

Nama: Guntur Eka Wardana

NIM: 1103174102

Technical Report: PyTorch Custom Datasets

Pendahuluan

Dalam laporan ini, kami akan membahas langkah-langkah penggunaan PyTorch untuk membangun dan melatih model menggunakan dataset kustom. Setiap langkah kunci dari impor PyTorch hingga pembuatan model dan evaluasi akan dianalisis secara mendalam.

0. Mengimpor PyTorch dan Menyiapkan Kode Perangkat-Agnostik

Pertama-tama, PyTorch diimpor untuk digunakan dalam pembuatan model. Selanjutnya, kode disesuaikan agar dapat berjalan di berbagai perangkat, mendeteksi ketersediaan GPU, dan secara otomatis beralih ke CPU jika perlu.

1. Mendapatkan Data

Langkah awal adalah memperoleh dataset kustom, yang merupakan subset dari dataset Food101. Dataset ini berfokus pada gambar pizza, steak, dan sushi, memberikan landasan kecil sebelum memperluas dataset atau model.

2. Menjadi Satu dengan Data (Persiapan Data)

Pemahaman mendalam tentang struktur dan karakteristik dataset merupakan langkah kunci dalam machine learning. Proses pembersihan, penyesuaian skala, dan pemformatan data dilakukan untuk memastikan kualitas data sebelumnya memasuki model.

2.1. Visualisasi Gambar

Visualisasi gambar dari dataset memungkinkan pemahaman visual yang lebih baik terhadap data. Fungsi visualisasi PyTorch atau matplotlib digunakan untuk menampilkan contoh gambar dari dataset.

3. Transformasi Data

Transformasi data melibatkan perubahan data mentah menjadi bentuk yang lebih sesuai untuk pembelajaran mesin. Ini dapat mencakup normalisasi, rotasi, atau perubahan lainnya tergantung pada format atau struktur data asli.

4. Opsi 1: Memuat Data Gambar dengan ImageFolder

ImageFolder dari PyTorch digunakan untuk memuat data gambar. Struktur folder sesuai dengan kelas atau label gambar, memanfaatkan kemudahan PyTorch dalam memuat dataset dan menerapkan transformasi pada setiap gambar.

5. Opsi 2: Memuat Data Gambar dengan Dataset Kustom

Jika PyTorch tidak menyediakan fungsi bawaan, pengguna dapat membuat kelas Dataset kustom. Hal ini memberikan fleksibilitas lebih besar dalam memuat dan memproses data sesuai dengan kebutuhan.

6. Bentuk-Bentuk Lain dari Transformasi (Augmentasi Data)

Pendekatan augmentasi data dieksplorasi menggunakan fungsi transformasi PyTorch bawaan. Augmentasi data meningkatkan variasi data pelatihan dan membantu model mengatasi beragam situasi.

7. Model 0: TinyVGG tanpa Augmentasi Data

Model dasar TinyVGG dibangun tanpa menggunakan augmentasi data. Fungsi pelatihan dan evaluasi model diterapkan untuk melihat kinerja dasar model.

8. Mengeksplorasi Kurva Kerugian

Kurva kerugian dieksplorasi untuk memahami bagaimana model belajar dari data. Evaluasi dilakukan terhadap apakah model cenderung underfitting atau overfitting.

9. Model 1: TinyVGG dengan Augmentasi Data

Model serupa dengan Model 0 dibangun, tetapi kali ini dengan penggunaan augmentasi data. Dengan menerapkan teknik augmentasi pada data pelatihan, tujuannya adalah meningkatkan kinerja dan generalisasi model.

10. Membandingkan Hasil Model

Hasil dari kedua model dibandingkan untuk mengevaluasi performa masing-masing. Pilihan dan diskusi diberikan untuk meningkatkan performa model.

11. Membuat Prediksi pada Gambar Kustom

Model yang telah dilatih diterapkan pada gambar kustom di luar dataset pelatihan. Proses ini menggambarkan penggunaan model untuk tujuan prediksi. Dengan memahami setiap langkah secara mendalam, kita dapat membangun, melatih, dan mengevaluasi model machine learning dengan lebih baik menggunakan PyTorch.