ECE/CS/ME 539 – Fall 2024 — Homework 12

Exercise 1

1. In this assignment, we will create a multi-layer LSTM model to predict daily minimum temperatures.

(a) Download the timeseries dataset `daily-min-temperatures.csv`.

(b) Load the data and split it into 80% training and 20% testing. Normalize the temperatures to have 0 mean and unit standard deviation.

(c) Create a dataset class that generates a sequence of temperatures over 8 days.

(d) Using `nn.LSTM()`, create an LSTM model with a single layer and hidden dimension of 128.

(e) Train the LSTM to predict daily minimum temperature prediction. We will track data from the past 7 days to predict data of the 8$^{\text{th}}$ day. Train the model for 100 epochs, batch size of 256, and learning rate 0.001, using the Adam optimizer and MSE loss function.

(f) Repeat (d) and (e) for LSTM with depth of 2 and 3. Compare the final validation errors. Discuss your findings.

Exercise 2

Please download the starter code `RNN_LSTM_starter.ipynb` from Canvas. In this exercise, we will (1) implement an LSTM model and (2) train it for the next word prediction model. The code downloads the short story *"The Time Machine"* by H. G. Wells, made public by Project Gutenberg.

1. The provided code is complete. It implements the dataset, an LSTM model, and the training loop.

   (a) Create an LSTM model with a hidden dimension of 256 and train it for 100 epochs with a learning rate of 0.001 and batch size 32. Make sure you are using CoLab with runtime type: T4 GPU in order to finish training faster.

   (b) What was the perplexity obtained by the LSTM model? How does it compare to the simple RNN model trained in the previous activity?

   (c) The provided codebase does not use a test set. In general, we should not evaluate the performance of a model using the training data. Modify the codebase to create a training and a test set (use the first 25000 lines of the book for training and the remaining for testing). Then, train the model again on the new training set, and evaluate it on the new test set.

   (d) What is the perplexity on the training and test datasets?

Exercise 3

In this activity, we will continue to train the sequence-to-sequence Transformer network to translate French to English. Please, download the notebook tutorial `Tranformer_tutorial.ipynb`. In this activity, we will ask you to modify the code tutorial to achieve two things.

(a) First, train the original model for 30 epochs. Report the training and test losses (both at the end of each epoch, and the best obtained throughout training).

(b) Increase the model size by increasing the number of layers to 4 and model dimension to 1024. Train the new larger model. Report the training and test losses. Discuss the results.

(c) Change the code to limit both vocabularies to only 3000 words. This will remove rare words. Train the model with the smaller vocabularies. Report again the training and test losses.

(d) Instead of translating from French to English, train a model to translate from English to French. Translate the following sentence: "I am going to do well on the next test."