Softmax Classifier for CIFAR-10

Overview

This repository contains code for training and testing a softmax classifier on the CIFAR-10 dataset.
#代码说明见实验报告

Files and Directories Description

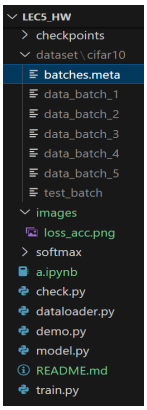
- **demo.py**: Demonstrates the model's classification performance on the test batch. It can be directly executed to view the results.
- **train.py**: Contains the training function for the model. The SEED is set to 0 to ensure reproducibility.
- **check.py**: Provides model checks using PyTorch's built-in functions. It verifies various functionalities including `softmax_loss_naive`, `softmax_loss_vectorized`, data loading, model training, and outputs.
- **dataloader.py**: Handles data loading operations.
- **softmax/**: This directory contains the naive and vectorized implementations of the softmax loss function.
- **checkpoints/**: Contains the weights saved during model training.
- **dataset/**: The directory where the CIFAR-10 dataset in `.pth` format should be placed. If you're using the `cifar10-python` version, make sure to place the files under `dataset/cifar10`.
- **model.py**: The model definition for the softmax classifier.

Setup and Usage

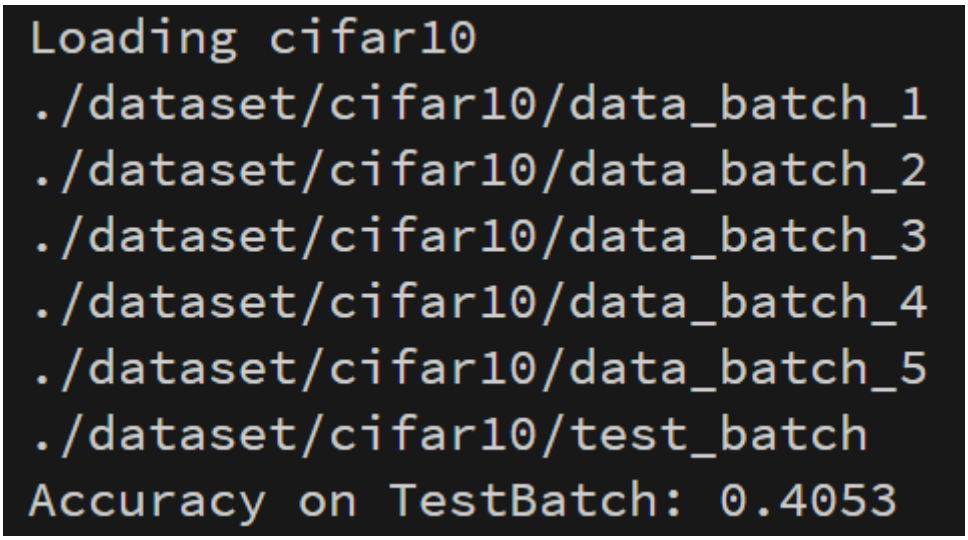
1. **Data Setup**:
    - Create a directory named `dataset`.
    - Run `demo.py` or `train.py` to automaticly download `CIFAR-10`:

