Report Date: 07/01/2022
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(aminatabinetabibiniang@gmail.com)

Summary

- Found some error in Machine Learning code and fixed it.
- Learned deep learning theory.

What TN completed this week

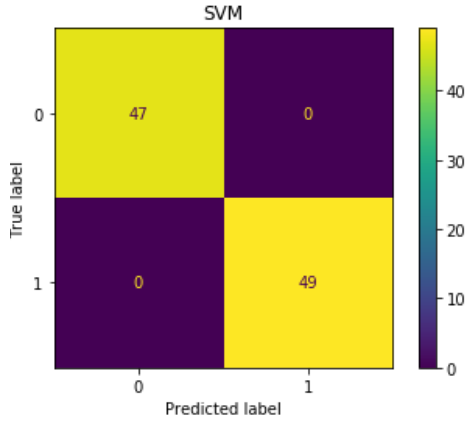
- Conducted EDA for dataset

Sec	Velocity	F 0	F 1	F 2	F 3	F 4	F 5	F 6	F 7	F 8	F 9	F 10	F 11	F 12	F 13	F 14	F 15	F 16	F 17	F 18	F 19
3	slow	-48.16	-25.84	19.66	0.91	-17.52	21.85	-10.59	20.11	-15.26	9.50	-0.63	-9.8	7.72	-1.24	-1.62	0.52	-4.19	-6.14	-3.47	1.66
3	fast	-55.64	-11.42	25.14	-3.21	-23.56	22.03	-14.79	14.18	-18.04	7.65	-3.59	-6.54	1.83	-1.03	-5.80	-0.25	-5.22	-2.39	-3.67	0.12
10	fast	-55.64	-11.42	25.14	-3.21	-23.56	22.03	-14.79	14.18	-18.04	7.65	-3.59	-6.54	1.83	-1.03	-5.80	-0.25	-5.22	-2.39	-3.67	0.12

Table 1. compare the MFCC feature of the sound dataset.

- The slow velocity and fast velocity dataset shows quite different features.
- It shows the reason why Machine Learning models show good accuracy.
- Fast 3 second and 10 second shows similar results. Therefore the yellow lines are not a problem to detect UAV velocity.

- Fixed the error in Machine Learning code




```
conf_mat = confusion_matrix(y_test, pred_y)
conf_mat
```

✓ 0.3s

```
array([[47,  1],
       [ 0, 48]])
```

Fig 1. The plot shows the wrong result.

- read the paper and added reference in methodology [1] - [3]
- Analysis of functions through code written to fit the Pytorch environment [5].
    - h(x) may been described “h(x)=w*x+b”. It is possible to calculate “h=x.matmul(W) + b”. In addition to this, the logical regression calculates “h=torch.sigmoid (x.matmul(W) + b)”.
      In linear regression, the cost function can be calculated using “mse_loss” of torch.nn.functional. In the logistic regression, the cost can be calculated with “binary_cross_entropy (h, y)” of torch.nn.functional.
    - Backward function is required to improve h(x) with cost. First, set the optimizer, and set the degree of change to zero before executing the backward step using “optimizer.zero_grad()”. In pytorch, the gradient is set to be added to the existing gradient every time loss.backward is executed, so the initialization of the gradient with “optimizer.zero_grad()” is essential before the backward. The gradient is then calculated using “cost.backward()”, and the improvement of h(x) ends by updating the values of w, b with ‘optimizer.step().’
    - When obtaining Loss with a linear neural network, logit (or score) comes out of the model and normalizes through sigmoid function. Then, we obtain Loss through Cross Entropy with the correct answer. In the case of a binary case with two classes, such as classification of drone speed and slowness, it is efficient to do Cross Entropy with BCELoss.
- Brad
- Learning how the DL algorithm works with YouTube lectures and book [4], [6].
    - Watched a YouTube lecture, called ‘DeepLearning for everyone’.
    - Learning how to calculate the output size of convolution.

$$\text{Output size} = \frac{input\ \ size - filter\ \ size + (2 * padding)}{Stride} + 1$$

    - The concept(principle) of Neuron, Convolution, Pooling, CNN implementation, padding, and stride.
    - Practicing run the CNN with MNIST datasets,

