Analisis dan Visualisasi Data Suara Menggunakan Python untuk Klasifikasi

**Firdis Firnandi¹, Nur Muhammad Ridho², Muhammad Hafizh³Saif Abdillah4**

¹²³4Teknik Informatika, Universitas Darussalam Gontor, Ponorogo, Indonesia

¹firdisfirnandi@student.cs.unida.gontor.ac.id ²nurassyauqi57@student.cs.unida.gontor.ac.id ³muhammadhafizh@student.cs.unida.gontor.ac.id 4saifabdillah@student.cs.unida.gontor.ac.id

**Abstrak**

pemrograman Python. Tahapan yang dilakukan meliputi data preprocessing, analisis deskriptif, serta visualisasi untuk memahami karakteristik dataset. Proses preprocessing dilakukan guna membersihkan dan menyiapkan data agar lebih representatif untuk analisis lanjutan. Selanjutnya, digunakan pustaka Python seperti Pandas, Matplotlib, dan Seaborn untuk melakukan eksplorasi serta menampilkan pola dari data.

Hasil penelitian menunjukkan bahwa visualisasi dapat membantu dalam mengidentifikasi tren, distribusi, dan hubungan antar variabel pada dataset suara. Dengan pendekatan ini, diperoleh pemahaman lebih baik terkait struktur data suara yang dapat digunakan sebagai dasar dalam tahap pemodelan dan klasifikasi selanjutnya.

**Kata kunci**: Data suara, Preprocessing, Visualisasi, Analisis data, Python.

#### I. PENDAHULUAN

**1.1 Latar Belakang**

Pemrograman komputer menjadi salah satu fondasi penting dalam perkembangan teknologi informasi. Dengan bahasa Python, mahasiswa dapat mengimplementasikan logika pemrograman untuk memecahkan permasalahan nyata. Dalam Ujian Akhir Semester (UAS) ini, diberikan sebuah tugas untuk membuat program sederhana yang dapat melakukan pengolahan data sekaligus menampilkan hasil keluarannya.

Python dipilih karena memiliki sintaks yang sederhana, fleksibel, dan mendukung berbagai pustaka yang memudahkan proses pengolahan data, baik numerik maupun teks. Program yang dibuat dalam tugas ini berfokus pada **(jelaskan sesuai kodenya, misalnya: perhitungan nilai mahasiswa, sistem sederhana, dsb.)**, serta dilengkapi dengan hasil uji coba (output) agar dapat diverifikasi kebenarannya.

**1.2 Rumusan Masalah**

Berdasarkan latar belakang tersebut, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana merancang sebuah sistem rekomendasi sederhana berbasis *user-based collaborative filtering* yang mampu memberikan saran produk kepada pengguna?
2. Bagaimana mengolah data rating pengguna terhadap produk agar dapat menjadi dasar perhitungan kemiripan antar pengguna (*similarity*)?
3. Bagaimana menampilkan hasil rekomendasi produk dalam bentuk yang mudah dipahami oleh pengguna, baik dari sisi algoritma maupun output program?

4. Sejauh mana metode *collaborative filtering* dapat meningkatkan pengalaman pengguna dalam memilih produk yang sesuai dengan preferensi mereka?

**1.3 Tujuan Penelitian**

Tujuan dari proyek ini adalah:

1. Menguji pemahaman mahasiswa dalam menyusun algoritma.
2. Mengimplementasikan logika perulangan, percabangan, dan struktur data.
3. Menampilkan hasil program yang sesuai dengan perintah.
4. Memberikan pengalaman praktis dalam membuat sistem sederhana menggunakan Python.

#### II. TINJAUAN PUSTAKA

**2.1. Pengolahan Sinyal Digital (Digital Signal Processing)**

Pengolahan Sinyal Digital (PSD atau DSP) merupakan bidang ilmu yang berfokus pada representasi, manipulasi, serta transformasi sinyal dalam bentuk digital. Menurut Oppenheim dan Schafer (2010), DSP memungkinkan sinyal analog seperti audio, citra, maupun data sensor diubah menjadi bentuk digital agar dapat diproses secara matematis menggunakankomputer.

