

Report Date: 06/17/2022

To:

- ematson@purdue.edu
- ahsmith@purdue.edu
- lhiday@purdue.edu
- lee3450@purdue.edu
- wang4070@purdue.edu

From: TN

- Eunyoung Bang (yeong35@kangwon.ac.kr)
- Yeongmin Seo (dudals1003@cu.ac.kr)
- Jeongyoun Seo (201810773@sangmyung.kr)
- Raymond Zeng (zeng172@purdue.edu)
- Aminata Niang(aminata.niang@telecom-sudparis.eu)

Summary

- The training code for deep learning was finished.

What TN completed this week

- The training code for the NN (Neural Networks) was finished.

```
1 import torch
2 import torch.nn as nn
3
4 # Neural Networks example
5 class ClassifireNN(nn.Module):
6     def __init__(self, drop_out=0.0):
7         super(ClassifireNN, self).__init__()
8         self.fc1 = nn.Linear(2 * 20, 32)
9         self.fc2 = nn.Linear(32, 16)
10        self.fc3 = nn.Linear(16, 8)
11        self.fc4 = nn.Linear(8, 4)
12        self.fc5 = nn.Linear(4, 1)
13
14        self.relu = nn.ReLU()
15
16        self.drop_out = nn.Dropout(p=drop_out)
17
```

```
54 model = ClassifireNN(drop_out=DROP_OUT).to(device)
55 optimizer = torch.optim.Adam(model.parameters(), lr=LR)
56 criterion = nn.BCELoss()
57
58 best_auc = 0
59 best_epoch = -1
60 best_pred = []
61
62 prev_model = None
63
64 writer = SummaryWriter()
65
66 for i in tqdm(range(EPOCHS)):
67
68     # Train
69     train_loss = 0
70     true_labels = []
71     pred_labels = []
72     model.train()
73
```

Fig 1. The example of a Neural Networks model.

Fig 2. The sample of training code.

- Learn about Machine Learning and Git [1], [2]
- Git is a free, scalable, open-source distributed version control system that is designed to provide a high level operations and full access to internals [3]. Learned how to use the git command following the lecture video

```

git init: you need to write this command to make a '.git' file, and initialize an empty :  

where you are now (directory).v

git config --global user.name (name): Set the name so that you do not confuse other  

create commits and commit messages.v

git config --global user.email (email address): Set up an email address to make it easier  

confusing when someone can contact you.v

git add: When you create or modify a file or something in the same branch, you must  

track and use commit commands.v

git status: checking the branch, Is there any modifying stuff or should I have to commit  

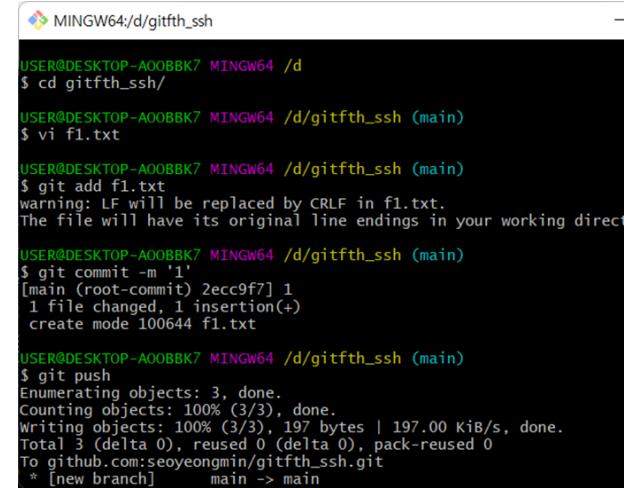
it.v

git commit: when you do some task at your repository and complete your task, for I  

add that all and commit it to save it.v

```

Fig 3. The definition of a git command.



```

MINGW64:/d/gitfth_ssh
USER@DESKTOP-AOOBBK7 MINGW64 /d/gitfth_ssh
$ cd gitfth_ssh/
USER@DESKTOP-AOOBBK7 MINGW64 /d/gitfth_ssh (main)
$ vi f1.txt
USER@DESKTOP-AOOBBK7 MINGW64 /d/gitfth_ssh (main)
$ git add f1.txt
warning: LF will be replaced by CRLF in f1.txt.
The file will have its original line endings in your working directory
USER@DESKTOP-AOOBBK7 MINGW64 /d/gitfth_ssh (main)
$ git commit -m '1'
[main (root-commit) 2ecc9f7] 1
 1 file changed, 1 insertion(+)
  create mode 100644 f1.txt
USER@DESKTOP-AOOBBK7 MINGW64 /d/gitfth_ssh (main)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 197 bytes | 197.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:seoyeongmin/gitfth_ssh.git
 * [new branch]      main -> main

```

Fig 4. The practice of using a git command.

