Implementasi Pembersihan Data untuk Analisis Tur Konser Terlaris Menggunakan Pandas dan NumPy

452024611053, Farrel Ghozy Affifudin, *farrelghozyaffifudin33@student.cs.unida.gontor.ac.id*, Universitas Darussalam Gontor - Teknik Informatika

Abstract - Kualitas data merupakan fondasi penting dalam analisis data dan *machine learning*. Dataset yang kotor, tidak lengkap, atau tidak konsisten dapat menghasilkan kesimpulan yang salah. Laporan ini merinci proses pembersihan data (*data cleaning*) pada dataset data_kotor.csv yang berisi informasi tur konser terlaris. Tantangan utama dalam dataset ini meliputi adanya data duplikat , *missing values* dalam jumlah signifikan, dan tipe data yang tidak konsisten pada kolom numerik. Metode pembersihan data diimplementasikan menggunakan *library* Pandas dan NumPy. Proses ini mencakup penghapusan data duplikat, penghapusan kolom yang tidak relevan, konversi tipe data, dan penanganan *missing values* menggunakan teknik imputasi *median* (untuk data numerik) dan *mode* (untuk data kategorikal). Hasil akhir dari proses ini adalah dataset baru bernama student_scores_cleaned.csv yang memiliki 20 entri unik dan bebas dari *missing values*, sehingga siap untuk analisis lebih lanjut.

Keywords—Data Cleaning, Pandas, NumPy, Missing Values, Data Duplication, Imputation

I. Pendahuluan

Dalam era *data-driven*, kuantitas data yang besar seringkali tidak diimbangi dengan kualitas data yang baik. Prinsip "Garbage In, Garbage Out" (GIGO) sangat relevan dalam *machine learning* dan analisis data. Data yang kotor—mengandung kesalahan, nilai yang hilang (missing values), atau duplikasi—dapat secara drastis menurunkan akurasi model dan mengarahkan pada pengambilan keputusan yang keliru. Oleh karena itu, pembersihan data (data cleaning) adalah salah satu langkah krusial dalam *data preprocessing*.

Tugas ini berfokus pada penerapan teknik-teknik dasar pembersihan data pada dataset data_kotor.csv, yang mencatat data historis tur konser terlaris. Analisis awal pada dataset ini (seperti yang ditunjukkan pada *cell 2* file notebook) mengidentifikasi beberapa masalah utama:

1. **Data Duplikat:** Terdapat 1 baris data yang terduplikasi secara identik.
2. **Missing Values:** Kolom 'Peak' dan 'All Time Peak' memiliki jumlah *missing values* yang substansial (masing-masing 12 dan 15 dari 21 data).
3. **Tipe Data Tidak Konsisten:** Kolom finansial ('Actual gross', 'Adjusted gross', 'Average gross') dan kolom peringkat ('Peak', 'All Time Peak') disimpan sebagai `object` (teks) karena mengandung karakter non-numerik seperti `$` , `,` , dan `[..]`, padahal seharusnya bertipe numerik.

Studi ini bertujuan untuk menerapkan proses pembersihan data secara sistematis untuk mengatasi masalah-masalah tersebut. Sesuai batasan tugas, implementasi hanya menggunakan *library* Pandas dan NumPy. Laporan ini akan menjelaskan metodologi yang digunakan, menyajikan hasil sebelum dan sesudah pembersihan, serta mendiskusikan dampak dari setiap langkah pembersihan terhadap kualitas dataset .

II. Metodhologi

Proses pembersihan data dilakukan melalui beberapa tahapan yang terstruktur, dimulai dari pemuatan data hingga penyimpanan data bersih.

A. **Pemuatan dan Analisis Awal** Dataset data_kotor.csv dimuat ke dalam DataFrame Pandas menggunakan pd.read_csv() [cell 2]. Analisis awal dilakukan menggunakan fungsi df.info() dan df.duplicated().sum() [cell 2] untuk mengidentifikasi jumlah total data, tipe data setiap kolom, jumlah *missing values*, dan jumlah baris duplikat.

B. **Penghapusan Kolom Tidak Relevan** Kolom 'Ref.' diidentifikasi sebagai kolom yang hanya berisi data sitasi (misalnya [1], [3]) [cell 2 output] dan tidak memiliki nilai prediktif atau analitis. Oleh karena itu, kolom ini dihapus menggunakan df.drop(columns=['Ref.']) [cell 3] untuk mengurangi kompleksitas dataset.

