Incremental Parsing in Bounded Memory

William Schuler
Department of Linguistics
The Ohio State University
schuler@ling.osu.edu

Abstract

This tutorial will describe the use of a factored probabilistic sequence model for parsing speech and text using a bounded store of three to four incomplete constituents over time, in line with recent estimates of human short-term working memory capacity. This formulation uses a grammar transform to minimize memory usage during parsing. Incremental operations on incomplete constituents in this transformed representation then define an extended domain of locality similar to those defined in mildly context-sensitive grammar formalisms, which can similarly be used to process long-distance and crossed-and-nested dependencies.

1 Introduction

This paper will describe the derivation of a factored probabilistic sequence model from a probabilistic context-free grammar (PCFG). The resulting sequence model can incrementally parse input sentences approximately as accurately as a bottom-up CKY-styple parser, incrementally estimating the contents of a bounded memory store of intended constituents, consisting of only three to four working memory elements, in line with recent estimates of human short-term working memory capacity (Cowan, 2001). The detailed derivation of this model is intended to illustrate how probabilistic dependencies from an original PCFG (or other types of syntax-derived dependencies) can be preserved in a processing model with human-like working memory constraints.

1.1 Notation

This paper will associate variables for syntactic categories c, trees or subtrees τ, and string yields $\bar{x}$ with constituents in phrase structure trees, identified using subscripts that describe the path from the root of the tree containing this constituent to the constituent itself. These paths may consist of left branches (indicated by '0's in the path) and right branches (indicated by '1's), concatenated into sequences η (or ι or κ). Thus, if a path η identifies a constituent, that constituent's left child would be identified by $\eta 0$, and that constituent's right child would be identified by $\eta 1$. The empty path ϵ will be used to identify the root of a tree.

The probabilistic parsers defined here will also use an indicator function $[\![\cdot]\!]$ to denote deterministic probabilities: $[\![\phi]\!] = 1$ if ϕ is true, 0 otherwise.

2 Parsing

For a phrase structure subtree rooted at a constituent of category c_η with yield $\bar{x}_\eta$ (a sequence of individual words x), the task of parsing will require the calculation of the inside likelihood probability or Viterbi (best subtree) probabilities. When the domain X of words x is a subset of the domain C of category labels c, these can be calculated using rule probabilities in a probabilistic context-free grammar (PCFG) model θ_{G}, notated:

$$\mathsf{P}_{\theta_{\mathrm{G}}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) = \mathsf{P}_{\theta_{\mathrm{G}}}(c_{\eta 0}\ c_{\eta 1} \mid c_\eta) \qquad (1)$$

Any yield $\bar{x}_\eta$ can be decomposed into prefix $\bar{x}_{\eta 0}$ and suffix $\bar{x}_{\eta 1}$ yields:

$$\bar{x}_\eta = \bar{x}_{\eta 0}\bar{x}_{\eta 1} \qquad (2)$$

Therefore, inside likelihood probabilities can be defined by marginalizing over all such decompositions:

$$\mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_\eta \,|\, c_\eta) = \begin{cases} \text{if } |\bar{x}_\eta| = 1 : [\![\bar{x}_\eta = c_\eta]\!] \\ \text{otherwise}: \sum\limits_{\bar{x}_{\eta 0} c_{\eta 0}, \bar{x}_{\eta 1} c_{\eta 1}} \mathsf{P}_{\theta_{\text{G}}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) \\ \qquad \cdot \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_{\eta 0} \,|\, c_{\eta 0}) \\ \qquad \cdot \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_{\eta 1} \,|\, c_{\eta 1}) \end{cases} \quad (3)$$

and Viterbi scores (the probability of the best tree) can be defined by maximizing over all such decompositions:

$$\mathsf{P}_{\theta_{\text{Vit(G)}}}(\bar{x}_\eta \,|\, c_\eta) = \begin{cases} \text{if } |\bar{x}_\eta| = 1 : [\![\bar{x}_\eta = c_\eta]\!] \\ \text{otherwise}: \max\limits_{\bar{x}_{\eta 0} c_{\eta 0}, \bar{x}_{\eta 1} c_{\eta 1}} \mathsf{P}_{\theta_{\text{G}}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) \\ \qquad \cdot \mathsf{P}_{\theta_{\text{Vit(G)}}}(\bar{x}_{\eta 0} \,|\, c_{\eta 0}) \\ \qquad \cdot \mathsf{P}_{\theta_{\text{Vit(G)}}}(\bar{x}_{\eta 1} \,|\, c_{\eta 1}) \end{cases} \quad (4)$$

From these, it is possible to obtain the probability of a sentence $\bar{x}_\epsilon$:

$$\mathsf{P}(\bar{x}_\epsilon) = \sum_{c_\epsilon} \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_\epsilon \,|\, c_\epsilon) \cdot \mathsf{P}(c_\epsilon) \quad (5)$$

and the most likely tree:

$$\hat{\tau}_\epsilon = \operatorname*{argmax}_{\tau_\epsilon} \mathsf{P}_{\theta_{\text{Vit(G)}}}(\bar{x}_\epsilon \,|\, c_{\tau_\epsilon}) \cdot \mathsf{P}(c_{\tau_\epsilon}) \quad (6)$$

3 Incremental Parsing using Incomplete Constituents

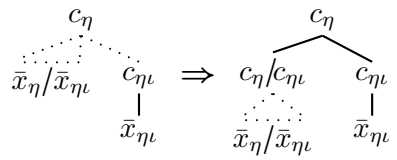
Note that any prefix of a yield $\bar{x}_\eta$ can be rewritten as an 'incomplete yield' consisting of the complete yield lacking some suffix of that yield $\bar{x}_{\eta\iota}$:

$$\bar{x}_\eta = (\bar{x}_\eta / \bar{x}_{\eta\iota}) \bar{x}_{\eta\iota} \quad (7)$$

