Total Marks:380 Assignment 2 (RNN/LSTM) Due Date: 12^{th} Mar 2020

Only Pytorch can be used as a DL library. During the demo, you are asked to reproduce the results mentioned in the report(Save weights for all the experiments). Failing of which will fetch zero marks. There will be no deadline extension and extra days for submission

Ques 1. [**120 marks**] Train a sequential model for emotion classification on text. For this, download the dataset from here. This comprises of texts spanned over 13 different classes of emotions. Preprocessing of the data may be required to make it suitable for training. Take first 80% samples for training and remaining 20% samples for validation for all your experiments.

(1) Train an RNN for the above data. Report the training and validation accuracy. Justify and report your choice of hyper-parameters and network architecture.
(2) Train another model based on LSTM for the same split of the data and repeat (1).
(3) Choose your best model out of the above two and make significant novel changes in the model with a proper motivation and justify your choices.
(4) Visualize the model performance with appropriate plots for loss as well as accuracy performance over the epochs.

- Save your best model for (1) and (2) for evaluation and report the accuracies and plots mentioned. [**40 marks (20 + 20)**]
- Save your best model for part (3) so that it can be used during evaluation along with the accuracies and plots. [**30 marks**]
- On the test set, you should be able to generate a '.csv' file with first column as 'ID' and second column as 'Class'. In the 'ID' column mention sample no. (1,2,3, etc) obtained from name of the sample and in 'Class' column mention the class predicted by your model. [**30 marks**]
- Mention in detail the steps taken for preprocessing the data, and all the results obtained in a report. [**15 marks**]
- Along with a tidy code, submit a descriptive README explaining how to train as well as evaluate your model. [**5 marks**]

Ques 2. [**110 marks**] In this question, you will be implementing the concept of Attention in multiple recurrent networks. Attention is a term used to refer to the process of constructing context vectors to capture the entire past information in a given sequence, and leveraging these context vectors to align the sequence outputs with respect to the temporal inputs. Refer to this paper for more details on how to implement attention in recurrent networks.

(1) Train an RNN with attention (using the same formulation as in the above referred paper) on the dataset (using the same data splits) given in question 1. Report the training and validation accuracy.
(2) Train an LSTM with attention (same formulation as in the above part) on the same dataset splits as in the above part. Report the training and validation accuracy.
(3) Visualize the performance using loss plots, accuracy plots and confusion matrices.

- Save your best models(1 LSTM and 1 RNN) for evaluation. [**20+20 marks**]
- In the report, mention your best model's performance on the validation sets (for both RNN with attention and LSTM with attention) in terms of accuracy, ROC plots and confusion matrices. [**30 marks**]
- On the test set, you should be able to generate a '.csv' file with first column as 'ID' and second column as 'Class'. In the 'ID' column mention sample no. (1,2,3, etc) obtained from name of the sample and in 'Class' column mention the class predicted by your model. This part will be gameified (Marks obtained will be a function of the test set accuracy) [**30 marks**]

- Perform the same steps for preprocessing the data. **DO NOT CHANGE YOUR PREPROCESSING STEPS**.
- Along with a tidy code, submit a descriptive README explaining how to train as well as evaluate your model. [**10 marks**]

Ques 3. [**150 marks**] The task is to perform audio classification. Download the dataset from here. The dataset consists of 10 classes. The naming convention of each sample is as follows: $xx\_y.wav$, where 'xx' is sample number, and 'y' is the sample class. For example, 10_1.wav represents 10^{th} sample belonging to class 1. Each class contains 30 samples, from which use samples 1_y.wav to 25_y.wav for training, and 26_y.wav to 30_y.wav for validation.

(1) Visualize 2 random samples for each class with corresponding labels. [**10 marks**]
(2) For given raw audio signal extract MFCC and spectrogram features and visualize them corresponding to raw audio signal (2 features per class). [**10 marks**]
(3) Train a CNN with MFCC and spectrogram features. [**30 marks**]
(4) Train LSTM for raw audio, MFCC and spectrogram features. Compare the performance with the CNN when MFCC and spectrogram is used. [**45 marks**]
(5) Report quantitative analysis (accuracy and confusion matrix) for validation set. [**15 marks**]
(6) During demo you are given a separate test data(format same as training dataset) and you have to report accuracy. Group with maximum performing model will be given full marks. [**30 marks**]
(7) Show the convergence plot. [**10 marks**]