CSCI433/CSCI933
Machine Learning Algorithms and Applications
Assignment Problem Set #2

Lecturer: Prof. Philip O. Ogunbona(philipo@uow.edu.au)
School of Computing and Information Technology
University of Wollongong

Due date: Saturday May 14, 6:00 p.m.

Introduction

This assignment is a designed as a combination of laboratory and problem styles. You are provided a chapter from a "hands-on book" on machine learning along with the **jupyter notebook** that implements the ideas. This **Python** code requires the **TensorFlow** framework, at least version 2.8.

In this assignment, you will learn about the recurrent neural network and how it can be used to process sequence data (i.e. time-series data). By going through the examples provided in the chapter you will be able to complete the core task in this assignment.

Your task is threefold.

1. One, you must install textbfTensorFlow on your personal computer (PC-Windows, Mac, PC-Linux) and ensure that the latest version along with all dependencies are present. You are encouraged to use Anaconda and create an environment in which you install TensorFlow. This is the most trouble-free method of installing TensorFlow and the dependencies..

2. Two, you are required to read the chapter provided along with the **jupyter notebook** in the assignment package. This is a laboratory-style part of the assignment.

3. Three, You will write a report based on the laboratory and the following tasks:

   (a) Summarize your understanding of the recurrent neural network (RNN) and its configurations (i.e. LSTM and GRU). This should be no more than one page.

   (b) Implement a 24-hour temperature predictor following the example given in the chapter. Your predictor will include various tricks used to optimize the predictor performance - stacking, dropout, length of input sequence, layer normalization, etc. Include a comparison of performance when you used LSTM and GRU. This should be no more than two pages.

   (c) The *Transformer* was indicated as the "next thing" for sequence data processing. Implement a *Transformer*-based network to realise a 24-hour temperature predictor using

the same dataset. See this example code for inspiration: `https://keras.io/examples/timeseries/timeseries_transformer_classification/`. Write a short report including your understanding of how transformers work and the experiments you carried out. Pay particular attention to *multi-head attention* and its performance relative to single head, LSTM and GRU. Report on your findings. This should be no more than two pages.

Report format

Introduction to recurrent nueral network - 10 marks

This should be **no more than one page** and provides a description of RNN (including LSTM and GRU). Include diagrams if you think they will help your description. Ensure you identify the differences and similarities among the configurations.

Temperature prediction using recurrent networks - 30 marks

This should be **no more than two pages** and must describe your data preproessing /organization for input into the network. You must expalin the rationale behind the various network performance optimization "tricks" you have employed. The result of employing the "tricks" must be indicated by showing performance with and without the "trick". You must also provide the result of using either LSTM or GRU as building block of your network. What happenes when you mix them?

Transformer-based temperature predictor - 40 marks

This section should be **no more than two pages** and must include your understanding of how transformers work. You must include a description of your network and the experiments carried out to ascertain the relative performance of the transformers in temperature prediction task. Note that you are using the same dataset as in the recurrent networks experiments.

What needs to be submitted and things to consider

1. if your code does not run, you can only score a maximum of 60% of the maximum marks available for the part of the report corresponding to your code.

2. Prepare a PDF report with sections as specified above. Name the report `report-student_name-student`

3. Prepare a working Python code (**not a jupyter notebook**) named `recurrent.py` which must run on command line as

   ```
   python3 recurrent.py -d jena_climate_2009_2016.csv
   ```

   When your code runs, it must conduct the experiments and print out the results you discussed in the appropriate part of your report.

4. Prepare a working Python code (**not a jupyter notebook**) named `transformer.py` which must run on command line as

```
python3 transformer.py -d jena_climate_2009_2016.csv
```

When your code runs, it must conduct the experiments and print out the results you discussed in the appropriate part of your report.

5. Be careful when writing your report. You will lose marks if your use of English language in your report is unintelligible.

6. You will create a folder and add the three items to be submitted. Archive it in a ".zip" or ".rar" file. Give your archive file your *student-name-student-number.[zip][.rar]*. Submit this archive file.