COMP5046 Assignment 1 [20 marks]

Individual Assessment

Deep learning based Sentiment Analysis AI bot

Sentiment analysis is contextual mining of text which identifies and extracts subjective information in source material, and helps a business to understand the social sentiment of their brand, product or service while monitoring online conversations. In this assignment, *you are to implement a sequence-based text classification model to predict the number of positive and negative reviews based on sentiments.*

The detailed information for each implementation step was specified in the following sections. Note that lab exercises would be a good starting point for the assignment. The useful lab exercises are specified in each section.

1. Data Preprocessing [2 Marks]

In this assignment, you are to use the *Large Movie Review Dataset from IMDB*[1], which contains a large set of movie reviews along with their associated binary sentiment polarity labels. With this dataset, you are to do the sentiment analysis - *refer to the Section 3*. Both the training and testing sets are provided in the form of csv files (imdb_train.csv, imdb_test.csv) and can be downloaded from the Google Drive using the provided code in the *Assignment1 Template ipynb*.

In this Data Preprocessing section, you are required to implement the following functions:

- **Preprocess data**: You are asked to pre-process the training set by integrating several text pre-processing techniques (e.g. tokenisation, removing numbers, converting to lowercase, removing stop words, stemming, etc.) - *Please refer to Lab 5.* You should justify the reason why you apply the specific preprocessing techniques. [*Justify your decision*]

2. Model Implementation [9 Marks]

In the 'Model Implementation' section, you are to implement three models: word embedding model, character embedding model, and Sequence model. While the model is being trained, you are required to display the *Training Loss* and the *Number of Epochs*. You are free to choose hyperparameters (size of vector for embeddings, learning rate, epochs, etc.)

1) Word Embedding [2 marks]

First, you are asked to build the word embedding model for your sentiment analysis AI bot as you will use word embedding (word vector representation - such as word2vec-CBOW, word2vec-Skip gram, fastText) as input for the sequence model.

[1] https://ai.stanford.edu/~amaas/data/sentiment/

Note that we used a one-hot vector as an input for the sequence model in the Lab3 and Lab4. In order to build the word embedding model, you are required to implement the following functions:

- **Preprocess data for word embeddings:** You are to use and preprocess imdb data (the one provided in the section 1) for word embeddings - *refer to lab2 and lab3. This can be different from the preprocessing technique that you used in Section 1. You can use both training and testing dataset in order to train the word embedding.* [*Justify your decision*]
- **Build training model for word embeddings:** You are to build the training model for word embeddings. You are required to articulate the hyperparameters you chose (size of vector for embeddings, learning rate, etc.). Note that any word embeddings model (e.g. word2vec-CBOW, word2vec-Skip gram) can be applied. - *refer to lab3 (PyTorch)* [*Justify your decision*]
- **Train model**: You are to implement the model in *PyTorch.* While the model is being trained, you are required to display the *Training Loss* and the *Number of Epochs*.
- **Save model**: You are to save the trained word embedding model to your *Google Drive* -*refer to lab5*. *Note that your assignment 1 will not be marked if you modify the model after the submission.*
- **Load model:** You are to implement a function to load the model saved in your *Google Drive*.

2) Character Embedding [4 marks]

Secondly, you are asked to build the character embedding model as input of the sequence model for your sentiment analysis AI bot. The character embedding model will be implemented by any RNN-based model (Vanilla RNN or LSTM or GRU or Bi-LSTM) -*refer to lab4.* In order to build the character embedding model, you are required to implement the following functions:

- **Preprocess data for character embeddings:** You are to use and preprocess imdb data (the one provided in the section 1) for character embeddings - *This can be different from the preprocessing technique that you used in Section 1. You can use both training and testing dataset in order to train the character embedding.* [*Justify your decision*]
- **Build training model for character embeddings:** You are to build the RNN-based training model (Vanilla RNN or LSTM or GRU or Bi-LSTM) for character embeddings. You are to build the **Many-to-One** (**N** to **One**) sequence model -*refer to lab4 or lab5.* (*e.g.* ***N****: each character 's', 'y', 'd', 'n', 'e', 'y',* ***One****: embedding for word 'sydney'*). Then, extract the last hidden state of the RNN to represent the character embeddings for each word. You are required to describe how hyperparameters (the Number of Epochs, learning rate, etc.) were decided. [*Justify your decision*]
- **Train model**: You are to implement the model in *PyTorch.* While the model is being trained, you are required to display the *Training Loss* and the *Number of Epochs*.

