AI2100/AI5100 Deep Learning, Spring 2022

Indian Institute of Technology Hyderabad

Homework 6, **15 points**. Deep Neural Nets and Time-series Models, Assigned 23.04.2022, Due **11:59 pm on 02.05.2022**

A computer would deserve to be called intelligent if it could deceive a human into believing that it was human. – Alan Turing

Instructions:

- It is **strongly recommended** that you work on your homework on an *individual* basis. If you have any questions or concerns, feel free to talk to the instructor or the TAs.
- **You are free to use PyTorch functions in this assignment.**
- Refer to the PyTorch NN tutorial. Help on RNN, GRU, and LSTM can be found here.
- Please turn in Python Notebooks with the following notation for the file name: `your-roll-number-hw6.ipynb`.
- Do not turn in images. Please use the same names for images in your code as in the database. The TAs will use these images to test your code.

1. Train the AlexNet and ResNet50 CNN models on the Tiny ImageNet dataset. A few pointers:
   - To keep things simple, choose a subset of 10 classes
   - You can load the models directly using the `hub` APIs. Just set the `pretrained` flag to `False` so that you can train from scratch
   - Print the model summary for each case and note the number of model parameters
   - As in the previous assignments, work with cross-entropy loss
   - Use a 70:10:20 data split for training, validation and testing
   - One not-so-well-defined aspect of training CNNs is when to stop. Monitor your validation loss to decide on when to stop. In other words, stop training when your validation loss starts to increase. If this is taking too many epochs, you can stop at a pre-defined number of epochs that is dependent on your hardware.

   Report the following:

   (a) Compute the *error* on the training and test data sets. *Plot* the training and test errors as a function of epochs (at the end of training). (1)

   (b) *Visualize* the activation maps of the trained model. You can pick a couple of representative slices from the activation volumes at a couple of convolution layers. (1)

   (c) Report the *accuracy* of your classifier. (1)

   (d) Use tSNE to visualize the bottleneck feature at the end of the first epoch and the last epoch. (1)

   (e) Compare the performance of the two models in terms of the accuracy and training time (number of epochs) required for the training loss to stabilize. Comment on which model you would pick for this task considering a trade-off between performance and the number of parameters. (1)

2. Time-series models.

   (a) For a start, replicate the results from this RNN tutorial. (0)

   (b) Replace the RNN in the previous question with a GRU and report the classification performance. GRU help can be found here. (5)

   (c) Replace the GRU in the previous question with an LSTM and report the classification performance. LSTM help can be found here. (5)