DL4NLP - Assignment 1: RNNs and Variants

Total points: 50
Due date: October 21

1 Assignment Goal

In this assignment, you will train and evaluate RNN, LSTM and GRU models, on the task of predicting the sentiment of a text (IMDB rreviews).

2 Dataset

The dataset you will use to train and test your models is available at: https://www.cs.cornell.edu/people/pabo/movie-review-data/ under the "Sentiment polarity datasets" section and precisely the dataset under the 'sentence polarity dataset v1.0' link. Please carefully read the readme on the same page. The dataset contains 5,331 positive and 5,331 negative processed sentences / snippets. Please download the dataset and create a train/dev/test split as follows: 60% of the original data goes into the training set, 20% of the original data goes into the dev set, and 20% of the original data goes into the test set. Please ensure that all your train/dev/test are balanced (i.e., please ensure the number of positive examples is equal to the number of negative examples in all train/dev/test).

3 Tasks

1. **Load and preprocess the data**

   We will use word-level representations for your models, specifically, use the glove word embeddings available from https://nlp.stanford.edu/projects/glove/. Please use the "glove.6B.zip" version trained on "Wikipedia 2014 + Gigaword 5" and 300d word vectors.

2. **Implement RNN, LSTM, and GRU models for the sentiment classification task.**

   - **Create the Model/Network:** Define models for vanilla RNN, LSTM and GRU networks, respectively.

- **Train the Model:** Train your models using train. You are free to choose learning parameters (sequence length, learning rate, batch size, stopping criteria, etc.). Plot the loss and accuracy versus the number of iterations for both train and dev subsets and choose the parameter setting that works best on the dev subset.
- **Experiment with Network Structure:** We want to explore how a network performs when we change its structure. For the sentence encoding, use the element-wise max of all hidden states or the final hidden state (one or the other). In the latter case, for the sentence encoding, use the hidden state corresponding to the last word in the sentence (thus, excluding hidden states corresponding to padding). Please note that you are not required to do both element-wise max or final hidden state. Choose either one of the two options.
  - Number of hidden units. Try doubling and halving your number of hidden units. Like above, plot the loss and accuracy versus the number of iterations for both train and dev subsets. Discuss your findings.
  - Sequence length. Try doubling and halving your length of sequence that feeds into the network. Plot the loss and accuracy versus the number of iterations for both train and dev subsets. Discuss your findings.
  - Experiment with one word embedding dimension (i.e., 300d) using pre-trained and fine-tuned word embeddings. Plot the loss and accuracy versus the number of iterations for both train and dev subsets. Discuss your findings.

Please note, you can find the hyper-parameters that work best for vanilla RNN and use those for LSTM and GRU. It should suffice for the max grade. However, if you would like, you can tune hyper-parameters for all RNNs variants.

3. **Evaluate the models:** Evaluate your best vanilla RNN, LSTM and GRU models on the test data and compare the results of the three models. Use precision, recall, and F1-measure to report the performance on the test set. Offer your intuition, high-level and briefly, behind any observed difference in performance between the models.

4 What to submit

Submit a Jupiter Notebook containing your code and results/discussion, or python code together with a report file showing the results/discussion. Submit your homework in blackboard.