It is therefore possible to decompose any inside likelihood or Viterbi probability into two parts: first, an 'incomplete constituent' probability of generating this incomplete yield $\bar{x}_\eta / \bar{x}_{\eta\iota}$ along with an *awaited* constituent of category $c_{\eta\iota}$, given an *active* constituent of category c_η; and second, an ordinary inside or Viterbi probability (or 'complete constituent' probability) of generating $\bar{x}_{\eta\iota}$ given $c_{\eta\iota}$:

$$\mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_\eta \,|\, c_\eta) = \sum_{\iota \in 1^+, \bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \cdot \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_{\eta\iota} \,|\, c_{\eta\iota}) \quad (8)$$

This decomposition can be represented graphically as a transformation of the structure of a recurrence from the standard PCFG recurrence above, corresponding to PCFG dependencies, to one involving incomplete constituents (in particular, this will transform the end of a right-expanding sequence into the beginning of a left-expanding sequence of incomplete constituents):

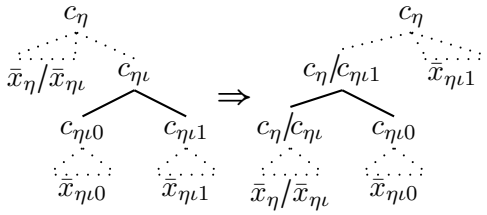


(9)

This decomposition gives incomplete constituent probabilities that can then be decomposed into other incomplete constituent probabilities (this will transform the middle of a right-expending sequence into the middle of a left-expanding sequence of incomplete constituents):

$$\begin{aligned} &\mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_\eta / \bar{x}_{\eta\iota 1},\ c_{\eta\iota 1} \,|\, c_\eta) \\ &\quad = \sum_{\bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \\ &\qquad\qquad \cdot \mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_{\eta\iota} / \bar{x}_{\eta\iota 1},\ c_{\eta\iota 1} \,|\, c_{\eta\iota}) \quad (10) \\ &\quad = \sum_{\bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \\ &\qquad\qquad \cdot \sum_{\bar{x}_{\eta\iota 0}, c_{\eta\iota 0}} \mathsf{P}_{\theta_{\text{G}}}(c_{\eta\iota} \rightarrow c_{\eta\iota 0}\ c_{\eta\iota 1}) \\ &\qquad\qquad\qquad \cdot \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_{\eta\iota 0} \,|\, c_{\eta\iota 0}) \quad (11) \end{aligned}$$

represented graphically:

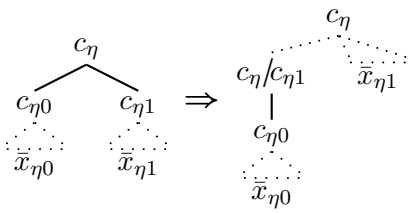


(12)

or into the product of a grammar rule probability and an inside probability at the end of a sequence of such decompositions (this will transform the beginning of a right-expanding sequence into the end of a left-expanding sequence of incomplete constituents):

$$\begin{aligned} &\mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_\eta / \bar{x}_{\eta 1},\ c_{\eta 1} \,|\, c_\eta) = \\ &\quad \sum_{\bar{x}_{\eta 0}, c_{\eta 0}} \mathsf{P}_{\theta_{\text{G}}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) \cdot \mathsf{P}_{\theta_{\text{Ins(G)}}}(\bar{x}_{\eta 0} \,|\, c_{\eta 0}) \quad (13) \end{aligned}$$

represented graphically:



(14)

Essentially this transformation turns right-expanding sequences of constituents (with subscript addresses ending in '1') into left-expanding sequences of incomplete constituents, which can be composed together as they are recognized incrementally from left to right. The decompositions of Equations 8–13, taken together, are a model-based right-corner transform (Schuler, 2009).

This transformation can then be defined on depth-specific rules, allowing the sequence model to keep track of a bounded amount of center-embedded phrase structure.

$$\mathsf{P}_{\theta_{\text{Ins(G),d}}}(\bar{x}_\eta \,|\, c_\eta) = \sum_{\iota \in 1^+, \bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \cdot \mathsf{P}_{\theta_{\text{Ins(G),d}}}(\bar{x}_{\eta\iota} \,|\, c_{\eta\iota}) \quad (15)$$

$$\begin{aligned} &\mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_\eta / \bar{x}_{\eta\iota 1},\ c_{\eta\iota 1} \,|\, c_\eta) \\ &= \sum_{\bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \cdot \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_{\eta\iota} / \bar{x}_{\eta\iota 1},\ c_{\eta\iota 1} \,|\, c_{\eta\iota}) \quad (16) \\ &= \sum_{\bar{x}_{\eta\iota}, c_{\eta\iota}} \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_\eta / \bar{x}_{\eta\iota},\ c_{\eta\iota} \,|\, c_\eta) \cdot \sum_{\bar{x}_{\eta\iota 0}, c_{\eta\iota 0}} \mathsf{P}_{\theta_{\text{G-R},d}}(c_{\eta\iota} \rightarrow c_{\eta\iota 0}\ c_{\eta\iota 1}) \cdot \mathsf{P}_{\theta_{\text{Ins(G),d+1}}}(\bar{x}_{\eta\iota 0} \,|\, c_{\eta\iota 0}) \quad (17) \end{aligned}$$

$$\mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_\eta / \bar{x}_{\eta 1},\ c_{\eta 1} \,|\, c_\eta) = \sum_{\bar{x}_{\eta 0}, c_{\eta 0}} \mathsf{P}_{\theta_{\text{G-L},d}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) \cdot \mathsf{P}_{\theta_{\text{Ins(G),d}}}(\bar{x}_{\eta 0} \,|\, c_{\eta 0}) \quad (18)$$