Keunggulan utama dari DSP dibandingkan pengolahan sinyal analog adalah fleksibilitasnya. Dengan DSP, kita dapat melakukan filtering, kompresi, analisis spektrum, hingga noise reduction dengan tingkat akurasi yang tinggi dan dapat disesuaikan dengan kebutuhan pengguna.

**2.2 Konsep Filtering pada Sinyal Audio**

Filtering adalah proses pemrosesan sinyal yang bertujuan untuk melewatkan frekuensi tertentu dan meredam frekuensi lainnya. Menurut Ifeachor dan Jervis (2002), filter dalam domain digital dibagi menjadi dua kategori utama, yaitu:

* **Finite Impulse Response (FIR):** Filter dengan respons impuls berhingga yang stabil dan tidak bergantung pada kondisi awal.
* **Infinite Impulse Response (IIR):** Filter dengan respons impuls tak berhingga yang lebih efisien secara komputasi namun memiliki risiko instabilitas.

Dalam konteks audio, filter berperan penting dalam mengurangi noise, menekankan frekuensi tertentu (misalnya bass atau treble), serta meningkatkan kualitas suara. Jenis filter yang umum digunakan antara lain **low-pass filter, high-pass filter, band-pass filter,** dan **band-stop filter**.

**2.3. Aplikasi Filtering pada Audio**

Menurut Mitra (2006), filtering pada sinyal audio dapat diterapkan untuk berbagai keperluan, seperti:

* **Noise Reduction:** Menghilangkan gangguan suara dari rekaman audio.
* **Equalization:** Menyesuaikan proporsi frekuensi agar suara terdengar lebih seimbang.
* **Audio Enhancement:** Memperjelas suara tertentu, misalnya vokal pada lagu atau suara instruksi pada rekaman.

Dalam penelitian dan pengembangan sistem berbasis machine learning maupun analisis data audio, filtering sering digunakan sebagai tahap **preprocessing** sebelum sinyal dimasukkan ke model analitik.

#### III. METODOLOGI PENELITIAN

**3.1. Jenis Penelitian**

Penelitian ini bersifat **eksperimen kuantitatif**, di mana dilakukan serangkaian uji coba terhadap sinyal audio menggunakan algoritma filtering digital. Eksperimen dilakukan untuk mengetahui bagaimana kinerja **filter Butterworth** dengan berbagai konfigurasi (low-pass, high-pass, dan band-pass) dalam memproses file audio.

**3.2. Tahapan Penelitian**

Metodologi penelitian ini terdiri dari beberapa tahap utama sebagai berikut:

**a. Studi Literatur**

Pada tahap awal dilakukan pengumpulan referensi terkait teori dasar **Pengolahan Sinyal Digital (Digital Signal Processing/DSP)**, konsep **filter digital (FIR & IIR)**, serta implementasinya pada sinyal audio. Literatur diambil dari buku teks, jurnal internasional, artikel ilmiah, serta dokumentasi library Python seperti **Librosa** dan **SciPy**.

**b. Perancangan Sistem**

Sistem dirancang dalam bentuk aplikasi interaktif berbasis **Kaggle Notebook** dengan memanfaatkan library Python. Komponen utama yang dirancang antara lain:

* **Input**: File audio dalam format .wav, .mp3, .flac, dan sejenisnya.
* **Proses**: Implementasi filter Butterworth dengan parameter *cutoff frequency* dan *filter type*.
* **Output**: Visualisasi bentuk gelombang (waveform), spektrogram audio, serta hasil audio setelah difilter

.

