CSE 842: NATURAL LANGUAGE PROCESSING
HOMEWORK 3 ASSIGNMENT

Due Date: November 20, 2024 at 11:59 PM

OVERVIEW

For this assignment, I want everyone to become familiar with working with BERT. I have linked to tutorials taken from HuggingFace below. The HuggingFace Tutorials are linked to Google Collab, so you can use Collab's GPUs to run your code. I've specified BERT throughout the assignment. However, you may use any BERT variant you prefer. RoBERTa works really well for NLP tasks; DistilBERT is a smaller, faster version of BERT. You can use PyTorch or Tensorflow.

PROBLEM 1: FINE-TUNE A PRETRAINED MODEL

Description: For this problem, you are to walk through the HuggingFace tutorial for fine tuning: Fine Tuning a Pretrained Model. The tutorial has sections for both PyTorch & TensorFlow. This is a simple fine-tuning approach that will not require you to alter the transformer layers of the pretrained model.

What to submit:

1. Code: Completed Python file of the tutorial
2. Written Report: A brief description including:
    a. the package you used (PyTorch, TensorFlow)
    b. which model you used (The tutorial uses *bert-base-cased*, you should replace this with a BERT model of your choosing. I suggest running the tutorial as is first.)
    c. brief description of model (model.summary( ))
    d. your results
    e. something new or interesting that you learned during implementation.

PROBLEM 2: OUTPERFORM BERT

Description: For this problem, you can choose to solve any NLP task we've discussed this semester. For example: PoS tagging, NER, sentiment classification, topic classification, QA, MT, summarization, SRL, etc. Your task is to develop a model (or modeling pipeline) and relevant features to try to outperform the current best BERT-related-model performance on your chosen task. *Your limitation is: you can't use BERT, transformers, encoders, or neural networks of any kind.* If you want to use word embeddings, you can use word2vec, GloVe, or ELMO.

What to submit:

1. Code: Python file of your model.
2. Written Report: Provide instructions on how to run your code as in previous homework assignments. Provide details on which task, features, model, etc. you chose and ***why*** you chose them. Include the following:

    a. Your chosen NLP task and why you wanted to study it.

b. The dataset you are using and why you chose it for this task.
   (For example, if you want to work with NER tagging, you probably don't want to choose a Twitter dataset labeled for sentiment.)
c. The model(s) you used. You may use any non-NN model: logistic regression, SVM, HMM, CRF, etc. You do not need to implement these from scratch. You can use scikit-learn or any other packages you need. However, do not use a pre-existing paper's entire code base including model & features.
d. The features you tested. For (c) & (d), I would suggest looking up the previous (not NN) state-of-the-art model for your task and using that to initialize your ideas. Link to that paper if you use this approach.
e. Link to the most recent, best-performing BERT (or related) model for this task. Include a table summary of their results (from their paper).
f. Summary table of your results (precision, recall, F1, accuracy).
g. Brief discussion of:
   i. where you think your modeling approach and features worked well
   ii. where you might have been able to improve
   iii. why you think your model did or did not outperform the BERT model of (e).
h. *Optional for 5 bonus points:* Add BERT contextual embeddings into your features and report/discuss your new results.
i. *Optional for 5 bonus points:* Replace your model in (c) with a bidirectional LSTM. Report/discuss your new results.

Problem 3: Last 10 Points

Problem 1 is worth 10 points. Problem 2 is worth 80. To get the remaining 10 points, you must fine-tune BERT (or variant) to solve your task. You may want to limit your task selection to those that HuggingFace provides an *AutoModelFor-Task-* for you to use. This step may be challenging or time consuming depending on your prior NN experience. So I've provided the 2 bonus opportunities above (2h & 2i).
If you complete everything, any excess bonus points can be applied to elsewhere in your grades.

Submission Summary

Please submit your Python files and *one* written report in **PDF** format.
Please zip your files together and include your last name in the zipped filename.