Natural Language Understanding
Assignment 3

Ponsuganth Ilangovan P
ponsuganthp@iisc.ac.in

1 Problem Statement

To estimate the PCFG probabilities from the Penn Treebank Dataset and to build a CKY parser which takes in a sentence and gives then POS tags for the sentence. The parser is also to be evaluated with the precision and recall of the parsed tags with the correct tags.

2 Methodology

CKY parsing is a classical algorithm for POS tagging. In CKY parsing we need to estimate the probabilities for the rules of grammar in our dataset. The rules of grammar are obtained from the dataset and the probabilities are found. With this probabilities all then most probable trees are formed. At the root we choose the most probable tree and back point to our resulting tree which at the leaf will give our POS tag. Before starting the parsing process out train set parse tree has to be normalised by which the rules should only go to two child and not more and also unary rules are removes by collapsing such as when NNP goes to NP and NP goes to 'some word' we collapse and write NNP goes to 'some word'. This is an important preprocessing step to make the algorithm work in cubic time.

The above figure shows the data structure to be used for a CKY parser. The diagonal elements form the words and the leaf rules which will be our POS tags. First we collect rules that are applicable for our words. Then we build our tree in a bottom up fashion. At the second level the rules can point to only two childs. Here is where the binarisation of rules by the chomsky normal form is very useful. By this way we continue up the tree and also make sure that we choose the child boxes in such a way that there is no clashed while backtracking. While building the tree we will be multiplying the correspondin probabilities and at

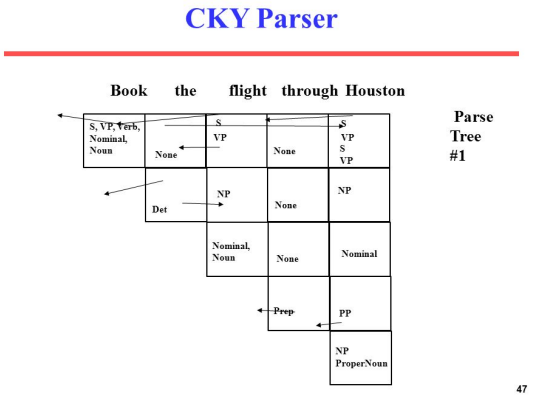


Figure 1: CKY Parse table

the root we choose the rule with the most probable probability and backtrack back to find the tree and the POS tag of the sentence.

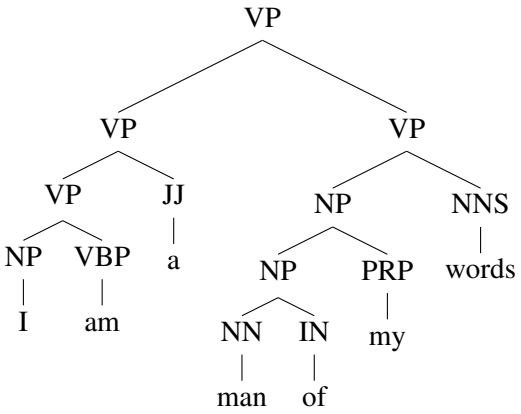
3 Implementation details

The PCFG probabilities were estimate from the penn treebank dataset obtaned through the nltk interface. The rules were collected after normalising the trees and the occurrence counts were used to estimate the probabilities. For the leaf nodes to get a comaparable distribution amongst other trees the leaf node probabilities are conditioned on words. The rules are choosen by querying corresponding words from the collected rules and the trees are built up. When parsing sentences due to the shortage of rules there are some errors while parsing and those sentences are excepted from parsing in testing.

3.1 Smoothing

Smoothing is only applied at the lowest level. When the parser encounters an unknown word then some probability mass should be assigned for it to choose a tag for that rule. This probability is

decided by collecting the tags at the last level and estimating its occurence probability which can be said as unigram probability of that tag. From these probabilities we create a distribution for the tags at the last level. When an unknown word is encountered we sample a tag from this distribution and use that tag to build up the trees. Smoothing in other levels can be applied but when the rules are permuted the complexity goes up and hence it was not done. But the method of creating a distribution of the rules can be used at each level but it was not attempted.



4 Experiments and Results

The penn treebank dataset was imported with the nltk package. The train and test sets were split from the whole dataset. Around 3000 sentences were used to train the parser and around 100 sentences to test the parser.

4.1 Task 1

The PCFG probabilities were estimated with occurence counts and they were used for parsing. Sentences for which the parser was not able to parse were skipped and the precision, recall and f1 score were calculated for the leaf POS tag.

Precision	Recall	F1 Score
0.10435	0.08891	0.08944

Table 1: Results of training

4.2 Task 2

While experimenting with the parser it was found that the parser failed for the following sentences

What this tells us is that he is not interested
That is my friend

The parser threw an error saying that it was not able to build the whole tree upto the root i.e the root box is empty. When investigated it was found that 'What' was mapped to 'WP: WH-pronoun' and this was mapped to 'DT: determiner' but there was no rule in the grammar which could map these two hence the parser was not able to build the complete tree. When a dummy rule '*NP –>WP DT*' was added the parser was able to parse the sentence. Similar debugging was done for the other sentence and the parse was obtained. The following is a made up test sentence