Estimating probabilities for the beginning and ending of these transformed left-expanding sequences will require the estimation of expected counts for repeated recursive decompositions of yet-unrecognized incomplete constituents according to Equation 8, marginalizing over values of $\bar{x}$. Since left children and right children are decomposed differently, the expected counts $\theta_{\text{G-RL}^*,d}$ will use PCFG probabilities $\theta_{\text{G-R},d}$ and $\theta_{\text{G-L},d}$ that are conditioned on whether the expanding constituent is a right or left child, and on the center-embedding depth d of the expanding constituent. The expected counts $\theta_{\text{G-RL}^*,d}$ are of constituent categories $c_{\eta\iota}$ anywhere in the left progeny of a right child of category c_η:

$$\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{0}{\rightarrow} c_{\eta 0}\ \ldots) = \sum_{c_{\eta 1}} \mathsf{P}_{\theta_{\text{G-R},d}}(c_\eta \rightarrow c_{\eta 0}\ c_{\eta 1}) \quad (19)$$

$$\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{k}{\rightarrow} c_{\eta 0^k 0}\ \ldots) = \sum_{c_{\eta 0^k}} \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{k-1}{\rightarrow} c_{\eta 0^k}\ \ldots) \cdot \sum_{c_{\eta 0^k 1}} \mathsf{P}_{\theta_{\text{G-L},d}}(c_{\eta 0^k} \rightarrow c_{\eta 0^k 0}\ c_{\eta 0^k 1}) \quad (20)$$

$$\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{*}{\rightarrow} c_{\eta\iota}\ \ldots) = \sum_{k=0}^{\infty} \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{k}{\rightarrow} c_{\eta\iota}\ \ldots) \quad (21)$$

$$\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{+}{\rightarrow} c_{\eta\iota}\ \ldots) = \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{*}{\rightarrow} c_{\eta\iota}\ \ldots) - \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{0}{\rightarrow} c_{\eta\iota}\ \ldots) \quad (22)$$

In practice the infinite sum is estimated to some constant K using value iteration (Bellman, 1957).

These expected counts can then be used to calculate left progeny probabilities:

$$\mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{*}{\rightarrow} c_{\eta\iota}\ \ldots) = \frac{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{*}{\rightarrow} c_{\eta\iota}\ \ldots)}{\sum_{c_{\eta\iota}} \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \overset{*}{\rightarrow} c_{\eta\iota}\ \ldots)} \quad (23)$$

which can be used to calculate forward or Viterbi probabilities for all incomplete constituents in the memory store, along with their yields:

$$\mathsf{P}_{\theta_{\text{Fwd}}}((\bar{x}_{\eta_d} / \bar{x}_{\eta_d \iota_d 1})_{d=1}^{D}, (c_{\eta_d})_{d=1}^{D}, (c_{\eta_d \iota_d 1})_{d=1}^{D}) = \prod_{d=1}^{D} \mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d}\ \ldots) \cdot \mathsf{P}_{\theta_{\text{IC(G)}}}(\bar{x}_{\eta_d} / \bar{x}_{\eta_d \iota_d 1}, c_{\eta_d \iota_d 1} \,|\, c_{\eta_d}) \quad (24)$$

Putting all the pieces together, probabilities for stores of incomplete constituents can now be defined in terms of transitions from possible previous stores of incomplete constituents (set apart by parentheses in the equation below) and reductions of incomplete constituents and terminal symbols into complete constituents (set apart by square brackets). These transitions either:

- add a layer of structure below an awaited constituent at some depth level (the first case below),
- add a layer of structure to the top of an active constituent at some depth level (the second case below), or
- carry forward the probability of an incomplete constituent at a depth at which no transition takes place (the third case below):

