Topics in Natural Language Processing

Shay Cohen

Institute for Language, Cognition and Computation

University of Edinburgh

Lecture 7

Administrativia

Reminder: the requirements for the class are presentations, assignment, brief paper responses and an essay.

I sent an email over the week with detailed instructions for all requirements other than the assignment.

Please send me your topic for the essay by this Friday 4pm (reply to the email I sent).

I expect to release the assignment 3-4 weeks before the deadline.

If something remains unclear, please email me.

Linear Score Function

Consider a model which is a PCFG.

$p(r_i)$

Probability of a tree: $t = (r_1 \ldots r_n)$

$$p(t) = \prod_{i=1}^{n} p(r_i) = \prod_{r \in t} p(r)^{freq(r,t)} =$$

$$= \prod_{r \in R} p(r)^{freq(r,t)}$$

"Best" tree y given sentence x:

$$t^*(x) = \underset{\substack{t \\ yield(t)=x}}{\arg\max}\ p(t)$$

sentence

Linear Score Function

'Best" tree given sentence x:

$$y^* = \arg\max_{y:\ \text{yield}(y)=x} \underbrace{\prod_{r \in y} p(r)^{\text{freq}(y,r)}}_{p(t)} = \arg\max_{y,\ \text{yield}(y)=x} \sum \big(\underbrace{\log p(r)}_{w(r)}\big) \times \text{freq}(y,r)$$

$$= \arg\max_{y,\ \text{yield}(y)=x} \sum_{r \in R} w(r) \times \text{freq}(y,r)$$

parameters

"refers" to the structure

$$\arg\max_{y} \underset{w}{\underset{\uparrow}{\theta}}^{T} f(y,x)$$

The CKY Algorithm

$$y^* = \arg \max_{y:\ \text{yield}(y)=x} \sum_{r \in y} \underline{w(r) \times \text{freq}(y, r)}$$

$\alpha(A, i, j)$ - the maximum score for a tree that has head A, and spans $x_i \dots x_j$

$$\alpha(A, i, j) =$$

$$= \max_{k=i}^{j-1} \max_{A \to B\ C} \alpha(B, i, k) + \alpha(C, k+1, j) + w(A \to B$$

Multiplicative version of the CKY algorithm

$$y^* = \arg \max_{y:\ \text{yield}(y)=x} \prod_{r \in y} p(r)^{\text{freq}(y,r)}$$

$\alpha(A, i, j)$

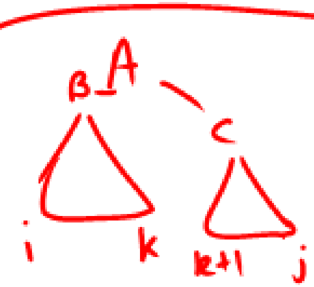
$$\alpha(A, i, j) = \max_{i \le k < j} \max_{A \to B\,C} \underline{\alpha(B, i, k)} \times \alpha(\underline{C, k+1}, j) \times \boxed{p(A \to B\ C \mid A)}$$

The Inside Algorithm

$$p(x) = \sum_{y:\ \text{yield}(y)=x} \prod_{r\in y} p(r)^{\text{freq}(y,r)}$$

$\alpha(A,i,j)$ the sum over all derivations spanning $x_i \ldots x_j$ headed by A.

$$\alpha(A,i,j) = \sum_{k=i}^{j-1} \sum_{A\to B\,C} \alpha(B,i,k) \times \alpha(C,k+1,j) \times p(A\to B\,C$$



$k+1 = j \Rightarrow k = j-$

C

x_j

$$p(\text{tree},x) = \prod_{\text{rule}} p(\text{rule})$$

$$p(x) = \sum_{\text{tree}} p(\text{tree},x)$$ marginalizing the tree

Inside and CKY

What is the connection between the inside algorithm and CKY?

CKY:

Version 1:

$$\alpha(A,i,j) = \max_{i \le k \le j-1} \max_{A \to B\,C} p(A \to B\,C|A)\alpha(B,i,k)\alpha(C,k+1,j)$$

Version 2:

$$\alpha(A,i,j) = \max_{i \le k \le j-1} \max_{A \to B\,C} w(A \to B\,C) + \alpha(B,i,k) + \alpha(C,k+1,j)$$

Inside:

$$\alpha(A,i,j) = \sum_{k=i}^{j-1} \sum_{A \to B\,C} p(A \to B\,C|A)\alpha(B,i,k)\alpha(C,k+1,j)$$

Semirings

What is a semiring?

