan memanfaatkan fitur *Mel-Frequency Cepstral Coefficients* (MFCC). Dataset yang digunakan terdiri dari 228 rekaman suara berdurasi pendek, yang diekstraksi menjadi 13 koefisien MFCC untuk masing-masing rekaman. Setelah dilakukan pengelompokan dengan algoritma K-Means, hasilnya menunjukkan bahwa data dapat dipisahkan menjadi dua cluster utama.



Gambar 2. Distribusi Gender Berdasarkan Clustering

Cluster 0 terdiri dari suara yang diklasifikasikan sebagai suara perempuan, sedangkan *Cluster 1* terdiri dari suara yang diklasifikasikan sebagai suara laki-laki. Distribusi hasil menunjukkan bahwa sekitar 70.2% dari data yang dikelompokkan berada di *Cluster 0*, sedangkan 29.8% berada di *Cluster 1*. Ini menunjukkan bahwa data suara perempuan dalam dataset lebih dominan dibandingkan suara laki-laki.

Proses ekstraksi fitur yang menggunakan MFCC mampu menangkap perbedaan signifikan dalam frekuensi suara, terutama antara suara laki-laki yang cenderung memiliki frekuensi rendah dan suara perempuan yang memiliki

frekuensi lebih tinggi. Proses normalisasi dan *trimming* yang dilakukan pada tahap *pre-processing* juga berhasil meningkatkan kualitas data, dengan menghilangkan bagian yang tidak relevan atau mengandung noise.

Hasil yang diperoleh dari penerapan algoritma K-Means dan fitur MFCC sangat efektif dalam mengelompokkan suara berdasarkan gender. MFCC mampu menangkap karakteristik frekuensi yang relevan untuk pengenalan suara manusia, terutama perbedaan antara frekuensi rendah (suara laki-laki) dan tinggi (suara perempuan). Keunggulan MFCC terletak pada kemampuannya memetakan frekuensi suara ke dalam skala Mel yang lebih sesuai dengan persepsi manusia. Bahkan dengan hanya 13 koefisien MFCC, model mampu menangkap perbedaan signifikan antara kedua jenis suara.

Algoritma K-Means juga menunjukkan performa yang baik dalam mengelompokkan suara berdasarkan kemiripan fitur tanpa memerlukan label pada data. Meskipun K-Means sederhana, kemampuannya untuk mengelompokkan data suara sangat cocok untuk tugas seperti ini. Keberhasilan K-Means diperkuat oleh tahap *pre-processing*, seperti normalisasi dan *trimming*, yang memastikan data suara bersih dari noise atau keheningan, serta menjaga semua data berada dalam rentang yang seragam agar tidak ada distorsi sinyal.

Distribusi cluster menunjukkan dominasi suara perempuan sebesar 70.2% dibandingkan laki-laki 29.8%, yang mungkin mencerminkan komposisi dataset. Meski data tidak seimbang, metode ini tetap konsisten dalam mengelompokkan suara berdasarkan gender.

Lebih lanjut, metode yang digunakan dalam penelitian ini dapat diterapkan pada berbagai aplikasi, seperti pengenalan suara berbasis gender, sistem keamanan berbasis suara, dan aplikasi lain yang memerlukan identifikasi gender berdasarkan suara. Keberhasilan K-Means dalam menangani dataset suara dengan presisi yang cukup baik juga menunjukkan potensi algoritma ini untuk diaplikasikan dalam tugas-tugas klasifikasi tanpa pengawasan lainnya, terutama dalam konteks data yang kompleks seperti data audio.

VI. KESIMPULAN

Penelitian ini berhasil mengimplementasikan kombinasi Mel-Frequency Cepstral Coefficients (MFCC) dan K-Means *Clustering* untuk mengelompokkan suara berdasarkan gender. Algoritma K-Means efektif dalam memisahkan suara laki-laki dan perempuan tanpa membutuhkan label awal, dengan hasil clustering menunjukkan 70.2% suara perempuan dan 29.8% suara laki-laki. Ekstraksi fitur MFCC yang meniru persepsi manusia terhadap frekuensi suara terbukti efektif dalam menangkap karakteristik penting suara. Tahap *pre-processing* yang meliputi normalisasi dan *trimming* juga berkontribusi signifikan dalam meningkatkan akurasi pengelompokan. Hasil ini menunjukkan potensi besar dalam aplikasi pengenalan suara berbasis gender.