$$\mathsf{P}_{\theta_{\text{Fwd}}}((\bar{x}_{\eta_d}/\bar{x}_{\eta_d\iota_d 1})_{d=1}^{D}, (c_{\eta_d})_{d=1}^{D}, (c_{\eta_d\iota_d 1})_{d=1}^{D}) =$$
$$\prod_{d=1}^{D} \begin{cases} \text{if } c_{\eta_{d+1}} = \bar{x}_{\eta_{d+1}}, \iota_d \neq \epsilon: \\ \quad \sum_{\bar{x}_{\eta_d\iota_d}, c_{\eta_d\iota_d}} \\ \quad \cdot \left(\begin{array}{l} \mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d} \ldots) \\ \cdot \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_{\eta_d}/\bar{x}_{\eta_d\iota_d},\ c_{\eta_d\iota_d} \,|\, c_{\eta_d}) \end{array} \right) \\ \quad \cdot \sum_{\bar{x}_{\eta_d\iota_d 0}, c_{\eta_d\iota_d 0}} \sum_{\kappa \ s.t.\ \bar{x}_{\eta_d\iota_d 0\kappa} = c_{\eta_d\iota_d 0\kappa}} \\ \quad \left(\begin{array}{l} \mathsf{P}_{\theta_{\text{G-RL}^*,d+1}}(c_{\eta_d\iota_d} \overset{*}{\rightarrow} c_{\eta_d\iota_d 0} \ldots) \\ \cdot \mathsf{P}_{\theta_{\text{IC(G),d+1}}}(\bar{x}_{\eta_d\iota_d 0}/\bar{x}_{\eta_d\iota_d 0\kappa},\ c_{\eta_d\iota_d 0\kappa} \,|\, c_{\eta_d\iota_d 0}) \end{array} \right) \\ \quad \cdot \left[\begin{array}{l} \frac{\mathsf{P}_{\theta_{\text{G-RL}^*,d+1}}(c_{\eta_d\iota_d} \overset{0}{\rightarrow} c_{\eta_d\iota_d 0} \ldots)}{\mathsf{P}_{\theta_{\text{G-RL}^*,d+1}}(c_{\eta_d\iota_d} \overset{*}{\rightarrow} c_{\eta_d\iota_d 0} \ldots)} \\ \cdot [\![\bar{x}_{\eta_d\iota_d 0\kappa} = c_{\eta_d\iota_d 0\kappa}]\!] \end{array} \right] \\ \quad \frac{\mathsf{P}_{\theta_{\text{G-R},d}}(c_{\eta_d\iota_d} \rightarrow c_{\eta_d\iota_d 0}\ c_{\eta_d\iota_d 1})}{\mathsf{E}_{\theta_{\text{G-RL}^*,d+1}}(c_{\eta_d\iota_d} \overset{0}{\rightarrow} c_{\eta_d\iota_d 0} \ldots)} \\ \text{if } c_{\eta_{d+1}} = \bar{x}_{\eta_{d+1}}, \iota_d = \epsilon: \\ \quad \sum_{\bar{x}_{\eta_d 0}, c_{\eta_d 0}} \sum_{\kappa \ s.t.\ \bar{x}_{\eta_d 0\kappa} = c_{\eta_d 0\kappa}} \\ \quad \left(\begin{array}{l} \mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d 0} \ldots) \\ \cdot \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_{\eta_d 0}/\bar{x}_{\eta_d 0\kappa},\ c_{\eta_d 0\kappa} \,|\, c_{\eta_d 0}) \end{array} \right) \\ \quad \cdot \left[\begin{array}{l} \frac{\mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{+}{\rightarrow} c_{\eta_d 0} \ldots)}{\mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d 0} \ldots)} \\ \cdot [\![\bar{x}_{\eta_d 0\kappa} = c_{\eta_d 0\kappa}]\!] \end{array} \right] \\ \quad \cdot \frac{\mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d} \ldots)}{\mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{+}{\rightarrow} c_{\eta_d 0} \ldots)} \\ \quad \cdot \mathsf{P}_{\theta_{\text{G-L},d}}(c_{\eta_d} \rightarrow c_{\eta_d 0}\ c_{\eta_d 1}) \\ \text{if } c_{\eta_{d+1}} \neq \bar{x}_{\eta_{d+1}}: \\ \quad \left(\begin{array}{l} \mathsf{P}_{\theta_{\text{G-RL}^*,d}}(c_{\eta_{d-1}\iota_{d-1}1} \overset{*}{\rightarrow} c_{\eta_d} \ldots) \\ \cdot \mathsf{P}_{\theta_{\text{IC(G),d}}}(\bar{x}_{\eta_d}/\bar{x}_{\eta_d\iota_d},\ c_{\eta_d\iota_d} \,|\, c_{\eta_d}) \end{array} \right) \end{cases} \tag{25}$$

Note that the left progeny probabilities above cancel out over time, leaving only the relevant original PCFG probabilities at the end of each sentence.

4 Parsing in a Factored Sequence Model

Store elements can now be abstracted away from (i.e. marginalized over) individual constituent structure addresses. Store elements are therefore defined to contain the only the active and awaited ($c^A_{s_t^d}$ and $c^W_{s_t^d}$) constituent categories necessary to compute an incomplete constituent probability:

$$s_t^d \overset{\text{def}}{=} \langle c^A_{s_t^d}, c^W_{s_t^d} \rangle \tag{26}$$
$$\overset{\text{def}}{=} \langle c_{\eta_d}, c_{\eta_d\iota_d} \rangle \ s.t.\ \iota_d \in 1^+, \eta_d \in \eta_{d-1}\iota_{d-1}0^+ \tag{27}$$

Reduction states are defined to contain only the complete constituent category $c_{r_t^d}$ necessary to compute an inside likelihood probability, as well as a flag $f_{r_t^d}$ indicating whether a reduction has taken place (to end a sequence of incomplete constituents):

$$r_t^d \overset{\text{def}}{=} \langle c_{r_t^d}, f_{r_t^d} \rangle \tag{28}$$
$$\overset{\text{def}}{=} \langle c_{\eta_d}, [\![\bar{x}_{\eta_d} = c_{\eta_d}]\!] \rangle \tag{29}$$

Since $\iota_d \in 1^+$, it follows that:

$$x_{1..t} = (\bar{x}_{\eta_d}/\bar{x}_{\eta_d\iota_d})_{d=1}^{D} \tag{30}$$

This allows store elements to be abstracted away from (marginalized over) tree addresses in the calculation of forward or Viterbi probabilities:

$$\mathsf{P}_{\theta_{\text{Fwd}}}(x_{1..t}, s_t^{1..D}) = \sum_{\eta_{1..D}, \iota_{1..D}} \mathsf{P}_{\theta_{\text{Fwd}}}((\bar{x}_{\eta_d}/\bar{x}_{\eta_d\iota_d 1})_{d=1}^{D}, (c_{\eta_d})_{d=1}^{D}, (c_{\eta_d\iota_d 1})_{d=1}^{D}) \tag{31}$$

Forward probabilities can then be factored into contributions from previous store states (θ_{Fwd} at $s_{t-1}^{1..D}$ below, parenthesized in Equation 25), reductions of terminal symbols (θ_{B} below, bracketed in Equation 25), and transition operations (θ_{A} below, not set apart in Equation 25):

$$\mathsf{P}_{\theta_{\text{Fwd}}}(x_{1..t}, s_t^{1..D}) = \sum_{s_{t-1}^{1..D}} \mathsf{P}_{\theta_{\text{Fwd}}}(x_{1..t-1}, s_{t-1}^{1..D}) \cdot \mathsf{P}_{\theta_{\text{A}}}(s_t^{1..D} \,|\, s_{t-1}^{1..D}) \cdot \mathsf{P}_{\theta_{\text{B}}}(x_t \,|\, s_t^{1..D}) \tag{32}$$