C. **Pembersihan dan Konversi Tipe Data** Beberapa kolom yang seharusnya numerik terdeteksi sebagai object. Prosedur konversi berikut diterapkan:

    a. **Kolom Peringkat ('Peak', 'All Time Peak'):** Kolom ini berisi teks seperti 7[2] [cell 2 output]. Untuk membersihkannya, nilai diubah menjadi string, kemudian dipisah (split) berdasarkan karakter [dan hanya bagian pertama (angka) yang diambil. Hasilnya kemudian dikonversi menjadi tipe data numerik menggunakan pd.to_numeric [cell 3].

    b. **Kolom Mata Uang ('Actual gross', 'Adjusted gross', 'Average gross'):** Kolom ini berisi karakter $ dan , [cell 2 output]. Skrip pembersihan dirancang untuk menghapus kedua karakter ini menggunakan .str.replace() dan kemudian mengkonversi hasilnya menjadi tipe data numerik [cell 3].

D. **Penanganan Data Hilang (Missing Value Imputation)** Setelah konversi tipe data, *missing values* (yang baru muncul dari *string* yang tidak valid atau yang sudah ada sebelumnya) ditangani menggunakan strategi imputasi [cell 3]:

    a. **Kolom Numerik:** Untuk semua kolom dengan tipe data numerik (seperti 'Peak', 'All Time Peak', dan kolom mata uang), *missing values* (NaN) diisi menggunakan nilai **median** dari kolom tersebut. Median dipilih menggantikan *mean* agar lebih robust (tidak sensitif) terhadap *outliers* potensial dalam data finansial.

    b. **Kolom Kategorikal/Object:** Untuk kolom object, *missing values* diisi menggunakan **mode** (nilai yang paling sering muncul) dari kolom tersebut.

E. **Penanganan Data Duplikat** Setelah semua nilai dibersihkan dan diimputasi, data duplikat ditangani. Berdasarkan analisis awal, ditemukan 1 baris duplikat [cell 2 output]. Baris ini dihapus menggunakan df.drop_duplicates(keep='first') [cell 3], sehingga setiap entri dalam dataset bersifat unik.

III. Hasil

Proses pembersihan data secara signifikan mengubah struktur dan kualitas dataset. Perbandingan status dataset sebelum dan sesudah pembersihan dirangkum dalam Tabel I.

TABEL I. PERBANDINGAN DATASET SEBELUM DAN SESUDAH CLEANING | Atribut | Sebelum Pembersihan [cell 2 output] | Sesudah Pembersihan [cell 4 output] | | :--- | :--- | :--- | | Jumlah Baris | 21 | 20 | | Jumlah Duplikat | 1 | 0 | | Missing Values ('Peak') | 12 | 0 | | Missing Values ('All Time Peak')| 15 | 0 | | Tipe Data ('Peak') | object | float64 | | Tipe Data ('All Time Peak')| object | float64 | | Tipe Data

('Actual gross') | object | object | | Tipe Data ('Average gross') | object | int64 |

Cuplikan data setelah proses pembersihan (output dari df_clean.head() [cell 4]) menunjukkan bahwa data telah terstruktur, meskipun kolom Actual gross dan Adjusted gross masih menampilkan format aslinya.

```
  --- DATA AKHIR (SETELAH DIBERSIHKAN) ---
    Rank  Peak  All Time Peak  Actual gross Adjusted gross (in 2022 dollars)  \
 0     1   1.0           2.0  $780,000,000                    $780,000,000
 1     2   1.0           7.0  $579,800,000                    $579,800,000
 2     3   1.0           2.0  $411,000,000                    $560,622,615
 3     4   2.0          10.0  $397,300,000                    $454,751,555
 4     5   2.0           8.5  $345,675,146                    $402,844,849
```

Gbr. 1. Cuplikan 5 baris data teratas setelah pembersihan.

