

# NLP SEE Day 1 Evaluation

Dec 19, 2016, 8:30 pm to 6:30 pm

## Objective

As a part of the SEE hands on evaluation, we will be working on developing a fine grain sentiment analyzer. We will be using the Stanford Sentiment Treebank dataset. Our goal is to do the following:

- (a) Using the Sentiment Treebank dataset, learn a 3 way classifier (Positive, Negative, Neutral) and a 5 way classifier (Very Positive, Positive, Neutral, Negative, Very Negative). We will learn this using 2 classifiers: Recurrent and Recursive.
- (b) Evaluate the performance and experiment with transfer learning

This question paper is meant for day 1 and morning session of day 2. You will receive the question paper for the second half of the evaluation on day 2.

## Steps

### (a) Dataset

- i. You are required to download the dataset from:  
<http://nlp.stanford.edu/sentiment/index.html>
- ii. Go through the README file to understand the details, format of the dataset.
- iii. You can use the code from  
<https://github.com/JonathanRaiman/pytreebank> for dataset processing
- iv. You are also required to download the training and test datasets in PTB format, please go through the details from the github URL as above.
- v. You are required to use the training, dev (validation) and test data as specified in the README file, do not mix this up.

### (b) Building RNN

- i. You can build a classifier using regular RNN with GRU or LSTM. First build it with plain RNN and later enhance it to GRU or LSTM. You may also try BiDi RNN.
- ii. Perform any text preprocessing, convert these to vectors using Word2Vec.
- iii. As the text field has many lines, you may clip the number of sentences to 1 or 2 and place an upper limit on number of words. Otherwise the RNN might encounter vanishing gradient problem

- iv. You can build a GRU (time permitting, optional) in order to get a better result
- (c) Discretize the expected output from continuous to discrete. Create 2 sets of 5 labels and 3 labels. Look at the Stanford site for suggestion on mapping continuous target to a discrete representation.
- (d) Train the system and validate. Till now you can treat the phrases as a list of inputs and the corresponding sentiments as targets. First try a 3 state system and then build a 5 state system
- (e) Measure the accuracy, precision, recall for each of the classifiers (3 state and 5 state) and report.
- (f) Use a parser (Either NLTK Parser or the Stanford Core NLP Parser) to parse a given input into a tree structure.
- (g) Predict the sentiment for each sub tree in the parse output and annotate the tree with sentiment score. Use the treebank dev and test data to check this.
- (h) Establish a baseline performance and report.

## Deliverables

Submit the following by 6:30 pm, 19<sup>th</sup> Dec 2016:

1. Source code of RNN, sentiment analyzer
2. Screen shots of results for 3 and 5 way classifier

Throughout your work, make sure you show the outputs to the invigilators.

Best wishes from the faculty, enjoy NLP development!

Anantharaman Palacode Narayana Iyer, Chitra GM