Probabilities for recognized sequences of incomplete constituents (Equations 8 through 13), and expected left progeny counts for unrecognized sequences of incomplete constituents (Equations 19

through 22) can be combined in a probabilistic pushdown automaton with a bounded pushdown store (to simulate the bounded working memory store of a human comprehender). This is essentially an extension of a Hierarchical Hidden Markov Model (HHMM) (Murphy and Paskin, 2001), which obtains a most likely sequence of hidden store states $\hat{s}_{1..T}^{1..D}$ of some length T and some maximum depth D, given a sequence of observations (e.g. words) $x_{1..T}$:

$$\hat{s}_{1..T}^{1..D} \stackrel{\text{def}}{=} \underset{s_{1..T}^{1..D}}{\operatorname{argmax}} \prod_{t=1}^{T} \mathsf{P}_{\theta_{\mathrm{A}}}(s_t^{1..D} \mid s_{t-1}^{1..D}) \cdot \mathsf{P}_{\theta_{\mathrm{B}}}(x_t \mid s_t^{1..D}) \tag{33}$$

The model generates each successive store only after considering whether each nested sequence of incomplete constituents has completed and reduced:

$$\mathsf{P}_{\theta_{\mathrm{A}}}(s_t^{1..D} \mid s_{t-1}^{1..D}) \stackrel{\text{def}}{=} \sum_{r_t^1..r_t^D} \prod_{d=1}^{D} \mathsf{P}_{\theta_{\mathrm{R}}}(r_t^d \mid r_t^{d+1} s_{t-1}^d s_{t-1}^{d-1}) \cdot \mathsf{P}_{\theta_{\mathrm{S}}}(s_t^d \mid r_t^{d+1} r_t^d \, s_{t-1}^d s_t^{d-1}) \tag{34}$$

The model probabilities for these store elements and reduction states can then be defined (from Murphy and Paskin 2001) to expand a new incomplete constituent after a reduction has taken place ($f_{r_t^d} = 1$), transition along a sequence of store elements if no reduction has taken place ($f_{r_t^d} = 0$):

$$\mathsf{P}_{\theta_{\mathrm{S}}}(s_t^d \mid r_t^{d+1} r_t^d \, s_{t-1}^d s_t^{d-1}) \stackrel{\text{def}}{=} \begin{cases} \text{if } f_{r_t^{d+1}} = 1,\ f_{r_t^d} = 1 : \mathsf{P}_{\theta_{\text{S-E},d}}(s_t^d \mid s_t^{d-1}) \\ \text{if } f_{r_t^{d+1}} = 1,\ f_{r_t^d} = 0 : \mathsf{P}_{\theta_{\text{S-T},d}}(s_t^d \mid r_t^{d+1} r_t^d \, s_{t-1}^d s_t^{d-1}) \\ \text{if } f_{r_t^{d+1}} = 0,\ f_{r_t^d} = 0 : [\![s_t^d = s_{t-1}^d]\!] \end{cases} \tag{35}$$

and possibly reduce a store element (terminate a sequence) if the store state below it has reduced ($f_{r_t^{d+1}} = 1$):

$$\mathsf{P}_{\theta_{\mathrm{R}}}(r_t^d \mid r_t^{d+1} s_{t-1}^d s_{t-1}^{d-1}) \stackrel{\text{def}}{=} \begin{cases} \text{if } f_{r_t^{d+1}} = 0 : [\![r_t^d = \mathbf{r}_\perp]\!] \\ \text{if } f_{r_t^{d+1}} = 1 : \mathsf{P}_{\theta_{\text{R-R},d}}(r_t^d \mid r_t^{d+1} \, s_{t-1}^d \, s_{t-1}^{d-1}) \end{cases} \tag{36}$$

where $s_t^0 = \mathbf{s}_\top$ and $r_t^{D+1} = \mathbf{r}_\top$ for constants $\mathbf{s}_\top$ (an incomplete root constituent), $\mathbf{r}_\top$ (a complete lexical constituent) and $\mathbf{r}_\perp$ (a null state resulting from the failure of an incomplete constituent to complete). The model is shown graphically in Figure 1.

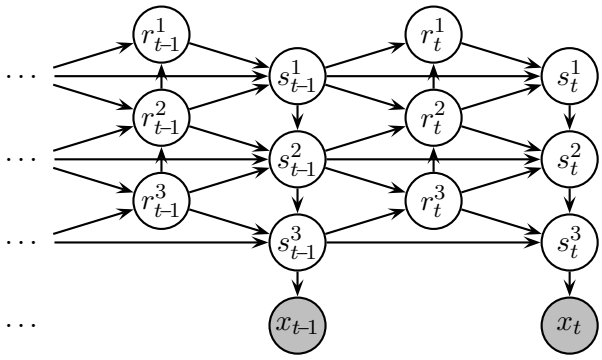


Figure 1: Graphical representation of the dependency structure in a standard Hierarchic Hidden Markov Model with $D = 3$ hidden levels that can be used to parse syntax. Circles denote random variables, and edges denote conditional dependencies. Shaded circles denote variables with observed values.