REFERENCES

- [1] E. Tikasni, E. Utami, and D. Ariatmanto, "Analisis Akurasi Object Detection Menggunakan Tensorflow Untuk Pengenalan Bahasa Isyarat Tangan Menggunakan Metode SSD," *JURNAL FASILKOM*, vol. 14, pp. 385–393, 2024.

- [2] V. Karenina, M. F. Erinsyah, and D. S. Wibowo, "Klasifikasi Rentang Usia Dan Gender Dengan Deteksi Suara Menggunakan Metode Deep Learning Algoritma Cnn (Convolutional Neural Network)," *Komputika : Jurnal Sistem Komputer*, vol. 12, no. 2, pp. 75–82, Sep. 2023, doi: 10.34010/komputika.v12i2.10516.
- [3] A. J. Dhruv, R. Patel, and N. Doshi, "Python: The Most Advanced Programming Language for Computer Science Applications," pp. 292–299, 2022, doi: 10.5220/0010307900003051.
- [4] S. A. Alazawi, N. M. Hussien, Y. M. Mohialden, and M. A. Mohammed, "A proposed software for controlling operating system-dependent functionality," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 532–538, Feb. 2023, doi: 10.11591/eei.v12i1.4536.
- [5] N. Rama, S. Khadifa, K. Hikam, M. T. Aditya, and P. R. Wibowo, "Analisis Penerapan Python Dalam Perhitungan Limit Fungsi Pada Matematika," *Algoritma dan Sains*, vol. 1, no. 1, 2023, [Online]. Available: <https://ojs.jurnalmahasiswa.com/ojs/index.php/newton>
- [6] S. M. Elghamrawy and S. Edin Ibrahim, "Audio Signal Processing and Musical Instrument Detection using Deep Learning Techniques," in *Proceedings of the 2021 International Japan-Africa Conference on Electronics, Communications, and Computations, JAC-ECC 2021*, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 146–149. doi: 10.1109/JAC-ECC54461.2021.9691427.
- [7] P. C. Kelulusan Siswa Dengan Pendekatan Algoritma, P. Sman, C. Selatan, P. Nabila Amir, and M. Fatchan, "Prediction of Student Graduation Using the C5.0 Algorithm Approach at SMAN 2 South Cikarang," *Jurnal Ilmiah Intech : Information Technology Journal of UMUS*, vol. 5, no. 01, pp. 19–31, 2023.
- [8] G. Putri, I. Rani, A. Aziz, M. Priyono, and T. Sulistyono, "IMPLEMENTASI EUCLIDEAN DAN CHEBYSHEV DISTANCE PADA K-MEDOIDS CLUSTERING," 2022.
- [9] K. Mohammed Rezaul *et al.*, "Enhancing Audio Classification Through MFCC Feature Extraction and Data Augmentation with CNN and RNN Models," 2024. [Online]. Available: www.ijacsa.thesai.org
- [10] M. H. Abdurrohman, E. Haerani, F. Syafria, and L. Oktavia, "IMPLEMENTASI K-MEANS CLUSTERING PADA DATA PENGELOMPOKAN PENDAFTARAN MAHASISWA BARU (STUDI KASUS UNIVERSITAS ABDURRAB)," *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 9, no. 1, pp. 138–147, Jan. 2024, doi: 10.36341/rabit.v9i1.4255.
- [11] M. Ahmed, R. Seraj, and S. M. S. Islam, "The k-means algorithm: A comprehensive survey and performance evaluation," Aug. 01, 2020, *MDPI AG*. doi: 10.3390/electronics9081295.
- [12] G. Sharma, K. Umopathy, and S. Krishnan, "Trends in audio signal feature extraction methods," *Applied Acoustics*, vol. 158, Jan. 2020, doi: 10.1016/j.apacoust.2019.107020.
- [13] M. Hameed, G. Vitagliano, F. Panse, and F. Naumann, "TASHEEH: Repairing Row-Structure in Raw CSV Files," in *Advances in Database Technology - EDBT*, OpenProceedings.org, Mar. 2024, pp. 426–439. doi: 10.48786/edbt.2024.37.