

Deep Learning Assignment 4

The objective of this assignment is to train LSTMs and GRUs using **Tensorflow(use latest version 1.5)**.

Dataset : Same as Assignment 2.

Experiment 1: Consider a 28*28 image as a sequence of 28 vectors of dimension 28. Create two separate recurrent neural networks with

1. Standard LSTM Cell
2. Standard GRU Cell

Write the cells from scratch. You can use [tf.contrib.rnn.static_rnn](#) to build a recurrent neural network using the specific cell.

Use the hidden unit sizes of 32, 64, 128, 256 and get the final accuracy on test set for both LSTM and GRU. Plot test accuracy vs hidden_unit size and observe what you get.

Task: Train your network and report the accuracy on the test set. See if your performance is better than what you got in Assignment 3.

Make a pdf file called <Roll No>_Report.pdf wherein you report the performance you obtain from LSTM, GRU with the different hidden unit sizes and why you think there are any differences if any. Also write down your model parameters. Attach the plot that you created.

The report **must be submitted**. No marks would be given without the report. Also the text in the report must be very brief. (3-4 sentences).

Constraints and Considerations:

1. You must **NOT** use **tf.contrib.rnn.LSTMCell** or **tf.contrib.rnn.GRUCell** modules for building the cells, instead implement them on your own.
2. Place the data in './data' folder and use the data_loader file (if needed) as done exactly in last file. You should not submit data. Always create a train-validation split on the training set and train and test on appropriate data only. You must train properly with proper stopping criterion. All decisions you take for model parameters must be logical.
3. Store all weights of your model in "weights/" folder. The weights folder must be inside the directory in which you are coding and must be submitted to Moodle.
4. If the weights are exceeding the upload constraints imposed by Moodle (may not happen), then upload the weights in some public site like github and **download them first, by writing appropriate code for it.**
5. Please **maintain the directory structure** as mentioned here **strictly** as your assignment may not be evaluated otherwise
6. **Do NOT hardcode any paths** we may replace the data in './data' folder to evaluate your work.
7. Include all training and testing codes. A file named 'main.py' must be present in the directory you submit. It takes in some command line arguments. '*python main.py --model='lstm' --train --hidden_unit=32*' should train the LSTM model with hidden unit size 32, '*python main.py --model='lstm' --test --hidden_unit=32*' should load all model weights, test the model and report the accuracy on the test data. Similarly for GRU, only pass the argument as *--model='gru'*. Do the

same for other hidden unit sizes as well. For argument parsing you can use python's argparse library or tf.flags(more preferred).

8. Name the **ZIP file as well the folder that you compressed to get the zip** as "Assignment4_<Roll No.>.zip" and Assignment4_<Roll No.>" respectively. Upload the zip file with codes and report to moodle. Note that the zip file should contain your python files, codes and the report.
9. Small differences in your method's accuracy would not be penalized (larger may be; remember to set a fixed random seed for reproducing your result). Your experimentation technique should be sound. (like, do not test on training data)
10. Use appropriate print statements so that a third party can understand your code.

Happy Coding!