These pushdown automaton operations are then refined for right-corner parsing (from Schuler 2009), distinguishing active transitions $\theta_{\text{S-T-A},d}$ (in which an incomplete constituent is completed, but not reduced, and then immediately expanded to a new incomplete constituent in the same store element) from awaited transitions $\theta_{\text{S-T-W},d}$ (which involve no completion):

$$\mathsf{P}_{\theta_{\text{S-T},d}}(s_t^d \mid r_t^{d+1} r_t^d \, s_{t-1}^d s_t^{d-1}) \stackrel{\text{def}}{=} \begin{cases} \text{if } r_t^d \neq \mathbf{r}_\perp : \mathsf{P}_{\theta_{\text{S-T-A},d}}(s_t^d \mid s_t^{d-1} r_t^d) \\ \text{if } r_t^d = \mathbf{r}_\perp : \mathsf{P}_{\theta_{\text{S-T-W},d}}(s_t^d \mid s_{t-1}^d r_t^{d+1}) \end{cases} \tag{37}$$

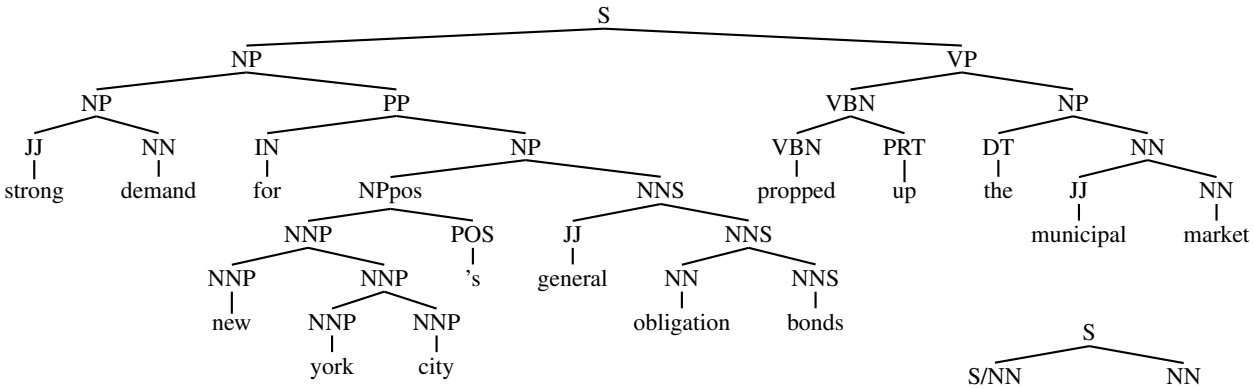
$$\mathsf{P}_{\theta_{\text{R-R},d}}(r_t^d \mid r_t^{d+1} s_{t-1}^d s_{t-1}^{d-1}) \stackrel{\text{def}}{=} \begin{cases} \text{if } c_{r_t^{d+1}} \neq x_t : [\![r_t^d = \mathbf{r}_\perp]\!] \\ \text{if } c_{r_t^{d+1}} = x_t : \mathsf{P}_{\theta_{\text{R-R},d}}(r_t^d \mid s_{t-1}^d s_{t-1}^{d-1}) \end{cases} \tag{38}$$

These right-corner parsing operations then construct a full inside probability decompositions, using Equations 8 through 13 and Equations 19 through 22, marginalizing out all constituents that are not required in each term:

- for expansions:

$$\mathsf{P}_{\theta_{\text{S-E},d}}(\langle c_{\eta\iota}, c'_{\eta\iota} \rangle \mid \langle -, c_\eta \rangle) \stackrel{\text{def}}{=} \mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{*}{\rightarrow} c_{\eta\iota} \ldots) \cdot [\![x_{\eta\iota} = c'_{\eta\iota} = c_{\eta\iota}]\!] \tag{39}$$

a) binarized phrase structure tree:



b) result of right-corner transform:

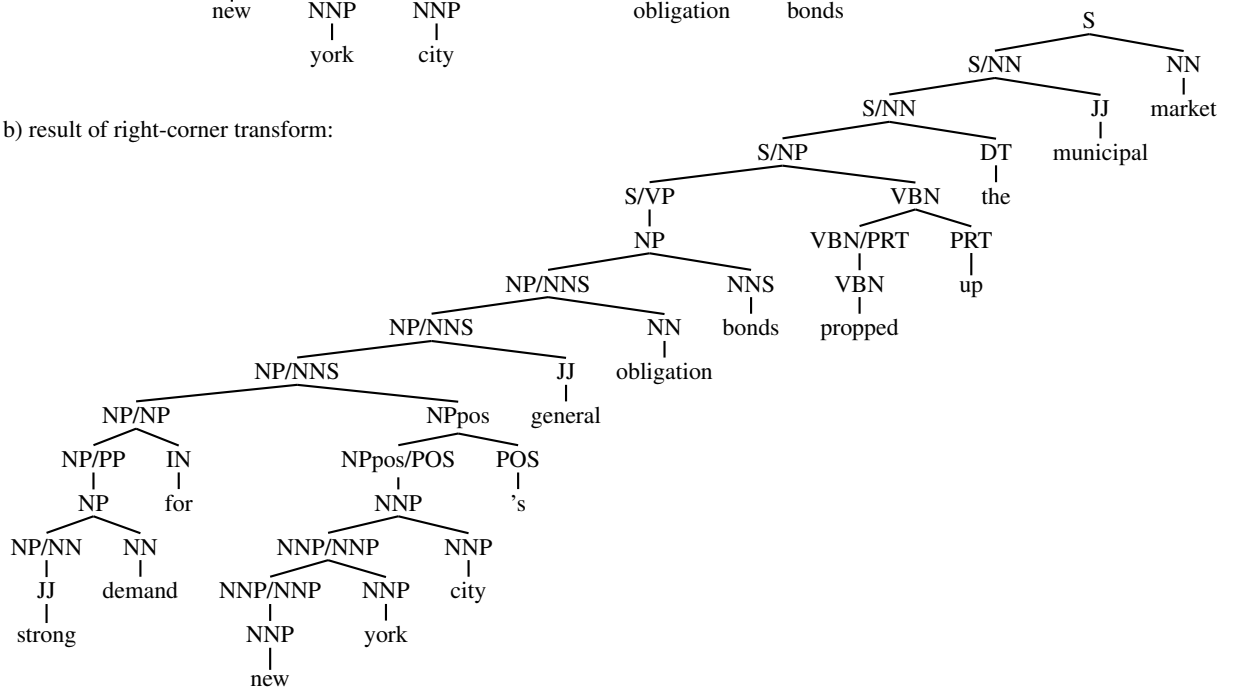


Figure 2: (a) Sample binarized phrase structure tree for the sentence *Strong demand for New York City's general obligations bonds propped up the municipal market,* and (b) a right-corner transform of this binarized tree.

