**Evaluation of various deep learning models for sentiment analysis**

You are given the reviews dataset. These are 194439 amazon reviews for cell phones and accessories taken from <https://jmcauley.ucsd.edu/data/amazon/> Use the “reviewText” and “overall” fields from this file. The goal is to predict the rating given the review by modeling it as a multi-class classification problem.

1. Take the first 70% dataset for train, next 10% for validation/development, and remaining 20% for test.
2. Traditional machine learning methods
   1. Design some good linguistic features. You can start with basic TFIDF features. Use these classifiers: J48 decision trees, SVMs with linear/RBF kernel, logistic regression, xgboost, random forests and report accuracy on test set.
3. Average of word embeddings
   1. Learn word2vec models using gensim on this dataset with the following settings: (a) Size=100, 200, 300, (b) Window=3,7, (c) Min\_count=2, 5. Use skipgram.
      1. This will give 12 word2vec models. For each of these models, for each review take average word embeddings and train a logistic regression. Report accuracy on test set.
   2. Use the already available google word2vec model. For each review take average word embeddings and train a logistic regression. Report accuracy on test set.
   3. Use the already available glove models – 50D, 100D and 200D. For each review take average word embeddings and train a logistic regression. Report accuracy on test set for each of the three sized embeddings.
4. Recurrent neural networks
   1. RNNs: Train a single directional RNN with L layers. Vary the number of layers (as 1,2,3,4) and also size of layers (20, 50, 100, 200). Report accuracy on test set.
   2. LSTMs: Train a single directional LSTM with L layers. Vary the number of layers (as 1,2,3,4) and also size of layers (20, 50, 100, 200). Report accuracy on test set.
   3. BiLSTM: Train a single directional RNN with L layers. Vary the number of layers (as 1,2,3,4) and also size of layers (20, 50, 100, 200). Report accuracy on test set.
5. BERT
   1. Use the pretrained 12-layer BERT model. Use input size as 128. Use Batch size: 16. Finetune BERT for few (say 5) epochs on train data. Report accuracy on test set. Also, track variation in train loss and validation loss to ensure there is no overfitting.

Finally, submit a 4-page project report summarizing your findings with respect to the accuracy obtained using various models you tried.