- **Save model**: You are to save the trained character embedding model to your *Google Drive* -*refer to lab5*. *Note that your assignment 1 will not be marked if you modify the model after the submission.*
- **Load model:** You are to implement a function to load the model saved in your *Google Drive*.

3) Sequence Modelling [3 marks]

Thirdly, you are asked to build the Many-to-One (N to 1) Sequence model in order to train a sentiment analysis AI bot. Note that your model should be the best model selected from the evaluation - *refer to section 3 'Evaluation'.* You are required to implement the following functions:

- **Apply/Import Word and Character Embedding models:** You are to apply the trained word embedding model to the sequence model
- **Build training sequence model:** You are to build the RNN-based (Vanilla RNN or LSTM or GRU or Bi-LSTM) Many-to-One (**N to One**) sequence model (**N:** word, **One:** Sentiment - Positive or Negative) -*refer to lab4*. You are required to describe how hyperparameters (the Number of Epochs, learning rate, etc.) were decided. [*Justify your decision*]
- **Train model:** While the model is being trained, you are required to display the *Training Loss* and the *Number of Epochs*.
- **Save model:** You are to save the trained sequence model to your *Google Drive* -*refer to lab5*. *Note that your assignment 1 will not be marked if you modify the model after the submission.*
- **Load model:** You are to implement a function to load the model saved in your *Google Drive*.

3. Evaluation (With Test Set) [3 Marks]

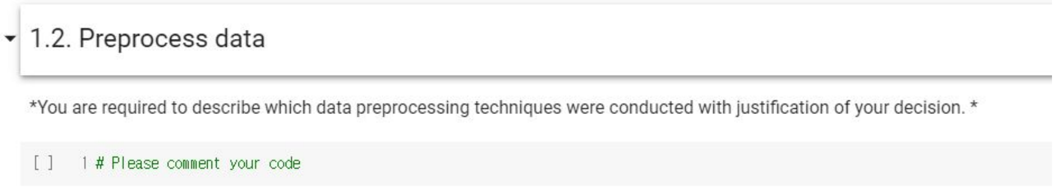
After completing all model training - *refer to the section 1 and 2*, you should apply the trained model to the **test set**. (Note: we just provide only train and test sets. No need to create validation dataset or do the cross-validation)

- **Performance Evaluation**: You are to represent the precision, recall, and f1 - *refer to lab4* of your model in the table [*Explain the performance*]
- **Hyperparameter Testing**: You are to provide the line graph, which shows the hyperparameter testing (with the test dataset) and explain the optimal number of epochs based on the learning rate you choose. You can have multiple graphs with different learning rates. In the graph, the x-axis would be # of epoch and the y-axis would be the f1. [*Explain the performance*]

4. Documentation [4 Marks]

In the section 1,2, and 3, you are required to describe and justify any decisions you made for the final implementation. You can find the tag '[*Justify your decision*]' for the point that you should justify the purpose of applying the specific technique/model.

For example, for section 1 ***(preprocess data),*** you need to describe which pre-processing techniques (removing numbers, converting to lowercase, removing stop words, stemming, etc.) were conducted and justify your decision (the purpose of choosing a specific pre-processing techniques, and benefit of using that technique or the integration of techniques for your AI) in ***your ipynb file***



1.2. Preprocess data

*You are required to describe which data preprocessing techniques were conducted with justification of your decision. *

```
[ ]  # Please comment your code
```

Figure3. The position of writing justification in ipynb file

5. Programming (coding) styles [2 Marks]

Your program needs to be easily readable and well commented. The followings are expected to be satisfied:

- **Readability**: Easy to read and maintain
- **Consistency & Naming**: Names are consistent in style
- **Coding Comments**: Comments clarify meaning where needed
- **Robustness**: Handles erroneous or unexpected input

Assignment 1 Submission Method

Submit a ipynb file that contains all above (1,2,3,4 and 5) contents.The ipynb template can be found in the Assignment 1 template. You also need to submit a word embedding model (zip file), character embedding model (zip file) and sequence model (zip file).
Due date: 5:00PM Friday 24 April 2020
Submission: Canvas Assignment 1 Submission Box
Submission Files:

- **ipynb file -** (file name: *your_unikey*_COMP5046_Ass1.ipynb)
- **zip file - word embedding model** (file name: *your_unikey*_word.zip)
- **zip file - character embedding model** (file name: *your_unikey*_character.zip)
- **zip file - sequence model** (file name: *your_unikey*_sequence.zip)