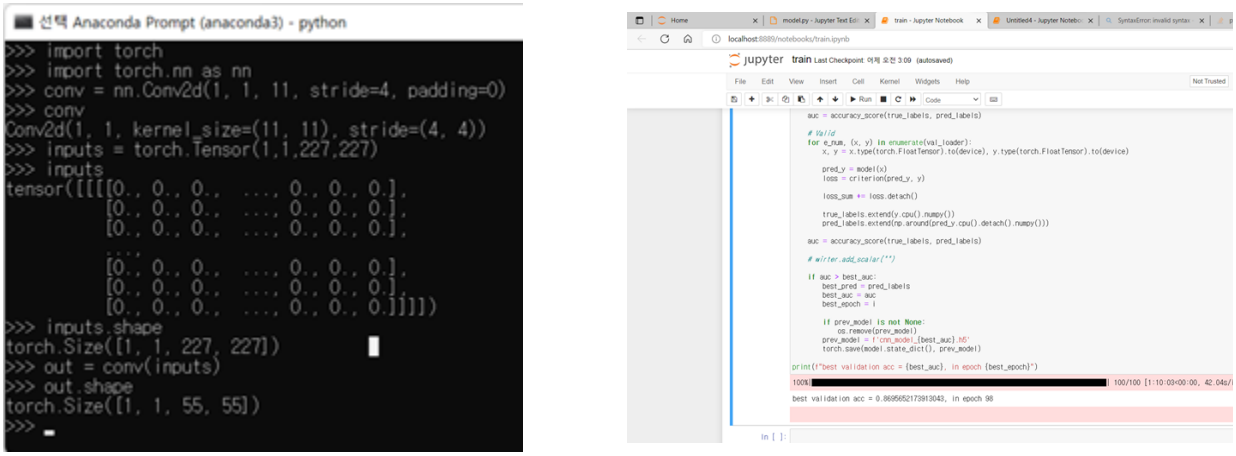


Fig 1. Practicing the writing DL algorithm code

- Conducting the writing DL algorithm code
    - Changed environment to ‘COLAB’ from ‘anaconda3’ for training the ML algorithm and DL algorithm, and running the code for understanding the code.
    - Starting to find the hyper parameter which can make the highest accuracy of CNN.
        - First, find the related paper with our research and set the number of channels, kernel size, stride, padding, etc.

Things to do by next week

- Collect a dataset using two types of drones.

Problems or challenges:

- Wait for the delivery about UAV and UAV’s charger.
- Using CUDA as GPU was unproceeded, and trying to solve using CUDA as GPU, reinstall the pytorch, cuda and also download the NVIDIA driver. In the end, we succeeded with upgrading the CUDA version 11.7. However, It was not the solution of using CUDA as a GPU when we are running the ‘train.ipynb’ code.

References

[1] O. K. Toffa and M. Mignotte, "Environmental Sound Classification Using Local Binary Pattern and Audio Features Collaboration," in IEEE Trans. on Multimedia, 2021, vol. 23, pp. 3978-3985.

[2] Racharla, Karthikeya, et al. "Predominant musical instrument classification based on spectral features." 2020 7th Int. Conf. on Signal Process. and Integr. Networks (SPIN). IEEE, 2020.

[3] R. K. Tripathy, S. Dash, A. Rath, G. Panda, and R. B. Pachori, "Automated Detection of Pulmonary Diseases From Lung Sound Signals Using Fixed-Boundary-Based Empirical Wavelet Transform," IEEE Sensors Letters, vol. 6, no. 5, pp. 1-4, May 2022, Art no. 7001504

[4] Deep Learning Zero To All, “[PyTorch] Lab-10-2 mnist cnn”, youtube.com. https://www.youtube.com/watch?v=wQtkdq3tmJ8&list=PLQ28Nx3M4JrhkqBVIXg-i5_CVVoS1UzAv&index=22&t=240s (accessed June. 29, 2022)

[5] Deep Learning Zero To All, “Deep Learning Zero To All - PyTorch”, github.com. https://github.com/deeplearningzerotoall/PyTorch (accessed June. 30, 2022)

[6] Deep Learning from scratch 1, 13t ed., Hanbit Publishing Netw., Seodaemun-gu, Seoul, Korea, 2021, pp.25-