EIE4122 Deep Learning and Deep Neural Networks

Lab 3: Gradient Vanishing and ResNet

A. Objectives and Outcomes

After finishing this lab, you will have learned to:

- Understand the gradient vanishing problem.
- Understand the architecture of ResNets.
- Use Google Colab to develop a ResNet to overcome the gradient vanishing problem.

B. Assessment Criteria

- Ability to successfully execute the code of the Colab notebook.
- Ability to modify, train, and test the FFNet and ResNet models.
- Ability to make insightful observations.
- Clarity of the report (i.e., quality of writing, structure, use of figures, and formatting).

C. Submission

- Write a report containing the following:
  - Your Discussion/observations of the "further investigations" suggested in Section 4 of the Colab file.
  - The screenshots of the histograms of the weights at the bottom and the upper layers.
  - An illustration of the structure of the ResNet (you can infer this from the code or use *resnet.summary()*).
  - An explanation of why the ResNet can be very deep without suffering from the vanishing gradient problem.
  - An explanation of why ReLU can help mitigating the vanishing gradient problem.
- The length of your report should be of minimum 1000, maximum 2000 words + figures.
- **Convert your file to PDF**. Do NOT submit zip, doc, docx, etc. Only PDF files will be accepted, any other format may result in a zero mark.
- Submit your report before the deadline specified on Blackboard.

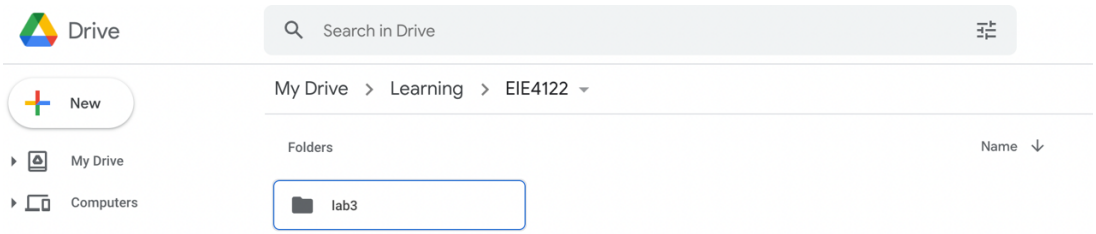
D. Instructions

D.1 Prepare Colab Environment

1. Open the Google Drive (https://drive.google.com/drive/) page in your browser. Create the following directory structure in your Google Drive:

   My Drive/Learning/EIE4122/lab3/

   After creating the folders, you should see something like this:



2. Download “Gradient_vanishing.ipynb” from and upload it to your Google Drive under “My Drive/Learning/EIE4122/lab3”.

3. Right click on “Gradient_vanishing.ipynb” and open it with Google Colaboratory, making sure to configure your notebook to use GPU by clicking Edit → Notebook settings.

4. Follow the instructions in the Colab file. Remember to capture the outputs, findings, and your observations in the report.

Hints:

1. You can find the histograms of the weights in TensorBoard as shown in Figure 1. “Distributions” shows the information in 2-D, while “Histograms” shows the same information in 3-D. **NOTE:** the “weights” are confusingly called “kernels” in TensorBoard.

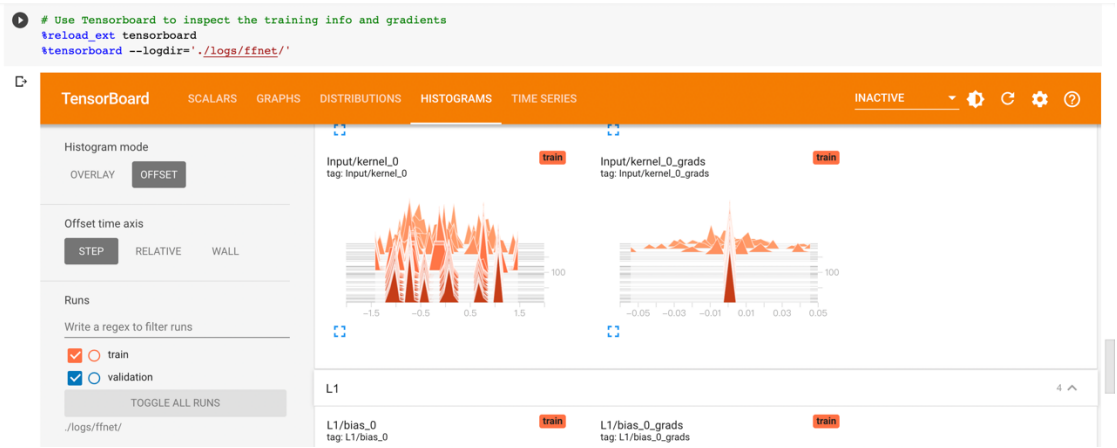


Figure 1: visualizing the histograms of the weights using TensorBoard.

2. A model using the ReLU activation function can suffer from the “dying ReLU problem” (https://en.wikipedia.org/wiki/Rectifier_(neural_networks)#Potential_problems), which means that the ReLU neurons only output values of 0. This can be avoided by adding a batch normalization layer after each dense layer. Search on the Internet for “Keras Batch Normalization” for the documentation and examples of how to use BN in Keras, and read the following stackovelfow discussion on where to place the BN layer: https://stackoverflow.com/questions/55827660/batchnormalization-implementation-in-keras-tf-backend-before-or-after-activa

References:

[1] https://machinelearningmastery.com/how-to-fix-vanishing-gradients-using-the-rectified-linear-activation-function/