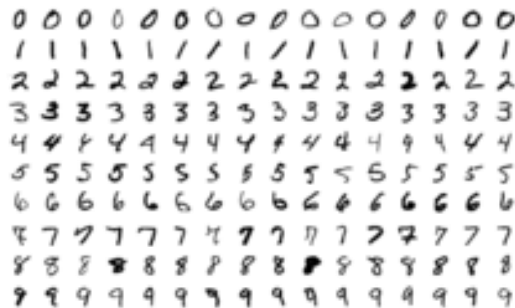


Artificial Intelligence Assignment 3

**CSI4108-02
Spring, 2018**

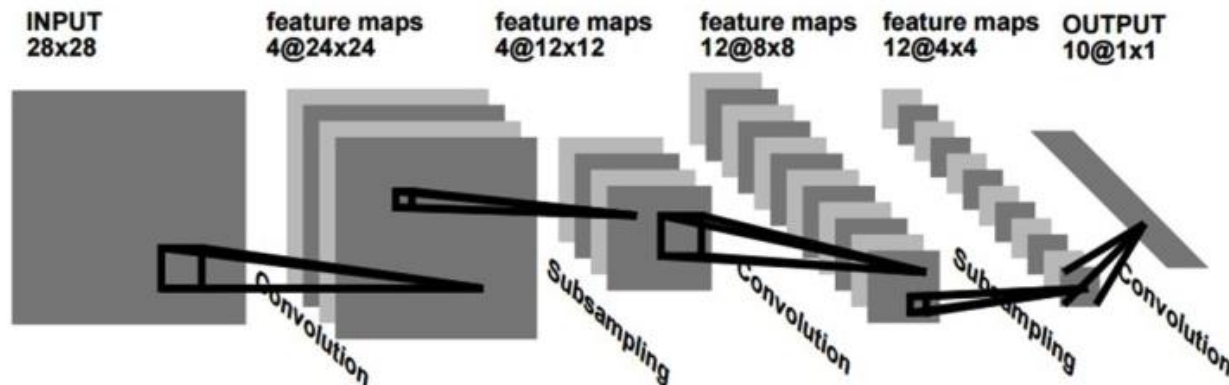
1. Introduction

- We will do MNIST classification by constructing LeNet.



0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9

<MNIST dataset>



<LeNet>

2. Detail-Code

- Framework: You can use Tensorflow or PyTorch. Also if you have GPU, you can use it.
- MNIST data: You can get MNIST data using below way

- Tensorflow(https://www.tensorflow.org/versions/r1.2/get_started/)

```
from tensorflow.examples.tutorials.mnist import input_data
mnist = input_data.read_data_sets("MNIST_data/", one_hot=True)
```

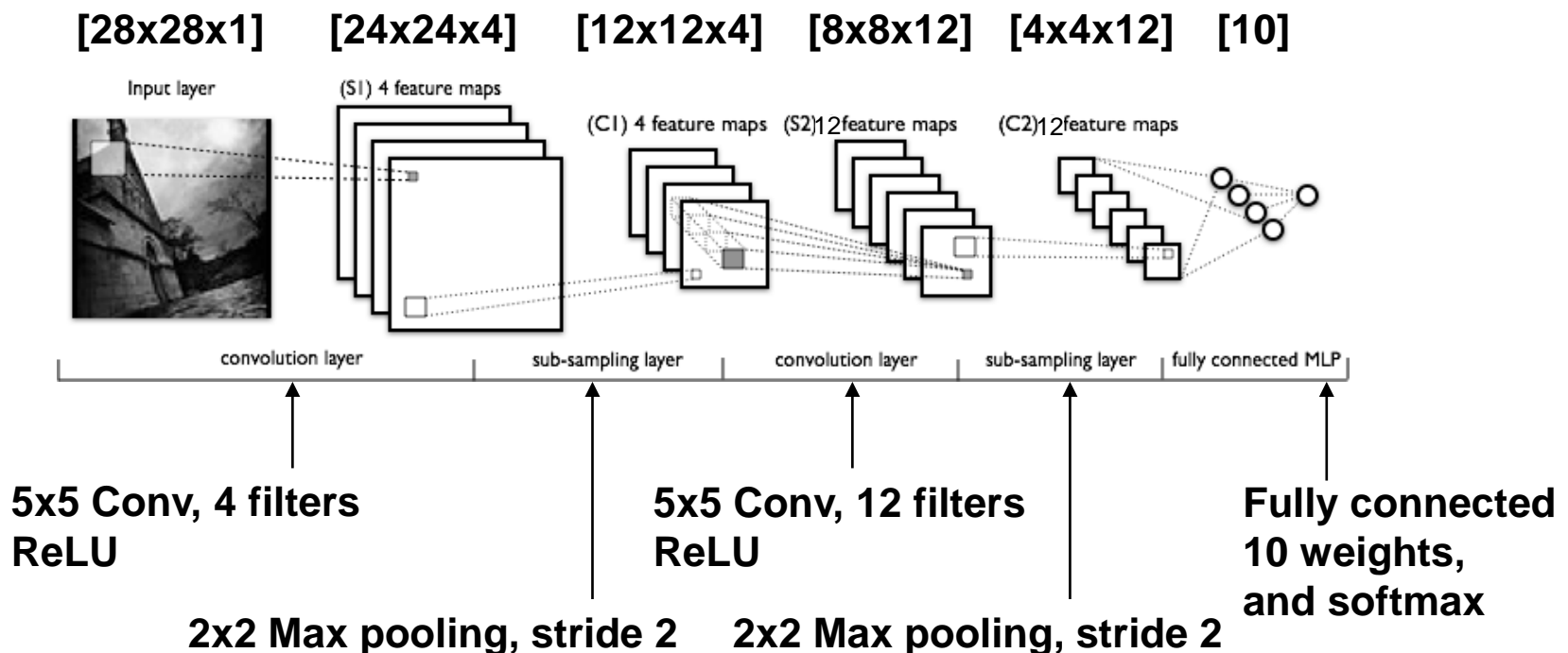
- PyTorch(<https://pytorch.org/docs/master/torchvision/datasets.html#mnist>)

```
class torchvision.datasets.MNIST(root, train=True, transform=None, target_transform=None,
download=False)
```