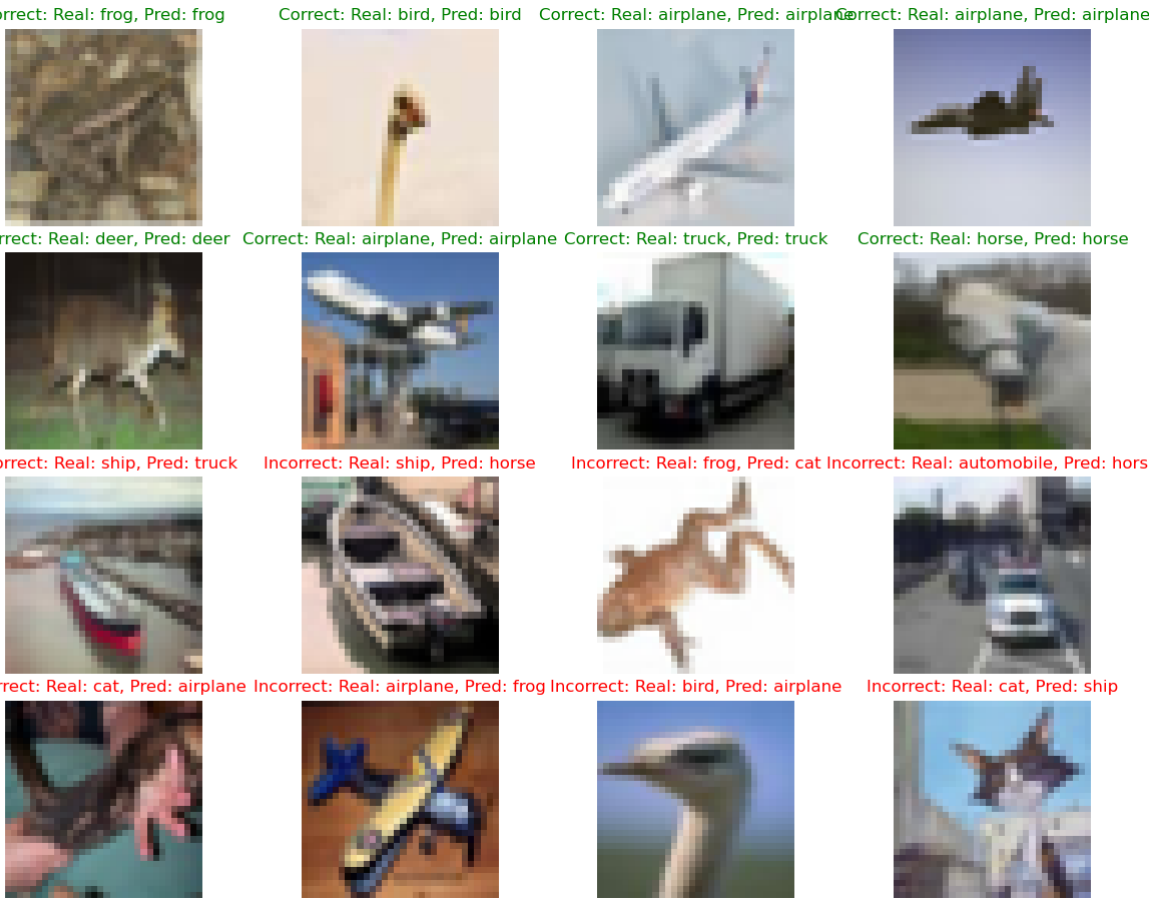
```
CIFAR10_URL = "https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz"
DATASET_DIR = "./dataset/cifar10"

# Create `./dataset/cifar10`  and put all 7 .pth files in folder `cifar10`
download_and_extract_cifar10(CIFAR10_URL, DATASET_DIR)
```

    - Or you can manually place the CIFAR-10 dataset (in `.pth` format) download inside `dataset/cifar10` like this:

- 

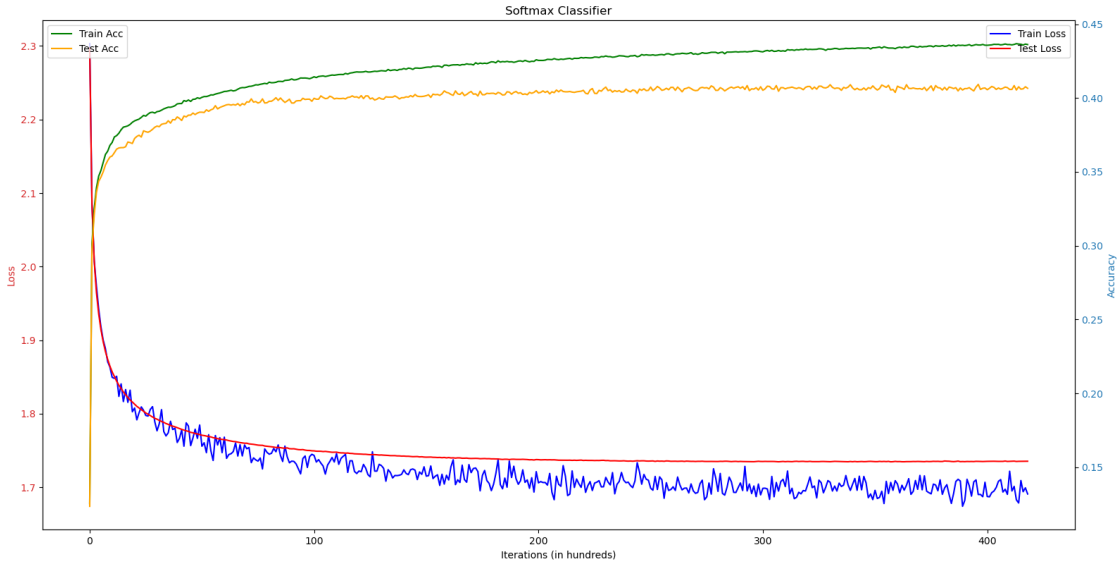
2. **Demonstration**:

