BBM497: Introduction to Natural Language Processing Lab. **(Due: 04/05/2020)**

Submission Assignment #3

Instructor: Burcu CAN & Necva BÖLÜCÜ *Name:* Anıl Aydıngün, *Netid:* 21526676

1 Introduction

In formal language theory, a context-free grammar (CFG) is a formal grammar in which every production rule is of the form

A $\rightarrow a$

where "A" is a single nonterminal symbol, and "a" is a string of terminals and/or nonterminals (a can be empty). A formal grammar is considered "context free" when its production rules can be applied regardless of the context of a nonterminal. No matter which symbols surround it, the single nonterminal on the left hand side can always be replaced by the right hand side

2 Data Structures

2.1 Read Dataset and Preprocess Steps

The dataset given in this paper is a type of grammar rules. The format of the CFG rules is as follows:

S NP VP, S –> NP VP
Noun book, Noun –> book

As we read the dataset, we take it as a sentence until the empty line arrives. We ignore the comment lines in the dataset. We split left hand side and right hand side, put them in the dictionary. The key is left hand side and its values of the dictionary are non-terminals or terminals in the right hand side. All of the terminals started with a lowercase letter. Accordingly, it was checked when generating sentences.

2.2 Generating Random Sentences

In the generating process, we use the 'rules' dictionary we keep while parsing the dataset. The generate process starts from root and follows the rules in an iterative way. This process continues until you arrive at the terminals. Rules are chosen randomly. And when it comes to terminals, terminals are selected randomly and sentence is created. In another way, sentences are generated by selecting random words only from terminals. We will put a limit on the number of words in the sentence we generate and see what tend there is between the sentences generated.

Generated sentences with by rules randomly :
"*is it true that it is me ?*"
"*you ate the old fine delicious mouse .*"
"*is it true that it need it ?*"
"*this president like this beautiful beautiful mouse !*"

Generated sentences with by words randomly :
"*with you on to every pickled to this .*"
"*from a you in president it washed sandwich !*"
"*it president to pickled mouse me every prefer .*"
"*beautiful on mouse on the a it that !*"

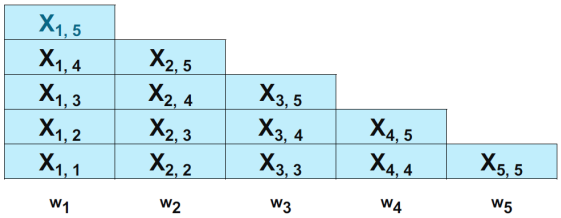
When we look at the sentences generated according to the rules on the left, we set the length of the sentence here "8". We have seen the sentences that start with "is it true that". For the rest, we have seen "Pronoun, Verb, Pronoun." Apart from that, we have seen different sentences. Since the sentence generated is grammatical, it is grammatically correct, but semantically, most are not.

When we look at the sentences generated according to the words on the right, we have determined the length of the sentence here "8". When we looked at it, we understood that the sentences were not grammatically correct since we randomly generated them. We have observed that there is no logical generation.

2.3 CYK Parser

In computer science, the Cocke–Younger–Kasami algorithm (alternatively called CYK, or CKY) is a parsing algorithm for context-free grammars. It employs bottom-up parsing and dynamic programming.

The Cocke–Younger–Kasami-Algorithm (CYK or CKY) is a highly efficient parsing algorithm for context-free grammars. This makes it ideal to decide the word-problem for context-free grammars, given in Chomsky normal form (CNF). The following tool can be used to check if a certain word $\mathbf{w} \in E$ is part of a language, given in CNF grammar.



In the implementation of the CYK algorithm, it starts by considering the table above and initializing the empty table. Then the table is filled in a bottom-up manner. While filling the table, we get the cell we will fill in at the moment from the information and rules in the bottom lines. And we look at the match. We try to find the left hand side from the right hand side. It evaluates the cells individually and deals with all states and sets for each cell. In the end of the algorithm, if the last element of table contains "S" the input belongs to the grammar.

function CYK($G = (N, \Sigma, S, P), w \in \Sigma^*$): $\{false, true\}$
Precondition: G is in the Chomsky Normal Form

1. $n \leftarrow |w|$
2. **for all** $i = 1, \ldots, n$ **do** $N_{i,1} \leftarrow \{A \in N \mid (A, w_{i,1}) \in P\}$
3. **for all** $j = 2, \ldots, n$ **do**
4. **for all** $i = 1, \ldots, n - j + 1$ **do**
5. $N_{i,j} \leftarrow \varnothing$
6. **for all** $k = 1, \ldots, j - 1$ **do**
7. $N_{i,j} \leftarrow N_{i,j} \cup \{A \in N \mid (A, BC) \in P, B \in N_{i,k}, C \in N_{i+k,j-k}\}$
8. **end for**
9. **end for**
10. **end for**
11. **return** $S \in N_{1,n}$

Figure 1: Pseudecode of CYK Algorithm

3 Results

Generated sentences with by rules randomly :

CORRECT –>"*every beautiful sandwich wanted i with me !*"
CORRECT –>"*is it true that i ate you ?*"
CORRECT –>"*you want every pickle from every sandwich .*"
CORRECT –>"*this president pickled that fine fine president .*"

Generated sentences with by words randomly :

INCORRECT –>"*it president to pickled mouse me every prefer .*"
INCORRECT –>"*mouse delicious kissed ate on me washed i !*"
INCORRECT –>"*on this a i to mouse me kissed .*"
INCORRECT –>"*in delicious ate ate is need this to !*"

As can be seen above on the left, the sentence generated from the rules always gave the "CORRECT" result in the CYK algorithm. In order to give "CORRECT" result, the last element of the table in the CYK algorithm must contain 'S'. This shows that the sentences start from ROOT as a rule and continue iterative with 'S'. Maybe, we can understand that it is grammatically correct.

As can be seen above on the right, only sentence randomly generated from words sent the CYK algorithm to "INCORRECT" most of the time. In order to give "INCORRECT" result, there should be no "S" in the last element of the table in the CYK algorithm. This means that only the sentences we randomly produce are not grammatically correct. When we look directly at the sentence, we can understand that it is not grammatically correct.