Validasi akhir menggunakan df_clean.info() [cell 4 output] mengkonfirmasi bahwa tidak ada lagi *missing values* di semua kolom. Namun, info() juga menunjukkan hasil konversi tipe data yang menarik, seperti yang terlihat pada Gbr. 2.

```
--- Info Akhir (Tipe Data & Missing) ---
<class 'pandas.core.frame.DataFrame'>
Index: 20 entries, 0 to 19
Data columns (total 10 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
...
 8   Shows                           20 non-null     int64
 9   Average gross                   20 non-null     int64
dtypes: float64(2), int64(3), object(5)
memory usage: 1.7+ KB
```

IV. Diskusi

Analisis hasil pembersihan data menunjukkan keberhasilan dalam beberapa aspek kritis dan menyoroti tantangan yang tersisa.

A. **Dampak Penghapusan Duplikat dan Imputasi** Penghapusan satu baris duplikat [cell 3 output] sangat penting untuk integritas statistik. Jika dibiarkan, data duplikat tersebut akan memberikan bobot ganda pada satu entri tur, yang menyebabkan bias dalam analisis rata-rata atau total pendapatan.

   Imputasi *missing values* pada kolom 'Peak' dan 'All Time Peak' (yang semula hilang lebih dari 50% datanya) dengan nilai median [cell 3] berhasil menyelamatkan kolom-kolom ini dari keharusan untuk dihapus. Kolom-kolom ini berhasil dikonversi ke `float64` (Gbr. 2) dan kini dapat digunakan dalam analisis kuantitatif.

B. **Analisis Kegagalan Konversi Tipe Data Parsial** Temuan paling menarik dari proses validasi adalah keberhasilan dan kegagalan parsial dalam konversi kolom mata uang. Seperti yang ditunjukkan pada Gbr. 2 (output `df_clean.info()`), kolom 'Average gross' berhasil dikonversi ke `int64`, namun kolom 'Actual gross' dan 'Adjusted gross (in 2022 dollars)' **tetap bertipe `object`**.

   Melihat Gbr. 1 (output `df_clean.head()`), terlihat bahwa karakter `$` dan `,` masih ada di kolom-kolom yang gagal tersebut. Hal ini mengindikasikan bahwa metode `.str.replace('$', '').str.replace(',', '')` [cell 3] tidak berhasil diterapkan pada kedua kolom tersebut.

   Hipotesis yang paling mungkin adalah adanya karakter tak terlihat (*hidden characters*) atau *non-breaking spaces* (misalnya `\xa0`) pada nama kolom atau di dalam data itu sendiri, yang tidak tertangani oleh skrip. Misalnya, nama kolom di *string* kode (contoh: `'Actual gross'`) mungkin tidak sama persis dengan nama kolom di DataFrame (contoh: `'Actual\xa0gross'`). Kegagalan ini menyebabkan data di kolom tersebut tidak pernah dikonversi menjadi numerik, sehingga pada langkah imputasi, data tersebut diperlakukan sebagai `object` dan diimputasi menggunakan *mode* (jika ada *missing values*), bukan *median*.

   Walaupun data bebas dari `NaN`, kegagalan konversi ini berarti dua kolom finansial utama tersebut belum siap untuk analisis matematis.

V. Kesimpulan

Proses pembersihan data pada dataset `data_kotor.csv` telah berhasil dilaksanakan sesuai dengan tujuan awal. Data duplikat telah dihapus, dan semua *missing values* telah ditangani melalui imputasi *median* dan *mode*. Dataset yang dihasilkan, `student_scores_cleaned.csv`, kini memiliki integritas data yang lebih baik dengan 20 entri unik.

Meskipun demikian, analisis pada tahap hasil (Gbr. 2) mengungkap bahwa konversi tipe data untuk kolom 'Actual gross' dan 'Adjusted gross' tidak berhasil, kemungkinan besar karena adanya karakter tak terlihat atau perbedaan format *string* yang tidak terduga.

Sebagai rekomendasi untuk pekerjaan di masa depan, skrip pembersihan data perlu disempurnakan. Langkah tambahan harus mencakup inspeksi karakter yang lebih mendalam (misalnya, dengan mencetak representasi *byte* dari *string*) dan menggunakan metode yang lebih robust, seperti ekspresi reguler (regex), untuk memastikan semua karakter non-numerik (termasuk *non-breaking spaces*) dapat diidentifikasi dan dihapus dengan benar sebelum konversi ke tipe data numerik.