CS 4650 Fall 2021: Homework 4

October 13, 2021

Instructions

- For this homework, you will code a POS tagger and experiment with incorporating pre-trained word embeddings.
- We generally encourage collaboration with other students. You may discuss the questions and potential directions for solving them with another student. However, you need to write your own solutions and code separately, and not as a group activity. Please list the students you collaborated with on the submission site.
- Upload 'data_utils.py' and 'lstm_model.py'to HW4 Code on Gradescope.

  You will also need to attach a pdf export of 'tagging.ipynb', including outputs, to your writeup, as well as copying outputs from other iPython notebooks into your write-up for Q1.
- Our deep-learning recitations will cover much of the content in this homework, so refer to those notes and recordings.
- Note: This is a large class and Gradescope's assignment segmentation features are essential. Failure to follow these instructions may result in parts of your assignment not being graded. We will not entertain regrading requests for failure to follow instructions.
- LaTeX solutions are strongly encouraged (a solution template is available on the class website), but scanned handwritten copies are also acceptable. Hard copies are not accepted.
- We generally encourage collaboration with other students. You may discuss the questions and potential directions for solving them with another student. However, you need to write your own solutions and code separately, and not as a group activity. Please list the students you collaborated with on the submission site.

Questions

1. In this assignment, you will implement a part-of-speech tagging model. You will first implement a word LSTM tagger and the necessary data handlers to create a vocabulary and set of labels. Then, you'll train your part-of-speech tagger on the Treebank dataset from NLTK. Afterwards, you'll change different pieces of the training pipeline and analyze their effects on training.

   You can download the assignment zip here:
   https://www.cc.gatech.edu/classes/AY2022/cs4650_fall/programming/h4_pos.zip

   (a) Open up 'tagging.ipynb', this will guide you through the assignment.
   - First, you will need to implement the dataset functionalities in 'data_utils.py'.
   - Next, you'll need to implement the LSTM in lstm_model.py, as well as the loss function and the training/validation loop code.
   - Once everything is finished, complete the notebook and observe your training/validation curves
   - **Include plots and final performance numbers in your write-up for 1a.**

   (b) After completing the notebook and code, set the 'SHUFFLE' flag to 'True', and revisit your dataset-splitting function to shuffle the dataset before splitting into train/validation data.
   - **Include final performance plots and numbers in your write-up for 1b.**
   - **How did the performance change?**
   - **Why did the performance change this way?**

   (c) After completing the notebook and code, set the 'PRE_TRAINED' flag to 'True', and revisit your LSTM to take in a set of pre-trained embeddings.
   - Run through the notebook again and plot training/dev performance.
   - **Include plots and final performance numbers in your write-up for 1c.**
   - **How did the performance change?**
   - **Why did the performance change this way?**

   (d) Revisit the hyper-parameters defined at the top of the notebook, and set the PRE_TRAINED flag to FALSE before doing the following:
   - Experiment with the sequence length (increase and decrease it). Re-run the notebook with your changes.
   - **Include final performance plots and numbers in your write-up for 1d.**
   - **How did the performance change?**
   - **Why did the performance change this way?**

(e) Revisit the hyper-parameters defined at the top of the notebook.

- Experiment with the batch size (increase and decrease it). Re-run the notebook with your changes.
- **Include final performance plots and numbers in your write-up for 1e.**
- **How did the performance change?**
- **Why did the performance change this way?**

(f) [Bonus] Revisit the hyper-parameters defined at the top of the notebook.

- Choose 2 LSTM parameters (embedding dimension, number of layers, activation functions, RNN type, etc.). Re-run the notebook with your changes.
- **State what you changed and how you changed it. Include final performance plots and numbers in your write-up for 1f.**
- **How did the performance change?**
- **Why did the performance change this way?**