- for awaited transitions, from Equation 11:

$$\mathsf{P}_{\theta_{\text{S-T-W},d}}(\langle c_\eta, c_{\eta\iota 1}\rangle \,|\, \langle c'_\eta, c_{\eta\iota}\rangle\ c_{\eta\iota 0}) \stackrel{\text{def}}{=} [\![c_\eta = c'_\eta]\!] \cdot \frac{\mathsf{P}_{\theta_{\text{G-R},d}}(c_{\eta\iota} \rightarrow c_{\eta\iota 0}\ c_{\eta\iota 1})}{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_{\eta\iota} \stackrel{0}{\rightarrow} c_{\eta\iota 0}\ \ldots)} \quad (40)$$

- for active transitions, from Equations 8 and 13:

$$\mathsf{P}_{\theta_{\text{S-T-A},d}}(\langle c_{\eta\iota}, c_{\eta\iota 1}\rangle \,|\, \langle -, c_\eta\rangle\ c_{\eta\iota 0}) \stackrel{\text{def}}{=} \frac{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{*}{\rightarrow} c_{\eta\iota}\ \ldots) \cdot \mathsf{P}_{\theta_{\text{G-L},d}}(c_{\eta\iota} \rightarrow c_{\eta\iota 0}\ c_{\eta\iota 1})}{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{+}{\rightarrow} c_{\eta\iota 0}\ \ldots)} \quad (41)$$

- for cross-element reductions:

$$\mathsf{P}_{\theta_{\text{R-R},d}}(c_{\eta\iota}, \mathbf{1} \,|\, \langle -, c_\eta\rangle\ \langle c'_{\eta\iota}, -\rangle) \stackrel{\text{def}}{=} [\![c_{\eta\iota} = c'_{\eta\iota}]\!] \cdot \frac{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{0}{\rightarrow} c_{\eta\iota}\ \ldots)}{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{*}{\rightarrow} c_{\eta\iota}\ \ldots)} \quad (42)$$

- for in-element reductions:

$$\mathsf{P}_{\theta_{\text{R-R},d}}(c_{\eta\iota}, \mathbf{0} \,|\, \langle -, c_\eta\rangle\ \langle c'_{\eta\iota}, -\rangle) \stackrel{\text{def}}{=} [\![c_{\eta\iota} = c'_{\eta\iota}]\!] \cdot \frac{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{+}{\rightarrow} c_{\eta\iota}\ \ldots)}{\mathsf{E}_{\theta_{\text{G-RL}^*,d}}(c_\eta \stackrel{*}{\rightarrow} c_{\eta\iota}\ \ldots)} \quad (43)$$

A sample phrase structure tree is shown as a right-corner transformed recursive structure in Figure 2,

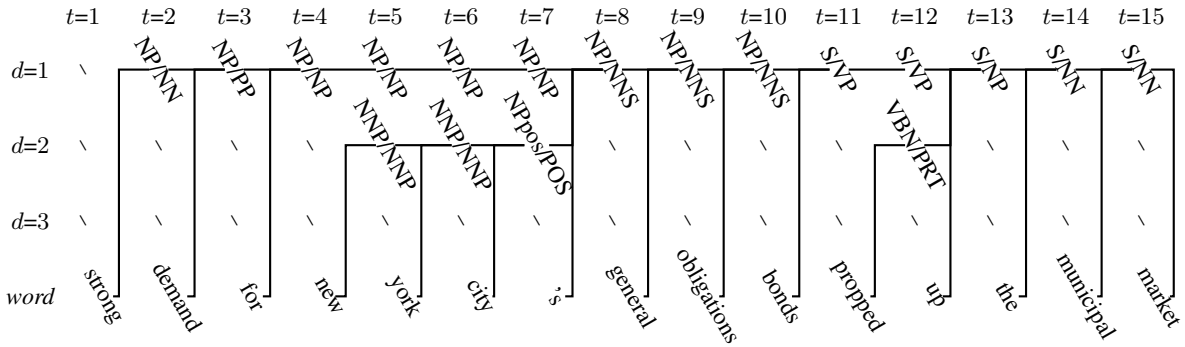


Figure 3: Sample tree from Figure 2 mapped to s_t^d variable positions of an HHMM at each depth level d (vertical) and time step t (horizontal). This tree uses at most only two memory store elements. Values for reduction states r_t^d are not shown.

and as a sequence of store states in Figure 3, corresponding to the output of Viterbi most likely sequence estimator. This estimation runs in linear time on the length of the input, so the parser can be run continuously on unsegmented or unpunctuated input. An ordinary phrase structure tree can be obtained by applying the transforms from Section 3, in reverse, to the right-corner recursive phrase structure tree represented in the sequence of store states.

