

Nama : Andhika Aria Pratama N

NIM : 1103202121

05 PyTorch Going Modular

- Unduh data

Tahap ini merupakan pengunduhan data yang nantinya akan digunakan untuk pengolahan.

- Create Datasets and DataLoaders (data_setup.py)

Pada tahap ini berfungsi untuk membantu dalam pembuatan PyTorch DataLoaders untuk data klasifikasi gambar. Modul `'data_setup'` menyediakan fungsi `'create_dataloaders'` yang dapat digunakan untuk mengonversi direktori pelatihan dan pengujian menjadi PyTorch Datasets, dan kemudian menghasilkan DataLoaders dengan parameter seperti transformasi, ukuran batch, dan jumlah pekerja. Dengan menggunakan modul ini, pengguna dapat dengan mudah membuat DataLoaders dan mendapatkan daftar nama kelas target dari dataset. Modul ini memudahkan pengaturan data dalam format yang sesuai untuk melatih dan menguji model klasifikasi gambar.

- Membuat fungsi `train_step()` dan `test_step()` serta `train()` untuk menggabungkannya

Kode pada tahap ini berisi fungsi-fungsi yang membantu dalam melatih dan menguji model PyTorch. Modul `'engine'` menyediakan fungsi `'train_step'` dan `'test_step'` yang masing-masing digunakan untuk satu epoch pelatihan dan pengujian model. Fungsi `'train'` menggunakan fungsi-fungsi tersebut untuk melatih dan menguji model dalam sejumlah epoch, menghitung dan menyimpan metrik evaluasi seperti loss dan akurasi pada set pelatihan dan pengujian. Modul ini memudahkan penggunaan fungsi-fungsi tersebut untuk melibatkan model PyTorch dalam pelatihan dan pengujian dengan parameter yang diberikan.

- Membuat fungsi untuk menyimpan model (utils.py)

```

%%writefile going_modular/utils.py
"""
Contains various utility functions for PyTorch model training and saving.
"""
import torch
from pathlib import Path

def save_model(model: torch.nn.Module,
               target_dir: str,
               model_name: str):
    """Saves a PyTorch model to a target directory.

    Args:
        model: A target PyTorch model to save.
        target_dir: A directory for saving the model to.
        model_name: A filename for the saved model. Should include
            either ".pth" or ".pt" as the file extension.

    Example usage:
        save_model(model=model_0,
                    target_dir="models",
                    model_name="05_going_modular_tinyvgg_model.pth")
    """
    # Create target directory
    target_dir_path = Path(target_dir)
    target_dir_path.mkdir(parents=True,
                           exist_ok=True)

    # Create model save path
    assert model_name.endswith(".pth") or model_name.endswith(".pt"), "model_name should end with '.pt' or '.pth'"
    model_save_path = target_dir_path / model_name

    # Save the model state_dict()
    print(f"[INFO] Saving model to: {model_save_path}")
    torch.save(obj=model.state_dict(),
                f=model_save_path)

# Import utils.py
from going_modular import utils

# Save a model to file
save_model(model=...
            target_dir=...,
            model_name=...)

```

Kode ini menyediakan fungsi `save_model` untuk menyimpan model PyTorch ke dalam direktori target dengan nama file yang ditentukan. Dalam implementasinya, fungsi ini membuat direktori target jika belum ada, kemudian menyimpan model menggunakan `torch.save` dengan menyertakan `state_dict()` dari model tersebut. Setelahnya, direktori tempat model disimpan dicetak sebagai informasi. Dengan demikian, fungsi ini memudahkan penyimpanan model PyTorch ke dalam file dengan mengikuti praktik yang umum digunakan.

- Train, evaluate and save the model (train.py)

Kode pada tahap ini melatih model klasifikasi gambar PyTorch menggunakan kode yang bersifat agnostik terhadap perangkat. Dengan mengatur hiperparameter, direktori data, perangkat target, dan transformasi data, skrip menggunakan modul `data_setup`, `engine`, `model_builder`, dan `utils` untuk membuat DataLoaders, model TinyVGG, dan melatih model tersebut. Selanjutnya, skrip menyimpan model yang telah dilatih ke dalam direktori "models" dengan nama file "05_going_modular_script_model_tinyvgg_model.pth". Dengan demikian, skrip ini membantu proses pelatihan dan penyimpanan model secara terstruktur dan modular.