An algebraic structure over $\mathbb{R}$

$\otimes$ $\quad a \oplus b$

$\oplus$ $\quad a \otimes b$

Semirings

What is a semiring?

- A set R $\mathbb{R}$ $[0,1]$
- Two operations: $\oplus$ and $\otimes$
- Identity element $\bar{1}$ for $\otimes$ $\bar{1} \otimes a = a$
- Identity element $\bar{0}$ for $\oplus$ $\bar{0} \oplus a = a$
- (... and a few more important properties)

CKY and Semirings

CKY:

$$\alpha(A,i,j) = \max_{i \le k \le j-1} \max_{A \to B\,C} p(A \to B\,C|A)\alpha(B,i,k)\alpha(C,k+1,j)$$

What is the semiring?

$\oplus$ $a \oplus b = \max\{a, b\}$

$\otimes$ $\times$

$\bar{1}$ 1

$\bar{0}$ 0

CKY and Semirings

CKY:

$$\alpha(A, i, j) = \max_{i \le k \le j-1} \max_{A \rightarrow B\,C} w(A \rightarrow B\,C) + \alpha(B, i, k) + \alpha(C, k+1, j)$$

What is the semiring?

$\oplus$ $a \oplus b = \max\{a, b\}$

$\otimes$ $a \otimes b = a + b$

$\bar{1}$

$\bar{0}$

Inside and Semirings

Inside:

$$\alpha(A,i,j) = \sum_{k=i}^{j-1} \sum_{A \to B\,C} p(A \to B\,C|A)\alpha(B,i,k)\alpha(C,k+1,j)$$

What is the semiring?

$\oplus$ $\quad a \oplus b = a + b$

$\otimes$ $\quad a \otimes b = a \times b$

$\bar{1}$ $\quad 1$

$\bar{0}$ $\quad 0$

Parsing as Weighted Logic Programming

$$\overset{\alpha}{\text{constit}}(a, i, j)\underline{\oplus} = \overset{\alpha}{\text{constit}}(b, i, k) \otimes \overset{\alpha}{\text{constit}}(c, k+1, j) \otimes \text{rule}(a \to b\, c)^{\overset{p(r)}{w}}$$

$$\text{constit}(a, i, i) \oplus = \text{rule}(a \to w)$$

(bottom tree conditions)

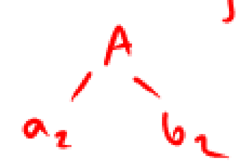
Goal: $\text{constit}(S, 0, n)$

$R = \mathbb{R} \times \text{TREES}$

$a + b = (\max\{a_1, b_1\},$ the corresponding combination of those two subtrees)

" (a_1, a_2)

number tree



Parsing with Tree Adjoining Grammars

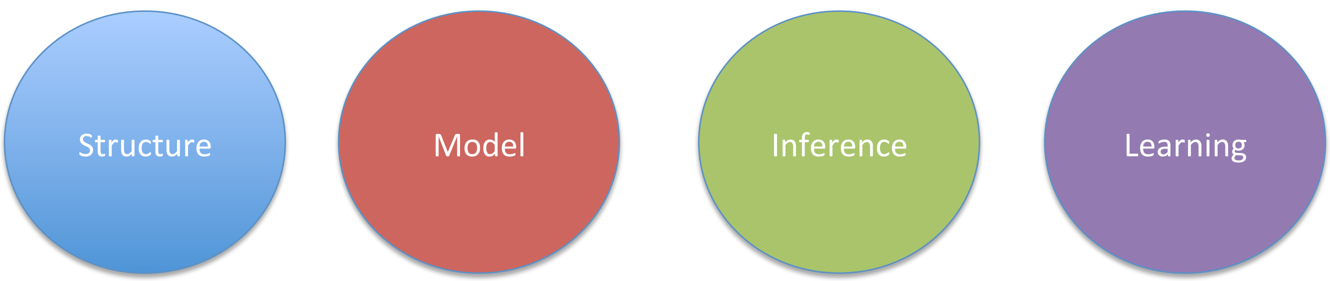
TAGs have a dynamic programming which resembles CKY

The items on the chart are $\mathrm{constit}(A, b, i, j, f_1, f_2)$:

- A is the head nonterminal
- b tells whether adjunction already happened for the foot node (Boolean)
- (i, j) is the endpoints that A spans
- (f_1, f_2) are the endpoints that the foot node spans (could be empty)

Solving an NLP problem

Reminder: the components of an NLP problem



Estimation until now

- Count and normalise
- Corresponds to maximum likelihood estimate for multinomial models