2. Detail-Code

- Network: You must follow the **structure of LeNet presented below**. (size of conv, number of filters, Non-linear activations, stride, etc...)

- LeNet



2. Detail-Code

- Output: For each Epoch, Print the **average Loss** and **Learning rate** for the train set. And after training, print the **average Loss** and **Accuracy** for the test set.(on standard output)

```
Train set: Epoch: 1, Average Loss: 1.803760, lr: 1.00e-02
Train set: Epoch: 2, Average Loss: 1.653157, lr: 1.00e-02
Train set: Epoch: 3, Average Loss: 1.586098, lr: 1.00e-02
Train set: Epoch: 4, Average Loss: 1.580281, lr: 1.00e-02
Train set: Epoch: 5, Average Loss: 1.491629, lr: 1.00e-02
Train set: Epoch: 6, Average Loss: 1.484734, lr: 1.00e-02
Train set: Epoch: 7, Average Loss: 1.482435, lr: 1.00e-02
Train set: Epoch: 8, Average Loss: 1.480637, lr: 1.00e-02
Train set: Epoch: 9, Average Loss: 1.479552, lr: 1.00e-02
Train set: Epoch: 10, Average Loss: 1.478573, lr: 1.00e-02
```

```
Test set: Average loss: 1.4813, Accuracy: 9800/10000 (98.00%)
```

<Example output>

2. Detail-Report

- **Report: It should contain the contents below.**
 - **Environment ex) OS, Framework, versions(pytorch, tensorflow, python, cuda, etc...), etc...**
 - **Detail description of each line of code (You should write a description of each function definition and role. For example, if you use MSELoss then you should explain about MSELoss)**
 - **Screenshot result output**
 - **Analysis of training process and result**
 - **Other information (If you want)**

3. Detail

- **Summary**

**We will do MNIST classification using machine learning framework.
The network structure uses LeNet.**

Print the training process and test result on standard output.

Write a report with the necessary contents.

4. Submission

- Deliverables: **2013147xxx_3.zip** (studentNumber_3.zip)
- Must include
 - **main.py**
 - Other codes (If you necessary)
 - **report.pdf**

(Your code with detail comments)

5. Grading environment & Directions

- Language: **Python**
- We grade your score in Linux(Ubuntu 16.04)
- Python3 ($\geq 3.5.2$)
- This is an individual project
- **You should follow the output format**
- **Never copy code**
- You will get **0** points if you cheat
- Do not manipulate the results.
- If the result is manipulated, you will get **0** points.

6. Grading policy

- **Code : 50pts**
 - Import MNIST data
 - Implement LeNet
 - Train and test are well
- **Report : 50pts**
 - Contain necessary contents

7. Due Date

- Due Date: **7/June/2018 23:59:00 KST**
- Delay Policy: **-15pts per day**

Please use YSCEC Q&A board to leave your question.