5 Discussion

This paper has presented a derivation of a factored probabilistic sequence model from a probabilistic context-free grammar, using a right-corner transform to minimize memory usage in incremental processing. This sequence model characterization is attractive as a cognitive model because it does not posit any internal representation of complex phrasal structure beyond the pair of categories in each incomplete constituent resulting from the application of a right-corner transform; and because these incomplete constituents represent contiguous connected chunks of phrase structure, in line with characterizations of chunking in working memory (Miller, 1956). Experiments on large phrase-structure annotated corpora (Marcus et al., 1993) show this model could process the vast majority of sentences in a typical newspaper using only three or four store elements (Schuler et al., 2008; Schuler et al., 2010), in line with recent estimates of human short-term working memory capacity (Cowan, 2001).

This derivation can be applied to efficiently incrementalize any PCFG model, preserving the probabilistic dependencies in the original model. But, since the model is ultimately defined through transitions on entire working memory stores, it is also possible to relax the independence assumptions from the original PCFG model, and introduce additional dependencies across store elements that do not correspond to context-free dependencies. These additional dependencies might be used approximate dependencies of mildly context-sensitive grammar formalisms like tree adjoining grammars (Joshi, 1985), e.g. to model long-distance dependencies in filler-gap constructions (Kroch and Joshi, 1986), or crossed and nested dependencies in languages like Dutch (Shieber, 1985).

Figure 4 shows a sample store sequence corresponding to a parse of a noun phrase modified by an object relative clause. Here, the affix '-xNP' is used to identify a phrase containing an extracted NP constituent, and the category label 'GCnpS' is used to identify the maximal projection of a gapped clause. If the GC constituent (and only this constituent) is associated with the referent or dependency information of the filler constituent (the bike in this example), this information can be made available during processing from the immediately superior incomplete constituent at the gap position in the relative clause, without passing dependency information down the tree as any kind of feature (Pollard and Sag, 1994), even though this position is not adjacent to the filler in the phrase structure tree (they are separated by the referent or dependency information of the bridge verb 'said'). This is important to

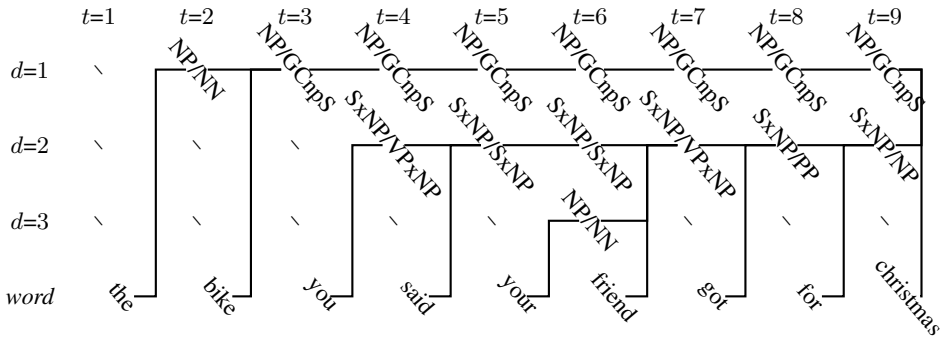


Figure 4: Sample store sequence containing long-distance dependency in a filler-gap construction.

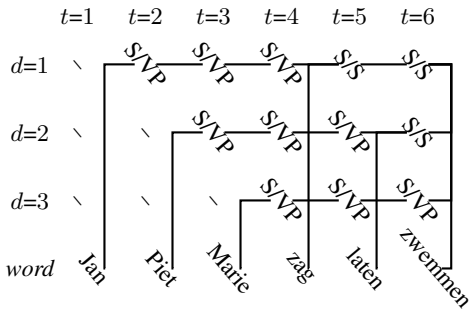


Figure 5: Sample store sequence containing crossed and nested dependencies.

satisfy claims that the model constructs chunks only for contiguous dependency structures (which is not true of propagated 'slash' features.

Figure 5 shows a sample store sequence corresponding to a parse of a Dutch sentence containing crossed and nested dependencies, featuring reductions across non-adjacent depth levels. This requires a more severe relaxation of PCFG independence assumptions, and is beyond the capacity of the HHMM as defined above, but this does preserve the notion of incomplete constituents and general composition established above, and is not beyond the capacity of factored sequence models with human-like memory bounds in general (note that the example sentence can still be parsed with only three store elements). This suggests a promising avenue of generalizing this model to learn parsing transitions that may be broader than those of a strict pushdown automaton.

References

Richard Bellman. 1957. *Dynamic Programming*. Princeton University Press, Princeton, NJ.

Nelson Cowan. 2001. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24:87–185.

Aravind K. Joshi. 1985. How much context sensitivity is necessary for characterizing structural descriptions: Tree adjoining grammars. In L. Karttunen D. Dowty and A. Zwicky, editors, *Natural language parsing: Psychological, computational and theoretical perspectives*, pages 206–250. Cambridge University Press, Cambridge, U.K.

Anthony S. Kroch and Aravind K. Joshi. 1986. The linguistic relevance of tree adjoining grammars. *Linguistics and Philosophy*.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

George A. Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97.

Kevin P. Murphy and Mark A. Paskin. 2001. Linear time inference in hierarchical HMMs. In *Proc. NIPS*, pages 833–840, Vancouver, BC, Canada.

Carl Pollard and Ivan Sag. 1994. *Head-driven Phrase Structure Grammar*. University of Chicago Press, Chicago.

William Schuler, Samir AbdelRahman, Tim Miller, and Lane Schwartz. 2008. Toward a psycholinguistically-motivated model of language. In *Proceedings of COLING*, pages 785–792, Manchester, UK, August.

William Schuler, Samir AbdelRahman, Tim Miller, and Lane Schwartz. 2010. Broad-coverage incremental parsing using human-like memory constraints. *Computational Linguistics*, 36(1).

William Schuler. 2009. Parsing with a bounded stack using a model-based right-corner transform. In *Proceedings of NAACL*, pages 344–352, Boulder, Colorado.

Stuart Shieber. 1985. Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8:333–343.