LING48 Computational Linguistics
Dartmouth College. Spring 2023

Exercises for Week 7
Creating Parsing Trees

Structure of this week's exercises

There are three exercises described below:

(1) Context-Free Grammar rules to parse Bribri sentences
(2) Context-Free Grammar rules to parse Cook Islands Māori sentences
(3) Writing Haiku in English using context-free grammar rules (this is a bit more challenging)

You are free to choose the exercise that interests you the most. You need to do the following:

(1) Doing **one** of the exercises.
(2) Turning in a report in a PDF/LibreOffice/Word file, as suggested in each exercise.
(3) Include screen captures in your PDF document, as explained below.
(4) Submit your code and your PDF with answers/screenshots before 11:59 pm of Thursday, May 25th, 2023.

Exercises 1 and 2: Parsing of languages other than English

CFG rules can model the syntax of any human language. It uses the rules described in section 1.2 of this webpage: https://www.nltk.org/book/ch08.html. The figure below shows you an example. In addition to this, the Canvas website has examples from Klingon and Tagalog syntactic parsing.

```
>>> groucho_grammar = nltk.CFG.fromstring("""
... S -> NP VP
... PP -> P NP
... NP -> Det N | Det N PP | 'I'
... VP -> V NP | VP PP
... Det -> 'an' | 'my'
... N -> 'elephant' | 'pajamas'
... V -> 'shot'
... P -> 'in'
... """)
```

This grammar permits the sentence to be analyzed in two ways, depending on whether the prepositional phrase *in my pajamas* describes the elephant or the shooting event.

```
>>> sent = ['I', 'shot', 'an', 'elephant', 'in', 'my', 'pajamas']
>>> parser = nltk.ChartParser(groucho_grammar)
>>> for tree in parser.parse(sent):
...     print(tree)
...
(S
  (NP I)
  (VP
    (VP (V shot) (NP (Det an) (N elephant)))
    (PP (P in) (NP (Det my) (N pajamas)))))
(S
  (NP I)
  (VP
    (V shot)
    (NP (Det an) (N elephant) (PP (P in) (NP (Det my) (N pajamas))))))
```

For your homework you need to **CHOOSE JUST ONE**[1] of the two languages below: Cook Islands Māori (Polynesian, Cook Islands) or Bribri (Chibchan, Costa Rica). Make a grammar that can provide a single, unambiguous parsing for each of the 5 sentences of the language you choose.

[1] You can do both if you'd like the practice, but you only need to turn one in.

If you've taken LING1: I recommend that you use phrase structure rules like the ones we studied this week. **I RECOMMEND THAT YOU DO NOT USE GOVERNMENT AND BINDING**. You can if you want to, but Python does not permit the use of empty categories (precisely because empty categories are theory-dependent), so you're going to have to get creative with how you provide the input. If you go ahead and use a G&B style of parsing, please use it in a uniform manner (e.g. all trees should have an IP/TP, all XP should have X̄). If you have not taken LING1, please disregard this paragraph.

You need to submit the following:

(1) Your Python code.
(2) A PDF/Word/LibreOffice document where you paste the syntactic trees that you get for each of these 5 sentences. Remember that you need to change the parentheses for square brackets so you can use a web tree maker (e.g. http://mshang.ca/syntree/). You can provide the parsed string if you want to, but you **must** provide the graphical tree from the website.
(3) Screenshots of the output of your program.

Cook Islands Māori (Polynesian, Cook Islands)

Some recommendations:

(a) This language is Verb+Subject+Object, so you can't make a single constituent that only includes the verb+object. Therefore, your VP cannot contain the object.[2]
(b) The TAM words should be grouped with the verb inside of a VP.
(c) The preposition *i* marks the direct object. I suggest you treat this as a special phrase (e.g. `NPOBJ`)

This orthography is the actual orthography used for the language.

(1) Kua tunu a Tere i te taro
TAM.PST plant DET Tere PREP.ACC DET taro
"Tere planted the taro"

(2) Te 'aere nei te va'ine ki te 'are maki
TAM.PROG go TAM.PROG DET woman to DET house sick(noun)
"The woman is going to the hospital"

(3) Kia orāna kōtou kātoatoa
TAM.OPT be.healthy you.all(noun) everyone(noun)
"Hi everyone!"

(4) E reka ana koe i te 'ānani
TAM.HABITUAL like TAM.HABITUAL you(noun) PREP.ACC DET orange
"Do you like oranges?"

(5) Kua 'aere ake koe ki Rarotonga
TAM.PST go DIRECTIONAL(TAM) you(noun) to Rarotonga
"Have you ever been to Rarotonga?"

[2] In case you're in linguistics, VSO languages present interesting challenges to Chomskyan Government and Binding, precisely because the VP and the object NP are decoupled.