   - To see the model's performance on the test batch, run `demo.py`.
   - Final Accuray on Test Batch is `40.53%`
   - 



3. **Training**:

   - Run `train.py` to start the training process, the `SEED` is set to zero.
   - The weights will be saved under the `checkpoints` directory.



4. **Model Verification**:

   - Use `check.py` to verify various functionalities of the model.
   - If you see this, the code is working correctly.

```
SOFTMAX_LOSS_NAIVE TEST
100%|██████████| 50/50 [00:01<00:00, 44.99it/s]
All checks pass on device 'cpu' |  avg: 0.02131 s/iter
100%|██████████| 50/50 [00:04<00:00, 11.98it/s]
All checks pass on device 'cuda' |  avg: 0.05566 s/iter
SOFTMAX_LOSS_VECTORIZED TEST
100%|██████████| 50/50 [00:04<00:00, 12.34it/s]
All checks pass on device 'cpu' |  avg: 0.02488 s/iter
100%|██████████| 50/50 [00:01<00:00, 41.25it/s]
All checks pass on device 'cuda' |  avg: 0.00052 s/iter
##########################################
100%|██████████| 5000/5000 [00:01<00:00, 2528.25it/s]
100%|██████████| 5000/5000 [00:03<00:00, 1561.42it/s]
[1.6066854000091553, 1.37345552444458, 1.2027146816253662, 1.0768338441848755, 0.981938362121582]
[1.6066856384277344, 1.37345552444458, 1.2027148008346558, 1.0768338441848755, 0.9819384813308716]
MODEL TEST PASSED
##########################################
LABEL READING TEST
10000 * 3072
labels:  [b'airplane', b'automobile', b'bird', b'cat', b'deer', b'dog', b'frog', b'horse', b'ship', b'truck']
[b'airplane', b'automobile', b'bird', b'cat', b'deer', b'dog', b'frog', b'horse', b'ship', b'truck']

##########################################
```

```
BATCH LOADING TEST
./dataset/cifar10/data_batch_1
dict_keys([b'batch_label', b'labels', b'data', b'filenames'])
batch_labels: b'training batch 1 of 5'
labels in batch_test: (10000,) [6, 9, 9, 4, 1, 1, 2, 7, 8, 3]
(10000, 3072)
filenames in batch_test: (10000,) [b'leptodactylus_pentadactylus_s_000004.png', b'camion_s_000148.png', b'tippe
r_truck_s_001250.png']
##########################################
CIFAR 10 LOADING TEST
Loading cifar10
./dataset/cifar10/data_batch_1
./dataset/cifar10/data_batch_2
./dataset/cifar10/data_batch_3
./dataset/cifar10/data_batch_4
./dataset/cifar10/data_batch_5
./dataset/cifar10/test_batch
training set
(45000, 3072)
(45000,)
validation set
(5000, 3072)
(5000,)
test set
(10000, 3072)
(10000,)
DATALOADER TEST PASSED
(intro2dl) PS C:\Users\liyj\Desktop\DL\Intro2DL\ML_and_DataMining\lec5_hw>
```

Notes

The training process is designed to be completely reproducible due to the fixed SEED value.