Natural Language Understanding with Distributed Representations - Assignment 2

In this assignment, you will do binary sentiment classification task on IMDB movie review dataset[1], with different model architectures.

1 Dataset Preparation

Dataset can be downloaded at `http://ai.stanford.edu/~amaas/data/sentiment/`, and we will only use labeled data here. The original training set and testing set each contains 25,000 movie reviews, half negative and half positive.

You should divide original training set into Training and Validation, and tune your models on these sets. Original testing set should only be used to report the performance from tuned model.

We will only use the most frequent 10k words as our dictionary, and words not in the dictionary should be mapped to '<oov>' (out of vocabulary).

2 Continuous Bag-of-Word classification

Implement Continuous Bag-of-Word (CBOW)[2, 3] model, and use it to do sentiment classification on IMDB dataset. You are not required to use pretrained word vector representation, but feel free to if you would like to try. Please note that, in the original paper, the model is used in language modeling, but the architecture could easily be used to do classification with changing the target from word to sentiment label.

3 FastText

Improve your original CBOW model by adding n-gram features. This is FastText model.[3] If you have time, try to experiment with different 'n', and see what effect does it have on the performance. Please note that, as we only have 2 target classes, hierarchical softmax, which is used to perform approximation for softmax when the number of target classes is high, doesn't need to be implemented as in the paper.

4 FastText tool from Facebook

Facebook released their implementation for FastText, and you can find it from Facebook's Github: `https://github.com/facebookresearch/fastText`. Use their tool to solve the same task, and report the performance you have.

5 Write-up

The deadline is *Monday, October 17th*. Please submit your report and corresponding code to dl4nlp2016@gmail.com.

The paper ($<$ 3 pages) should consist of a:

- description of data preprocessing (train/validation split)

- description of the architecture (number and type of layers, number of hidden layers, size of word embedding matrix, etc.)
- description of training procedure (what optimization method?(Adam, SGD, Adadelta, etc.), learning rate, used dropout?, train/valid/test error, etc.)

References

[1] Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. (2011). Learning Word Vectors for Sentiment Analysis. The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).

[2] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv preprint arXiv:1301.3781.

[3] Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of Tricks for Efficient Text Classification. arXiv preprint arXiv:1607.01759.