

Zero to Mastery Learn PyTorch for Deep Learning Course Summary

04. PyTorch Custom Datasets

Modul ini membahas cara membuat dan menggunakan dataset kustom dalam PyTorch untuk melakukan pemodelan visi komputer. Kita akan belajar langkah-langkah seperti mengimpor dan mempersiapkan data, transformasi data, pembuatan dataset kustom, dan pengembangan model menggunakan dataset tersebut.

1. Mendapatkan Data

Pertama-tama, kita membutuhkan data.

Gunakan dataset Food101 yang berisi 101.000 gambar makanan dari 101 jenis makanan.

Untuk latihan, kita gunakan subset dataset tersebut, yaitu 10% dari 3 kelas (pizza, steak, sushi).

Data disusun dalam format klasifikasi gambar, terstruktur dalam direktori kelas masing-masing.

Unduh data dari GitHub dan persiapkan menggunakan Python.

2. Memahami Data (Persiapan Data)

Tahap ini melibatkan pemahaman struktur dan karakteristik data sebelum membangun model.

Data terdiri dari gambar pizza, steak, dan sushi, disimpan dalam direktori terpisah.

Fungsi dibuat untuk mengeksplorasi jumlah direktori dan gambar di setiap direktori.

3. Transformasi Data

Diperlukan untuk mempersiapkan data gambar sebelum digunakan oleh PyTorch.

Gunakan `torchvision.transforms` untuk mengubah gambar menjadi tensor dan memberikan transformasi seperti `resize` dan `flip horizontal`.

Buat fungsi untuk memvisualisasikan transformasi yang diterapkan pada beberapa gambar.

4. Memuat Data Gambar Menggunakan ImageFolder

Gunakan `torchvision.datasets.ImageFolder` untuk mengonversi data gambar menjadi Dataset PyTorch.

Transformasi data diterapkan saat membuat Dataset.

Inisialisasi dataset pelatihan dan pengujian menggunakan fungsi `ImageFolder` dan tampilkan informasinya.

Tampilkan contoh gambar dan label dari dataset.

5. Opsi 2 - Memuat Data Gambar dengan Dataset Kustom

Jika kita memerlukan dataset kustom, kita dapat membuatnya sendiri.

Pro: Dapat membuat dataset dari hampir segala sesuatu.

Kontra: Memerlukan penulisan kode lebih banyak.

Langkah-langkah melibatkan mengimpor modul, membuat fungsi bantu, dan mengimplementasikan kelas Dataset kustom.

6. Bentuk Lain dari Transformasi (Augmentasi Data)

Transformasi data dapat digunakan untuk mengubah gambar dengan berbagai cara, termasuk augmentasi data.

Contoh transformasi data untuk pelatihan dan pengujian mencakup augmentasi data pada set pelatihan dan konversi gambar menjadi tensor.

7. Model 0: TinyVGG tanpa augmentasi data

Membuat transformasi sederhana dan memuat data untuk model.

Membuat model TinyVGG dengan lapisan konvolusi dan pengkitaan.

Melakukan forward pass pada satu gambar untuk pengujian model.

Menggunakan torchinfo untuk mendapatkan gambaran bentuk yang dilalui oleh model.

Membuat fungsi loop pelatihan dan pengujian.

8. Evaluasi Model 0 dan Strategi untuk Memperbaikinya

Beberapa strategi untuk meningkatkan kinerja model melibatkan penanganan overfitting dan underfitting.

Strategi untuk overfitting: Dapatkan lebih banyak data, sederhanakan model, gunakan augmentasi data, transfer learning, dropout layers, learning rate decay, dan early stopping.

Strategi untuk underfitting: Tambahkan lebih banyak lapisan atau unit ke model, sesuaikan learning rate, gunakan transfer learning, latih model lebih lama, kurangi regularisasi.

9. Model 1 - TinyVGG dengan Augmentasi Data

Membuat transformasi data dengan augmentasi.

Membuat dataset dan dataloader.

Membangun dan melatih model TinyVGG dengan augmentasi data.

Plot kurva kerugian model 1.

10. Membandingkan Hasil Model

Ubah hasil model menjadi DataFrame.

Bandingkan hasil model menggunakan grafik.

11. Membuat Prediksi pada Gambar Kustom

Memuat gambar kustom dengan PyTorch.

Memprediksi gambar kustom dengan model PyTorch yang telah dilatih.

Membuat fungsi untuk prediksi dan plot gambar.

Ihsan Ridho Asmoro
1103204184

Dengan memahami dan menerapkan langkah-langkah ini, kita dapat membuat dataset kustom, melatih model, dan meningkatkan kinerjanya sesuai kebutuhan.