

## Assignment 7: Constituency Parser

*Instructor:* Prasenjit Majumder

**Learning Outcome:** At the end of this assignment you will POS tagger. And evaluating it using precision, recall and F1 score

### 1 Problem description

Constituency parsing aims to extract a constituency-based parse tree from a sentence that represents its syntactic structure according to a phrase structure grammar.

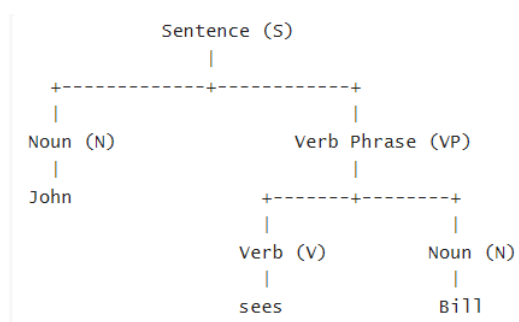


Figure 1: Example of Parsed Sentence

Recent approaches convert the parse tree into a sequence following a depth-first traversal in order to be able to apply sequence-to-sequence models to it. The linearized version of the above parse tree looks as follows: (S (N) (VP V N)).

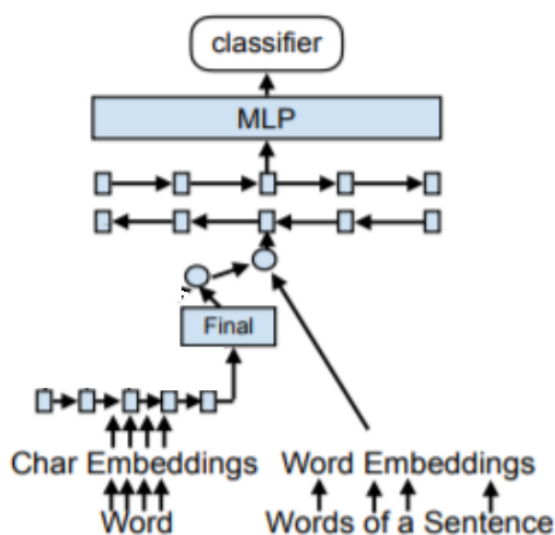


Figure 2: Architecture of POS tagger

## 2 Implementation

### 2.1 Dataset

- We will use the Penn tree bank dataset.
- In order to perform batching and converting to vocab ids. You can directly use any of the below urls:
  - <https://pytorch.org/text/0.8.1/datasets.html#penntreebank>
  - [https://pytorchnlp.readthedocs.io/en/latest/\\_modules/torchnlp/datasets/penn\\_treebank.html](https://pytorchnlp.readthedocs.io/en/latest/_modules/torchnlp/datasets/penn_treebank.html)

### 2.2 Exercise

- Use the character of each word and pass it through an unidirectional LSTM (char-LSTM)
- Represent each word using Word2Vec
- Concatenate the Output vector of char-LSTM and Word vector of word to represent vector of each word.
- Pass the words in a sentence through Bidirectional LSTM
- Use the hidden vector of each word and pass it through an MLP and apply softmax to get the POS tag of each of the word
- Use Cross Entropy as the loss function and Adam optimizer for optimization
- Report the Precision, Recall and F1-score

$$Recall = \frac{(\# \text{ correct constituents in candidate})}{(\# \text{ constituents in gold tree})} \quad (1)$$

$$Precision : \frac{(\# \text{ correct constituents in candidate})}{(\# \text{ constituents in candidate})} \quad (2)$$

$$F1 = \frac{(2 * precision * recall)}{(precision + recall)} \quad (3)$$

## 3 References

- <https://github.com/qGentry/MetaBiLSTM>
- <https://web.stanford.edu/~jurafsky/slp3/13.pdf>
- <https://aclanthology.org/P18-1246.pdf>

## 4 Submission

- You have to submit your assignment in notebook with proper comments and explanation of your approach.
- Report the Precision, Recall and F1=score for the approach
- The submission deadline for this assignment is **11th October 2021 11 pm**