Data Science HW6

Prof M.-S. Chen & Prof M.-L. Lo

TA Office hours

- Every Monday 15:00-16:00 @ 博理603
- Contact
  - email: ntuds2021@gmail.com

HW6 - Neural Network

Problem: Convolutional Neural Network

- In this problem, we need to train CNN models to recognize handwritten digits (0~10).
- The dataset we're going to use is *MNIST-M*, which contains 70k images of handwritten digits from "0" to "9".

Ganin, Yaroslav, et al. "Domain-adversarial training of neural networks." The journal of machine learning research 17.1 (2016): 2096-2030.

Problem: Convolutional Neural Network

- We'll use ***kaggle*** for the evaluation of this homework: https://www.kaggle.com/t/41f1ecb6b78845fca557e3dac60b7393
  please **edit your team name** as **<student_id>_<what ever you want>**
- The data is provided on the website above and has been split into 3 sets: training set (45,000 images), validation set (15,000 images), and testing set (10,000 images).
- Please **DO NOT** train your models with **the testing set**.
- Validate your models with the validation set and predict the labels of all the images in the testing set.

Problem: Convolutional Neural Network

- Please output your **prediction result into a .csv file**, and follow the format (sample.csv) provided on kaggle.
- Upload the output .csv file to kaggle and the testing accuracy of the public testing set will show right the way (the one for the private testing set will be revealed after the **submission deadline**, e.g., **2021/12/31 23:59, UTC+8**).
- You have only **5 chances** per day for submission, be careful before you submit.
- The **sample code** of training, validation and inference is also available on kaggle, please refer to it if any problem encountered.

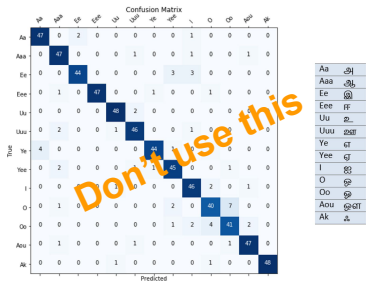
Problem: Convolutional Neural Network

- Any predefined models is welcomed (pretrained weights are **FORBIDDEN** though), however, they may not give you better performance. So, we encourage you to design your own CNN models.
- **Guidance for passing baselines:**
  - **simple baselines (30%)**
    modify the learning rate, train for more epochs, etc.
  - **strong baselines (20%)**
    employ deeper models, conduct data augmentation, utilize dropout/batch normalization layers, etc.

Link

Problem: Convolutional Neural Network

- Report **(50%)**
  - State all the **hyperparameters** you need for training (learning rate, #epochs, weight decay, moment, etc.) and how you tune them (**10%**)
  - Show the **structure** of your best model. (hint : print(model)) (**5%**)
  - Explain the **design of your model** and what you've observed. (**10%**)
  - Plot the **learning curve** during training (CrossEntropy Loss). (**10%**)
  - Plot the **confusion matrix** for validation set, and briefly explain what you've observed. (**15%**)



Problem: Convolutional Neural Network

- Grading(100%):
  - **15% -** public simple baseline (93.557%)
  - **10% -** public strong baseline (98.100%)
  - **15%** - private simple baseline (***Unknown***)
  - **10% -** private strong baseline (***Unknown***)
  - **50%** - report

Problem: Convolutional Neural Network

- Submission
  - Submit the .csv files of your prediction for the testing images to kaggle
  - **MUST edit your team name** on kaggle correctly as p.5 mentioned.
  - Submit the code & report (in .**zip**) to ceiba. After TA unzips, there should be
    - hw6_<學號>/
      - report.pdf
      - other training/testing python files
  - 10-point deduction for wrong format!
  - **The deadline is on 23:59, 12/31 (Fri.)**
    no grade would be given for late submission!

Have fun training models!