Bribri (Chibchan, Costa Rica)

Some recommendations:

(a) Remember to keep the verb and its direct object together in a VP (verb phrase).
(b) You might need to make two kinds of noun phrases: NP and NP-ERG. The last one is for the subjects that have ergative postpositions.
(c) You will also need a CP (complement phrase).

The orthography presented here has two simplifications:

(a) It uses the letter 'h' to indicate nasality. In the actual orthography of Bribri, nasality is indicated with a line underneath the vowel: *Ye' tö ù sú̲* "I saw the house" [dʒeɻ tʊɻ uɬ sũɣ].

(b) It uses the letter *ö^*, with the tone separate from the umlaut. The actual orthography of the language has both symbols combined: *kôtkkôtk* 'ideophone: running fast' [kʊtkɻkʊtkɻ].

(1) Ye' tö ù súh
I(noun) ERG house see-PFV.PROSP
"I saw the house"

(2) Ák tër mèhsa kih
stone rest.on-IPFV table on
"The stone is on the table" (lit: "The stone lies on the table")

(3) Ye' dé bikâkala apánuhk
you(noun) go-PFV.PROSP priest wait-INF
"I came here to wait for the priest"

(4) Ie' dör nihmàh tké cháchá
3.SG(NOUN) ERG fish shoot.with.arrow-PFV.PROSP ADVERB:IDEOPHONE
"He shot an arrow through the fish... boom!"

(5) Nahmùh túhn kö^tkkö^tk
tiger run-IPFV ADVERB:IDEOPHONE
"The tiger ran [with its back arched]... whoosh!"

Exercise 3: Writing Haiku in English

Part 1: Writing a 5-7-5 poem in English

One of the characteristics of haiku poetry is that it has a strict syllabic structure: It should have 3 lines, with 5-7-5 syllables. For example:

Since my house burned down
I now own a better view
of the rising moon.
- Mizuta Masahide

Oh, tranquility!
Penetrating every rock
a cicada's voice.
- Matsuo Bashō

The file `haiku-template.py` has a blank context-free grammar. Please use CFG rules to write haiku (5-7-5 syllable structures) in English. This will need some creativity, because you will have to decide what the rules are going to contain. For example, what words will you have within a phrase of seven syllables? You can include terminals manually, according to your creativity and the kind of poems you want the computer to generate. If you don't want to create your terminals manually, you can create them programmatically. (The CFG is, after all, a string. You can generate it programmatically and then create the CFG from that).

Part 2: Writing an actual haiku

You **do NOT need** to implement this; this is a question: Actual haiku are more than just a 5-7-5 structure. Haiku also have two important components:

(1) A *kigo* (季語), a "seasonal word". A haiku needs a word that tells you what season of the year it is meant to evoke. For example, in the haiku below, the word *moon* is meant to evoke autumn, and the word *cicada* is meant to invoke summer. Here you'll find lists of common *kigo*: https://en.wikipedia.org/wiki/Kigo.

(2) A *kireji* (切れ字), a "cutting word". These are meant to cut the haiku into two parts. They are usually grammatical words in Japanese. In the second haiku, the word *ya* emphasizes the previous phrase and cuts the haiku in two sections. In the first haiku, the word *kaná* "I wonder" is at the end of the poem; it doesn't split the poem in two, but it indicates a clear end to the statement.

蔵焼けて	kura yakete	*Since my house burned down*
障るものなき	sawaru mono naki	*I now own a better view*
月見**哉**	tsukimi **kana**	*of the rising moon.*
閑けさ**や**	kan kesa **ya**	*Oh, tranquility!*
岩にしみいる	iwa ni shimi iru	*Penetrating every rock*
蝉の声	semi no koe	*a cicada's voice.*

How would you implement seasonal words and cutting words in the CFG you made for part 1? Please write an explanation and some basic pseudocode.

Things you need to turn in for exercise 3:

You need to perform four tasks:

(1) Modify the code to produce haikus, as described in part 1.
(2) Make an example screenshot of the haikus that the program is producing.[3]
(3) Create a PDF/Word/LibreOffice document with your favorite three haikus, and with the answer to the question about seasonal words and cutting words.
(4) Submit the Python/Jupyter file, the screenshot(s) and the PDF to Canvas.

Grading for HW7:

CFG of CIM/Bribri:

0.9 * 5	Tree for each of the 5 sentences
0.5	Other instructions (adequate screenshots for each sentence, orderly code, your name in the code).

CFG for Haiku:

3	Correct structure (5-7-5 sylls)
1	Ideas to expand kigo and kireji
0.5	Creativity and writing quality (what is the range of haiku that can be generated? Do they look like real haiku?)
0.5	Other instructions (adequate example screenshots, orderly code, your name in the code and the document)

[3] Be careful when you generate the haiku. As we studied on week 2, the generative capacity of human language is, if not infinite, at least *very* large. You'll quickly see that the possible number of haikus you can generate is growing out of control. I recommend generating no more than 500 million, and printing something like 1 in every million so you can choose your favorites.