Take Home Project #1 (25 Mark)

IS6751 Text & Web Mining

Due on 19 March 2025

1. **Task 1: Build a Sentiment Classifier with the following requirements.**
   - **1.1. Dataset**
     - Use **reviews_with_splits_lite.csv** in mlp-sentiment\data\yelp.
     - Do not change split values in the file since everyone should use the same test data.
   - **1.2. Algorithm**
     - Use code in **3_5_Classifying_Yelp_Review_Sentiment.ipynb** in mlp-sentiment as a baseline model
     - Use only one or two Jupyter notebook files for Task 1.
     - Use only **Multilayer Neural Networks**. You must not use other deep learning models, such as RNN, LSTM, and CNN.
   - **1.3. Improve the model using following approaches**
     - Text pre-processing, such as removing special characters, removing stop words, and using lemmatization, case folding, unigram + bigram words, sentiment lexicon (e.g., # of positive words and # of negative words), etc.
     - Change hyperparameters, such as learning rate, # of hidden units, mini-batch size, # of layers, dropout, batch norm, regularization, etc.
     - Apply any other techniques and modify any program statements if you want.
     - Write a report in Word or PDF that discusses your observations, such as test results (accuracy and loss values) with various approaches (up to 3 pages). Note that you should report **at least the test results of your five best models** (note that each model refers to a version with different hyperparameters.).

2. **Task 2: Build a Surname Classifier with the following requirements.**
   - **2.1. Dataset**
     - Use **surnames_with_splits.csv** in mlp_surnames\data\surnames.
     - Do not change split values in the file since everyone should use the same test data.
   - **2.2. Algorithm**
     - Use code in **4_2_Classifying_Surnames_with_an_MLP.ipynb** in mlp_surnames as a baseline model
     - Use only one or two Jupyter notebook files for Task 2.
     - Use only **Multilayer Neural Networks**. You must not use other deep learning models, such as RNN, LSTM, and CNN.
   - **2.3. Improve the model using following approaches.**
     - Text pre-processing, such as case folding, multi-gram characters (not multi-gram words since surname contains only one word), etc.
     - Change hyperparameters, such as learning rate, # of hidden units, mini-batch size, # of layers, dropout, batch norm, regularization, etc.
     - Apply any other techniques and modify any program statements if you want.
     - Write a report in Word or PDF that discusses your observations, such as test results (accuracy and loss values) with various approaches (up to 3 pages). Note that you should report **at least the test results of your five best models** (note that each model refers to a version with different hyperparameters.).

Submission:

Submit **one zip file** (use **only zip** compression file), named **take-home-project-no-1-yourname.zip**, that contains **your report file, Jupyter Notebook files, data files (i.e., input data) and the best model files (i.e., the best model.pth for Tasks 1 & 2)** through **Turnitin** on the class website.

- Use one report file for Tasks 1 & 2 with a cover page.
- The Jupyter notebook files must show all output results of **your best model of Tasks 1 and 2**. So please make sure that you run all the cells in the notebook files before your submission.

- Note that Turnitin does not allow you to resubmit your assignment file.
- Reports and required files submitted in after the due date will be marked down by **10% per day**.