

Nama: Pratyenggo Damar I.P.

NIM: 1103194141

Technical Report: PyTorch Going Modular

Pendahuluan

Dalam eksplorasi PyTorch Going Modular, langkah-langkah penting dalam membangun dan mengevaluasi model machine learning (ML) dijelaskan secara rinci. Fokusnya terletak pada penggunaan PyTorch untuk mempersiapkan dataset, transformasi data, membangun arsitektur model, pelatihan, pengujian, evaluasi kinerja, dan penyimpanan model.

1. Persiapan Awal

Pertama, dilakukan impor semua dependensi yang diperlukan, seperti PyTorch, os, dan lainnya. Selanjutnya, penentuan mode sel sebagai notebook atau script dilakukan untuk menyesuaikan alur kerja sesuai kebutuhan.

2. Pengambilan Data

Dataset gambar kustom dengan kategori pizza, steak, dan sushi diunduh dari GitHub. Penting untuk memastikan struktur dataset telah diunduh dan terorganisir dengan baik sebelum melanjutkan.

3. Persiapan Data

Struktur dataset dieksplorasi untuk memastikan ketersediaan dan keorganisasian yang tepat. Folder penyimpanan dataset diperiksa, dan jika belum ada, data diunduh dan diekstraksi ke folder yang sesuai.

4. Transformasi Data

Proses ini melibatkan penyesuaian ukuran gambar agar seragam, normalisasi nilai pixel, dan konversi gambar menjadi tensor menggunakan torchvision.transforms.

5. Memuat Data dengan ImageFolder (Opsi 1)

ImageFolder dari PyTorch digunakan untuk memuat data jika dataset mengikuti struktur standar.

6. Memuat Data Gambar dengan Dataset Kustom

Jika struktur dataset tidak standar, kelas dataset kustom dibuat untuk memberikan fleksibilitas dalam mengelola dataset dengan struktur yang kompleks.

7. Transformasi Lainnya (Augmentasi Data)

Augmentasi data diterapkan untuk meningkatkan variasi dataset pelatihan. Opsi transformasi tambahan seperti rotasi dieksplorasi.

8. Model 0: TinyVGG tanpa Augmentasi Data

TinyVGG, model konvolusional sederhana, diimplementasikan bersama dengan fungsi pelatihan dan evaluasi.

9. Eksplorasi Kurva Loss

Kurva loss dipantau untuk mengevaluasi kinerja model dan mendeteksi overfitting atau underfitting.

10. Model 1: TinyVGG dengan Augmentasi Data

Versi lain dari TinyVGG dibuat dengan penambahan augmentasi data.

11. Membandingkan Hasil Model

Performa kedua model dibandingkan untuk mengevaluasi efektivitas augmentasi data. Diskusi tentang peningkatan model diajukan.

12. Membuat Prediksi pada Gambar Kustom

Model yang telah dilatih diimplementasikan untuk membuat prediksi pada gambar kustom.

13. Penyimpanan Model

Modul utilitas untuk menyimpan model saat pelatihan dibuat. Semua komponen sebelumnya digabungkan dalam skrip **train.py**.

14. Train.py - Penggabungan Semua Komponen

Skrip ini mencakup semua langkah sebelumnya, termasuk pengaturan hyperparameter, inisialisasi model, pelatihan, dan penyimpanan model yang telah dilatih.

Kesimpulan

Dengan mengikuti langkah-langkah ini, PyTorch Going Modular memberikan panduan komprehensif untuk mempersiapkan data, membangun model, dan melatihnya. Dengan fokus pada modularitas, laporan ini mengilustrasikan bagaimana membangun fondasi yang kuat untuk pengembangan model machine learning menggunakan PyTorch.