

## Technical Report Chapter PyTorch



Oleh:

**Khairi Hibatullah Ridho/1103228240**

**PRODI S1 TEKNIK KOMPUTER  
FAKULTAS TEKNIK ELEKTRO  
UNIVERSITAS TELKOM  
BANDUNG**

## **Pendahuluan**

Laporan ini mendokumentasikan penerapan PyTorch, sebuah framework penting dalam deep learning, dalam serangkaian latihan terstruktur. Tujuan laporan ini adalah untuk menguraikan berbagai aspek pembelajaran mesin, mulai dari manipulasi data hingga penyebaran model, dengan menggunakan PyTorch.

### **00. PyTorch Fundamentals Exercises**

Fokus awal pada modul `torch.Tensor` dan `torch.cuda` membuka jalan bagi eksplorasi lebih dalam mengenai pemrograman tensor dan komputasi paralel. Latihan dalam menciptakan tensor acak dan penerapan operasi dasar seperti transpose dan perkalian matriks memberikan pemahaman praktis mengenai manipulasi data dalam tensor. Pentingnya pengaturan seed acak untuk konsistensi hasil, terutama dalam komputasi menggunakan GPU, menjadi kunci dalam eksperimen ini. Operasi matematis lanjutan seperti pencarian nilai maksimum dan minimum dalam tensor, serta manipulasi bentuk tensor, seperti menghilangkan dimensi yang tidak diperlukan, menjadi landasan untuk analisis data dalam konteks deep learning.

### **01. PyTorch Workflow**

Proses pembuatan dataset linier menunjukkan aplikasi praktis dari rumus regresi linier. Pembuatan 100 data titik, dengan pembagian 80% data latihan dan 20% data pengujian, mengajarkan pentingnya pemisahan data dalam pembelajaran mesin. Fokus pada konsep-konsep dasar PyTorch seperti `torch.Tensor`, serta penggunaan GPU untuk komputasi tensor, memperluas pemahaman tentang operasi dasar PyTorch. Latihan ini juga melibatkan penyesuaian bentuk tensor, yang menjadi keterampilan penting dalam penyesuaian format data untuk pemodelan deep learning.

### **02. PyTorch Classification**

Pembelajaran tentang operasi tensor dan pengaturan seed acak di GPU menempatkan fondasi bagi pemahaman yang lebih mendalam tentang deep learning. Selanjutnya, latihan ini melibatkan penerapan matematika dalam konteks deep learning, seperti pencarian nilai ekstrem dalam tensor dan manipulasi bentuknya. Pembuatan model klasifikasi dasar di PyTorch, dengan fokus pada preprocessing data dan pemilihan fungsi loss dan optimizer, menyoroti pentingnya pemilihan komponen model yang tepat. Pengujian dan evaluasi model menuntun ke pemahaman yang lebih baik tentang penggunaan metrik evaluasi seperti akurasi, precision, dan recall.

### 03. PyTorch Computer Vision

Langkah awal dalam computer vision dengan PyTorch melibatkan penyiapan dan pengolahan data menggunakan dataset seperti MNIST dan Fashion-MNIST. Pembangunan model deep learning dengan kelas `nn.Module` memperkenalkan konsep pembuatan arsitektur model yang kompleks. Proses pelatihan dan evaluasi model menjadi langkah penting untuk menilai kinerja model yang telah dibangun.

### 04. PyTorch Custom Datasets

Latihan ini berfokus pada penyiapan data yang kompleks, termasuk pembuatan kelas dataset yang mewarisi dari `torch.utils.data.Dataset`. Penerapan transformasi data yang diperlukan dan integrasi dengan `DataLoader` menjadi kunci dalam memahami bagaimana PyTorch mengelola data yang kompleks. Proses ini penting dalam memahami penanganan data khusus dalam deep learning.

### 05. PyTorch Going Modular

Eksplorasi tentang modularitas dalam PyTorch mengajarkan pemisahan definisi model, data loading, dan fungsi training ke dalam file terpisah. Penggunaan kelas dan fungsi untuk mengkapsulasi kode membantu dalam memahami aliran dan struktur kode yang lebih baik. Modularitas ini membuktikan manfaatnya dalam debugging dan pengujian komponen kode secara terpisah.

### 06. PyTorch Transfer Learning

Menggunakan model pra-latih seperti ResNet dan menyesuaikannya untuk tugas klasifikasi khusus melalui 'fine-tuning' menjadi fokus utama. Pembekuan bobot pada lapisan awal dan penerapan teknik augmentasi data mengajarkan cara efektif dalam transfer learning. Proses ini memperkuat pemahaman tentang memanfaatkan model pra-latih untuk meningkatkan efisiensi pelatihan.

### 07. PyTorch Experiment Tracking

Pelacakan eksperimen menjadi aspek penting dalam pengembangan model. Menggunakan library khusus dalam PyTorch untuk mencatat detail eksperimen membantu dalam melakukan

analisis komparatif yang efektif. Proses ini mengajarkan pentingnya merekam detail seperti arsitektur model, parameter, dan metrik kinerja.

### **08. PyTorch Paper Replicating**

Replikasi arsitektur Vision Transformer (ViT) menyoroti kemampuan PyTorch dalam mengolah citra menggunakan teknik yang serupa dengan pemrosesan bahasa alami. Eksperimen ini melibatkan pelatihan dan evaluasi model terhadap dataset uji, memberikan pemahaman tentang implementasi teori dalam praktek.

### **09. PyTorch Model Deployment**

Penyiapan lingkungan kerja, penyiapan data, dan latihan tentang pengerjaan serta penyebaran model menutup rangkaian pembelajaran ini. Latihan ini mengajarkan tentang pelatihan, evaluasi, dan pentingnya penyebaran model yang efektif dalam praktek.

### **Kesimpulan:**

Melalui serangkaian latihan yang terstruktur, pengetahuan mendalam tentang PyTorch dan penerapannya dalam pembelajaran mesin telah diperoleh. Mulai dari manipulasi data hingga penyebaran model, setiap latihan memberikan wawasan baru dan menguatkan pemahaman tentang aspek-aspek kunci dalam pembelajaran mesin menggunakan PyTorch. Kemampuan untuk menerapkan konsep teoritis dalam konteks praktis membuktikan menjadi aset penting dalam perjalanan pembelajaran ini. Laporan ini tidak hanya mendokumentasikan proses belajar tetapi juga menjadi referensi berharga untuk penerapan PyTorch dalam proyek-proyek mendatang.