E1 246: Natural Language Understanding (2019)
Assignment-3: CYK parser using PCFG

Abhishek Kumar
abhishekkumar@iisc.ac.in

Abstract

This documents contains the final report of 3^{rd} Assignment of Course: $E1-246$. In this assignment I have build a model for parsing which parses a given sentence or text file using CYK parser with the help of PCFG rules. I have experimented with PCFGs and Constituency Parsing and this document contains the various details,charts and results of those experiments.

1 Introduction

A Probabilistic Context-Free Grammar (PCFG) is simply a Context-Free Grammar with probabilities assigned to the rules such that the sum of all probabilities for all rules expanding the same non-terminal is equal to one. PCFGs extend context-free grammars similar to how hidden Markov models extend regular grammars. Each production is assigned a probability. The probability of a derivation (parse) is the product of the probabilities of the productions used in that derivation.

2 Datasets

For this task I have used Penn Treebank data available with NLTK.
NLTK provides access to 10% sample of the Penn Treebank. Training file contains 80% of the Treebank data and Testing dataset conatins 20% of the Treebank data.

• There were $24,937$ production rules (CNF Rules) on which this model is trained to calculate PCFGF rules.

3 Model

The model contains the probabilistic CFG rules which I have induced using the CFG rules provided by the treebank.
It learns from a set of CFG productions rules and constructs a weighted grammar with the ML estimation of the production distributions i.e., Probabilities are calculated using relative frequency estimation: The probability of a production $X->a$ in a PCFG is:
$P(X->a) = count(X->a)/count(X)$
• All the CFG production rules were first converted into CNF rules.
• All the unary production rules were collapsed before calculating the probabilities.

I have added out of vocabulary words i.e., new words seen by the model during validation sets (10% of the test set) to the grammar as the noun production.

4 EVALUATION

I have applied add-1 Smoothing and Jelinek-Mercer smoothing (interpolation) in this model. I am presenting my evaluation stats for both model seperately.

4.1 Interpolation Smoothing

To implement interpolation smoothing I have assigned the same probabilities to each unknown word production to that of average probability of that concerned non-terminal.

The model evaluated using the Interpolation smoothing achieve the following results:

METRIC	**VALUE**
Labeled Precision	28.34
Labeled Recall	38.47
F1	32.60

Table 1: Interpolation smoothing

4.2 Add-1 smoothing

Standard laplace smoothing is applied in this model to learn probabilities of the production. I have increased the count of each production rules by 1 before learning the PCFG rules.

METRIC	VALUE
Labeled Precision	25.46
Labeled Recall	37.13
F1	30.208

Table 2: Add-1 smoothing

5 Model comparison

I have compared my best parser with these existing parsers available online

- Berkeley Parser
- Stanford parser

5.1 Short sentence

I have used these short sentences to evaluate my model against the online parser for which my parser is giving wrong answer while the online parsers are giving correct parse tree.

- Old men and women.
- Look at the dog with one eye.

my model is not able to pasre these sentences probably due to low training data and also very basic smoothing methods are applied to learn the probabilities due to which it generalizes very poorly.

5.2 Long sentence

- Mary ate a salad with spinach from Califonia for lunch on Tuesday.
- I saw a man on a hill with a telescope. same problem as with the short sentences.

Since most of the words are unseen, they are replaced by $< UNK >$ word due to which each unknown words related production rules were given same probabilities.

6 Conclusion

The parser is generalizing very poorly on unknown words and texts because of the following reasons:

- Unseen words are treated as same in every case.
- Many times a word can act as verb phrase(VP) as well as noun phrase (NP) depending on the context and the relative positioning of that word in the sentence but our model cannot differentiate between due to no contextual sense.

We can improve the accuracy of our model further by implementing following points:

- Using larger training datasets to train our model.
- Implementing better smoothing and interpolation technique.
- Implementing contextual sensitive rules in our parser to deal with contextual ambiguity.

7 Refrences

Wikipedia article on:
CYK algorithm
Probabilistic context-free grammar
NLTK (http://www.nltk.org/howto/corpus.html).
Linguistics 165, Professor Roger Levy's lecture notes