- Confusion metrics are as follows. In the logistic regression, if $h(x) \geq 0.5$, predict $y=1$ and if $h(x) < 0.5$, predict $y=0$. In this case, this 0.5 is referred to as a threshold, and when this threshold is increased, the Precision value increases and the Recall value decreases. Conversely, when the threshold is lowered, the Precision value is lowered and the Recall value is increased.

		Actual Class: 1	Actual Class: 0	$Accuracy = \frac{(true\ positives + true\ negatives)}{(total\ examples)}$
		True Positives	False Positives	$Precision = \frac{(true\ positives)}{(true\ positives + false\ positives)}$
		False Negatives	True Negatives	$Recall = \frac{(true\ positives)}{(true\ positives + false\ negatives)}$
Predicted Class: 1	Predicted Class: 0			$F1\ score = \frac{(2 * precision * recall)}{(precision + recall)}$

Fig 5. Calculation method of Accuracy, Precision, Recall, and F1 score.

- SVM (Support vector machine) is one of machine learning and a model that defines a decision boundary for pattern recognition and data analysis. When there are two classes, the distance between the decision boundary separates the two classes and the class is called margin. SVM is a model for finding decision boundary that maximizes this margin.
- Learned about new python library [4], [5]:
 - Pytorch** is a framework developed by Facebook AI Research for deep learning, featuring debugging tools and high-level of customization for advanced users.
 - torchaudio** provides powerful audio I/O functions, preprocessing transforms and dataset, model implementations and application components and is powered by Pytorch.
 - NumPy** (standing for Numerical Python) is a Python library used for working with arrays. Numpy aims to provide an array object that is up to 50x faster than traditional Python lists.

- **Pandas** used for working with the dataset. It has functions for analyzing, cleaning, exploring, and manipulating data.

Things to do by next week

- The training code for CNN will be finished.

Problems or challenges:

- The charger is still lost. The other charger will be bought.
- The fast speed dataset shows two yellow lines.
 - Lines assumed that made when UAV changed direction. Yellow lines will be cut off and other parts will be used for our project.

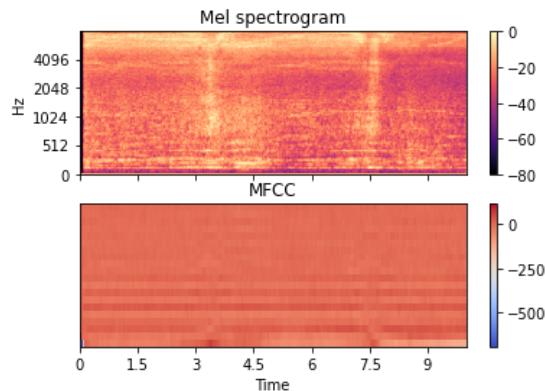


Fig 6. The fast speed dataset.

The dataset shows lines in the Mel spectrogram.

References

- [1] holehouse, “Stanford Machine Learning”, holehouse.org. <http://www.holehouse.org/mlclass/> (accessed June. 02, 2022)
- [2] Coding Everybody, “자] 옥에서 온 Git (새 수업으로 대체)”, opentutorials.org. <https://opentutorials.org/course/2708> (accessed June. 16, 2022)
- [3] Linus Torvalds “Git source code” github.com. <https://github.com/git/git> (accessed June. 16, 2022)
- [4] Soumith Chintala, “Deep Learning with Pytorch”, pytorch.org. https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html (accessed June. 17, 2022)
- [5] “Pandas Tutorial”, w3schools.com. https://www.w3schools.com/python/pandas/pandas_intro.asp (accessed June. 17, 2022)