Analisa Kode Program Pembuatan Model Data Mining

---

1. **Import Library**

   Pada bagian awal kode saya, saya mengimpor sejumlah library penting seperti pandas, numpy, seaborn, dan matplotlib.pyplot. Library pandas saya gunakan untuk mengelola dan memanipulasi data dalam bentuk tabel (dataframe), sedangkan numpy mendukung perhitungan numerik. Untuk membuat visualisasi data agar lebih mudah dipahami, saya menggunakan seaborn dan matplotlib.pyplot

2. **Membaca Dataset**

   Kode saya membaca dataset utama menggunakan fungsi pd.read_excel() dari file bernama "Data_Hoaks_2023.xlsx". File ini berisi kumpulan berita dalam Bahasa Indonesia yang dikategorikan sebagai berita hoaks atau bukan. Proses ini merupakan tahap awal yang penting untuk mempersiapkan data sebelum dilakukan pembersihan dan analisis lebih lanjut.

3. **Preprocessing Data**

   Saya melakukan preprocessing sederhana untuk membersihkan teks sebelum ditokenisasi.
   Langkah-langkahnya meliputi:

   - Case folding (mengubah semua huruf menjadi huruf kecil).
   - Normalisasi Singkatan (mengubah kata tidak baku ke dalam bentuk baku).
   - Menghapus tanda baca, angka, dan karakter khusus.

4. **Exploratory Data Analysis (EDA)**

   Setelah data dibersihkan dan sebelum model dilatih, saya melakukan Exploratory Data Analysis (EDA) untuk memahami struktur, distribusi, dan karakteristik isi teks berita dalam dataset. Tujuan EDA adalah untuk mengidentifikasi pola, ketidakseimbangan kelas, serta memberikan gambaran awal yang berguna dalam proses klasifikasi.
   Langkah-langkah EDA dalam kode saya meliputi:

   a. **Distribusi Label Kelas**

      Saya memvisualisasikan sebaran antara berita hoax dan valid menggunakan diagram batang (bar chart) untuk melihat apakah data seimbang atau tidak. Hasil visualisasi ini

menunjukkan bahwa distribusi antara kelas valid dan hoax relatif seimbang, yaitu sekitar 305 untuk berita valid dan 425 untuk hoax. Hal ini baik untuk pelatihan model karena tidak terjadi ketimpangan ekstrim antar kelas (Imbalanced Dataset).

b. Distribusi Panjang Teks

Saya menghitung panjang teks (jumlah kata) pada setiap entri untuk memastikan bahwa data tidak terlalu pendek atau terlalu panjang, serta untuk memahami variasi panjang teks yang masuk ke model. Mayoritas teks memiliki panjang antara 100 hingga 300 kata. Ini ideal untuk model IndoBERT karena tidak terlalu pendek dan tidak terlalu panjang hingga melewati batas maksimal input token (512 token).

c. WordCloud untuk Masing-Masing Kelas

Saya menggunakan WordCloud untuk menampilkan kata-kata yang paling sering muncul dalam berita hoax dan berita valid. Tujuannya adalah untuk melihat perbedaan konten dominan antar kelas. Hasil WordCloud menunjukkan bahwa berita hoax cenderung menggunakan kata-kata umum dan provokatif, seperti “tersebut”, “masyarakat”, atau “viral”. Sebaliknya, berita valid menampilkan kata-kata yang lebih spesifik dan relevan terhadap isu aktual, seperti “jokowi”, “video”, atau “ferdy sambo”.

5. Membagi Data Latih (Training) dan Data Uji (Testing)

Saya membagi data menjadi 80% untuk pelatihan dan 20% untuk pengujian menggunakan train_test_split dari scikit-learn, kemudian data dimasukkan ke dalam Dataset.

6. Tokenisasi

Setelah teks dibersihkan, saya melakukan tokenisasi menggunakan tokenizer dari IndoBERT. Tokenisasi ini mengubah teks menjadi input ID dan attention mask yang dapat diproses oleh model.

7. Membuat DataLoader

Untuk mempermudah proses pelatihan, saya menggunakan DataLoader untuk membaca data per batch.

8. Mengatur Optimizer

Saya menggunakan optimizer AdamW dengan learning rate sebesar 2e-5.

9. Pelatihan Model (Training Loop)

Saya melatih model selama 10 epoch dengan langkah-langkah:

- Menjalankan forward pass
- Menghitung loss
- Melakukan backward propagation
- Memperbarui bobot model
- Mereset gradien

10. Evaluasi Model

Setelah pelatihan selesai, saya mengevaluasi performa model terhadap data uji (Testing). Model berhasil mengklasifikasikan seluruh data uji dengan benar, tanpa kesalahan. Ini ditunjukkan oleh nilai support sebanyak 61 data valid dan 85 data hoax, yang seluruhnya terklasifikasi secara tepat.

11. Deploy Model

Setelah model dilatih dan dievaluasi, saya menyimpan model ke file .pt atau .bin untuk keperluan deployment. Model ini kemudian saya gunakan dalam aplikasi web berbasis Flask yang di deploy di PythonAnywhere.