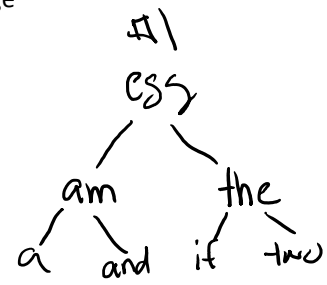
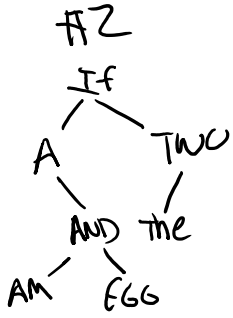
2019-04-22 Optimal BSTs

Monday, April 22, 2019 4:23 PM

- Scenario: we're trying to build a tree that translates words from English to Spanish.
- Question: how should this tree be arranged?
  - Obvious approach: make it well-balanced ((a-z)/2 is root)
    - Good average search time
  - Based on rate of occurrence
    - Prioritizes faster lookups for more common words
  - Which would be faster overall?
- To answer this question conclusively, we would need not only a dictionary of words but also frequency of usage

Word	Probability
A	0.22
AM	0.18
AND	0.20
EGG	0.05
IF	0.25
THE	0.02
TWO	0.08

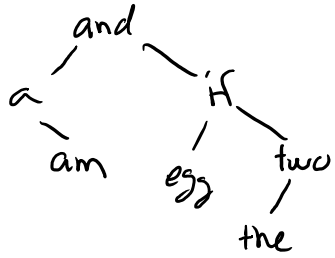




Evaluating the trees

Word	Probability	Balanced Tree # of checks	Balanced Tree Cost	Greedy Tree # edges	Greedy Tree cost
A	0.22	3	0.22*3 = 0.66	2	0.22*2 = 0.44
AM	0.18	2	0.18*2 = 0.36	4	0.18*4 = 0.72
AND	0.20	3	0.20*3 = 0.6	3	0.20*3 = 0.6
EGG	0.05	1	0.05*1 = 0.05	4	0.05*4 = 0.2
IF	0.25	3	0.25*3 = 0.75	1	0.25*1 = 0.25
THE	0.02	2	0.02*2 = 0.04	3	0.02*3 = 0.06
TWO	0.08	3	0.08*3 = 0.24	2	0.08*2 = 0.16
TOTAL	1.00		2.70		2.43

Consider the following tree:



Word	Probability	Magic Tree # of checks	Magic Tree Cost
A	0.22	2	.44

AM	0.18	3	.54
AND	0.20	1	.20
EGG	0.05	3	.15
IF	0.25	2	.50
THE	0.02	4	.08
TWO	0.08	3	.24
TOTAL	1.00		2.15

Brute Force Approach

A	AM	AND	EGG	IF	THE	TWO