Parsing Beyond CFG
Homework 3: TALs and TAG Parsing (CYK)

Laura Kallmeyer, Tatiana Bladier

Sommersemester 2018

Question 1 (Formal properties of TALs)

a) Using pumping lemma for tree-adjoining languages (TALs), show that the language L_1 is not a TAL:

$$L_1 = \{w \mid w \in \{a, b, c, d, e\}^*, w = a^n b^n c^n d^n e^n \text{ and } n \geq 0\}$$

Solution:

If we assume that L_1 is a TAL, L_1 must satisfy the pumping lemma for TALs for some constant p. Let $w = a^{p+1}b^{p+1}c^{p+1}d^{p+1}e^{p+1}$. In this case, according to the weak pumping lemma for TALs, there exist the substrings w_1, w_2, w_3, and w_4 (of which at least one should not be empty) such that these substrings can be iterated repeatedly into w to create new words in L_1. Due to the condition that every word in the language L_1 contains an equal number of five different terminal symbol, one of the substrings w_1, w_2, w_3, or w_4 must contain more than two different symbols. However, if we iterate a substring with two different terminal symbols at least once, we won't be able to produce a word with equal numbers of five different symbols. This means that the obtained word would not belong to L_1. This is a contradiction to the assumption that L_1 belongs to the class of TALs.

b) Show that the following language L_2 is also not a TAL:

$$L_2 = \{w \mid w \in \{a, b, c, d, e\}^*, |w|_a = |w|_b = |w|_c = |w|_d = |w|_e\}$$

Hint: Intersect L_2 with a suitable regular language. You can assume that we have already proved that the language $L_1 = \{w \mid w \in \{a, b, c, d, e\}^*, w = a^n b^n c^n d^n e^n \text{ and } n \geq 0\}$ is not a TAL.

Solution:

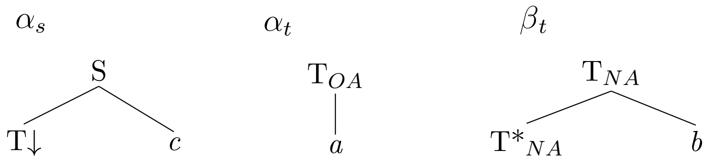
We assume that L_2 is a TAL. Then, since TALs are closed under intersection with regular languages, the language L_4 resulting from the intersection of L_2 with the regular language $L_3 = \{w \mid w \in \{a, b, c, d, e\}^*, w = a^*b^*c^*d^*e^*\}$ should also be a TAL:

$$L_4 = L_2 \cap L_3 = \{w \mid w \in \{a, b, c, d, e\}^*, w = a^n b^n c^n d^n e^n \text{ and } n \geq 0\}$$

L_4 must be a TAL as well. However, since we have already proved that L_4 is not a TAL. Therefore, our assumption that L_2 is a TAL is false.

Question 2 (TAG CYK parsing)

Consider the TAG consisting of the following elementary trees α_s, α_t, and β_t:



Give the trace of the CYK parse (the version from the course slides) of $w = abc$, i.e., a list of all items that get generated. Explain for each item, by which operation it is obtained and from which antecedent items.

Solution:

	Item	Rule
1.	$[\alpha_t, 1_\top, 0, -, -, 1]$	lex-scan (a)
2.	$[\beta_t, 2_\top, 1, -, -, 2]$	lex-scan (b)
3.	$[\alpha_s, 2_\top, 2, -, -, 3]$	lex-scan (c)
4.	$[\beta_t, 1_\top, 0, 0, 0, 0]$	foot-predict
5.	$[\beta_t, 1_\top, 0, 0, 1, 1]$	foot-predict
6.	$[\beta_t, 1_\top, 0, 0, 2, 2]$	foot-predict
7.	$[\beta_t, 1_\top, 0, 0, 3, 3]$	foot-predict
8.	$[\beta_t, 1_\top, 1, 1, 1, 1]$	foot-predict
9.	$[\beta_t, 1_\top, 1, 1, 2, 2]$	foot-predict
10.	$[\beta_t, 1_\top, 1, 1, 3, 3]$	foot-predict
11.	$[\beta_t, 1_\top, 2, 2, 2, 2]$	foot-predict
12.	$[\beta_t, 1_\top, 2, 2, 3, 3]$	foot-predict
13.	$[\beta_t, 1_\top, 3, 3, 3, 3]$	foot-predict
14.	$[\alpha_t, \epsilon_\bot, 0, -, -, 1]$	move-unary from 1.
15.	$[\beta_t, \epsilon_\bot, 0, 0, 1, 2]$	move binary from 2. and 5.
16.	$[\beta_t, \epsilon_\bot, 1, 1, 1, 2]$	move binary from 2. and 8.
17.	$[\beta_t, \epsilon_\top, 0, 0, 1, 2]$	null-adjoin from 15.
18.	$[\beta_t, \epsilon_\top, 1, 1, 1, 2]$	null-adjoin from 16.
19.	$[\alpha_t, \epsilon_\top, 0, -, -, 2]$	adjoin 17. in 14.
20.	$[\alpha_s, 1_\top, 0, -, -, 2]$	substitute 19.
21.	$[\alpha_s, \epsilon_\bot, 0, -, -, 3]$	move-binary from 20. and 3.
22.	$[\alpha_s, \epsilon_\top, 0, -, -, 3]$	null-adjoin from 21. $\rightarrow$ goal item