**c. Implementasi Program**

Kode program dikembangkan dalam lingkungan Kaggle Notebook dengan struktur sebagai berikut:

1. **Inisialisasi Library**: Import library yang dibutuhkan (librosa, matplotlib, scipy.signal, ipywidgets).
2. **Pengolahan File Audio**: Membaca file audio menggunakan fungsi librosa.load().
3. **Fungsi Filtering**: Implementasi filter Butterworth menggunakan scipy.signal.butter() dan scipy.signal.lfilter().
4. **Visualisasi**: Menampilkan waveform dan spektrogram menggunakan matplotlib dan librosa.display.
5. **Antarmuka Interaktif**: Dibangun dengan ipywidgets agar pengguna dapat memilih filter, cutoff, dan langsung melihat hasil audio.

**d. Pengujian Sistem**

Pengujian dilakukan dengan cara:

* Menggunakan beberapa file audio berbeda (musik, suara manusia, suara lingkungan).
* Menerapkan filter dengan variasi cutoff frequency dan jenis filter (low-pass, high-pass, band-pass).
* Membandingkan hasil audio sebelum dan sesudah filtering melalui visualisasi waveform, spektrogram, serta playback audio.

**e. Evaluasi Hasil**

Evaluasi dilakukan dengan pendekatan kualitatif dan kuantitatif:

* **Kualitatif**: Mendengarkan hasil audio untuk menilai kejernihan, kualitas suara, serta seberapa efektif filter mengurangi noise atau frekuensi tertentu.
* **Kuantitatif**: Menganalisis perubahan spektrum frekuensi dengan membandingkan spektrogram sebelum dan sesudah filtering.

**3.3 Alur Penelitian**

Alur penelitian secara umum dapat digambarkan sebagai berikut:

1. Identifikasi masalah → 2. Studi pustaka → 3. Perancangan aplikasi → 4. Implementasi kode di Kaggle Notebook → 5. Uji coba dengan berbagai audio → 6. Analisis hasil → 7. Penarikan kesimpulan.

#### V. KESIMPULAN

Berdasarkan hasil implementasi dan analisis pada proyek *FINAL PROJECT PSD – Aplikasi Filtering Audio untuk Kaggle Notebook*, dapat disimpulkan beberapa poin penting:

1. **Pemrosesan sinyal digital (DSP) terbukti efektif** dalam mengurangi noise pada file audio dengan menerapkan metode filtering, khususnya filter *high-pass* dan *low-pass*.
2. Proses filtering berhasil meningkatkan kualitas audio, yang ditunjukkan oleh spektrum frekuensi sebelum dan sesudah filtering, di mana frekuensi noise berkurang secara signifikan.
3. Implementasi pada **Kaggle Notebook** memberikan kemudahan dalam pengolahan data berbasis Python karena mendukung integrasi dengan pustaka seperti *NumPy*, *SciPy*, dan *Matplotlib*.
4. Hasil eksperimen menunjukkan bahwa pemilihan parameter filter (cut-off frequency, order filter) sangat mempengaruhi hasil akhir. Oleh karena itu, pemilihan parameter harus disesuaikan dengan karakteristik sinyal audio yang digunakan.
5. Proyek ini tidak hanya membuktikan manfaat filtering dalam konteks akademis, tetapi juga menunjukkan potensi aplikasinya di bidang **multimedia, komunikasi, hingga machine learning** yang membutuhkan data audio dengan kualitas lebih baik.

Dengan demikian, filtering audio melalui pendekatan DSP dapat menjadi solusi praktis dalam meningkatkan kualitas data audio, sekaligus memperkuat pemahaman mahasiswa terhadap konsep teori dan implementasi pemrosesan sinyal digital.

#### VI. DAFTAR PUSTAKA

[1] Oppenheim, A. V., & Schafer, R. W. (2010). *Discrete-Time Signal Processing* (3rd ed.). Pearson.

[2] Proakis, J. G., & Manolakis, D. G. (2006). *Digital Signal Processing: Principles, Algorithms, and Applications* (4th ed.). Pearson Prentice Hall.

[3] Smith, S. W. (2003). *Digital Signal Processing: A Practical Guide for Engineers and Scientists*. Newnes.