

A Tree Sampler for Bounded Context-Free Languages

Breandan Considine

Main Idea

- Analytic combinatorics: If you can count it, then you can sample it!
- We implement a bijection between labeled binary trees in BCFLs and $\mathbb{Z}_{|T|}$
- Allows for communication-free parallel no-replacement sampling in $\tilde{O}(1)$

Semiring Parsing

Given a CFG $\mathcal{G} := \langle V, \Sigma, P, S \rangle$ in Chomsky Normal Form (CNF), we may construct a recognizer $R_{\mathcal{G}} : \Sigma^n \rightarrow \mathbb{B}$ for strings $\sigma : \Sigma^n$ as follows. Let 2^V be our domain, where \emptyset is \emptyset , \oplus is \cup , and \otimes be defined as:

$$s_1 \otimes s_2 := \{C \mid \langle A, B \rangle \in s_1 \times s_2, (C \rightarrow AB) \in P\}$$

If we define $\hat{\sigma}_r := \{w \mid (w \rightarrow \sigma_r) \in P\}$, then construct a matrix with unit nonterminals on the superdiagonal, $M_0[r+1 = c](G', \sigma) := \hat{\sigma}_r$ the fixpoint $M_{i+1} = M_i + M_i^2$ is fully determined by the first diagonal:

$$M_0 := \begin{pmatrix} \emptyset & \hat{\sigma}_1 & \emptyset & \emptyset \\ & \ddots & \ddots & \ddots \\ \emptyset & & \hat{\sigma}_n & \emptyset \\ \emptyset & & \emptyset & \emptyset \end{pmatrix} \Rightarrow \begin{pmatrix} \emptyset & \hat{\sigma}_1 & \Lambda & \emptyset \\ & \ddots & \ddots & \ddots \\ \emptyset & & \hat{\sigma}_n & \emptyset \\ \emptyset & & \emptyset & \emptyset \end{pmatrix} \Rightarrow \dots \Rightarrow M_{\infty} = \begin{pmatrix} \emptyset & \hat{\sigma}_1 & \Lambda & \Lambda_{\sigma}^* \\ & \ddots & \ddots & \ddots \\ \emptyset & & \hat{\sigma}_n & \Lambda \\ \emptyset & & \emptyset & \emptyset \end{pmatrix}$$

we obtain the recognizer, $R(G', \sigma) := [S \in \Lambda_{\sigma}^*] \Leftrightarrow [\sigma \in \mathcal{L}(G)]$.

Parsing Dynamics

Let us consider an example with two holes, $\sigma = 1 _ _$, and the grammar being $G := \{S \rightarrow NON, O \rightarrow + \mid \times, N \rightarrow 0 \mid 1\}$. This can be rewritten into CNF as $G' := \{S \rightarrow NL, N \rightarrow 0 \mid 1, O \rightarrow + \mid \times, L \rightarrow ON\}$.

	2^V	$\mathbb{B}^{ V }$	$\mathbb{B}^{ V } \rightarrow \mathbb{B}^{ V }$
M_0	$\begin{pmatrix} \{N\} \\ \{N, O\} \\ \{N, O\} \end{pmatrix}$	$\begin{pmatrix} \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \end{pmatrix}$	$\begin{pmatrix} V_{0,1} \\ V_{1,2} \\ V_{2,3} \end{pmatrix}$
M_1	$\begin{pmatrix} \{N\} & \emptyset \\ \{N, O\} & \{L\} \\ \{N, O\} & \{N, O\} \end{pmatrix}$	$\begin{pmatrix} \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \end{pmatrix}$	$\begin{pmatrix} V_{0,1} & V_{0,2} \\ V_{1,2} & V_{1,3} \\ V_{2,3} & \end{pmatrix}$
M_{∞}	$\begin{pmatrix} \{N\} & \emptyset & \{S\} \\ \{N, O\} & \{L\} \\ \{N, O\} & \end{pmatrix}$	$\begin{pmatrix} \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \\ \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \blacksquare \end{pmatrix}$	$\begin{pmatrix} V_{0,1} & V_{0,2} & V_{0,3} \\ V_{1,2} & V_{1,3} \\ V_{2,3} & \end{pmatrix}$

This procedure returns whether $\exists \sigma' \in \mathcal{L}(G) \mid \sigma' \sqsubseteq \sigma$ but forgets genealogy.

Encoding CFL Sketching into SAT

- CYK parser can be lowered onto a Boolean tensor $\mathbb{B}^{n \times n \times |V|}$ (Valiant, 1975)
- Binarized CYK parser can be compiled to SAT to solve for M^* directly
- We simply encode the characteristic function, i.e. $1_{\subseteq V} : V \rightarrow \mathbb{B}^{|V|}$
- \oplus, \otimes are defined as \boxplus, \boxtimes , so that the following diagram commutes:

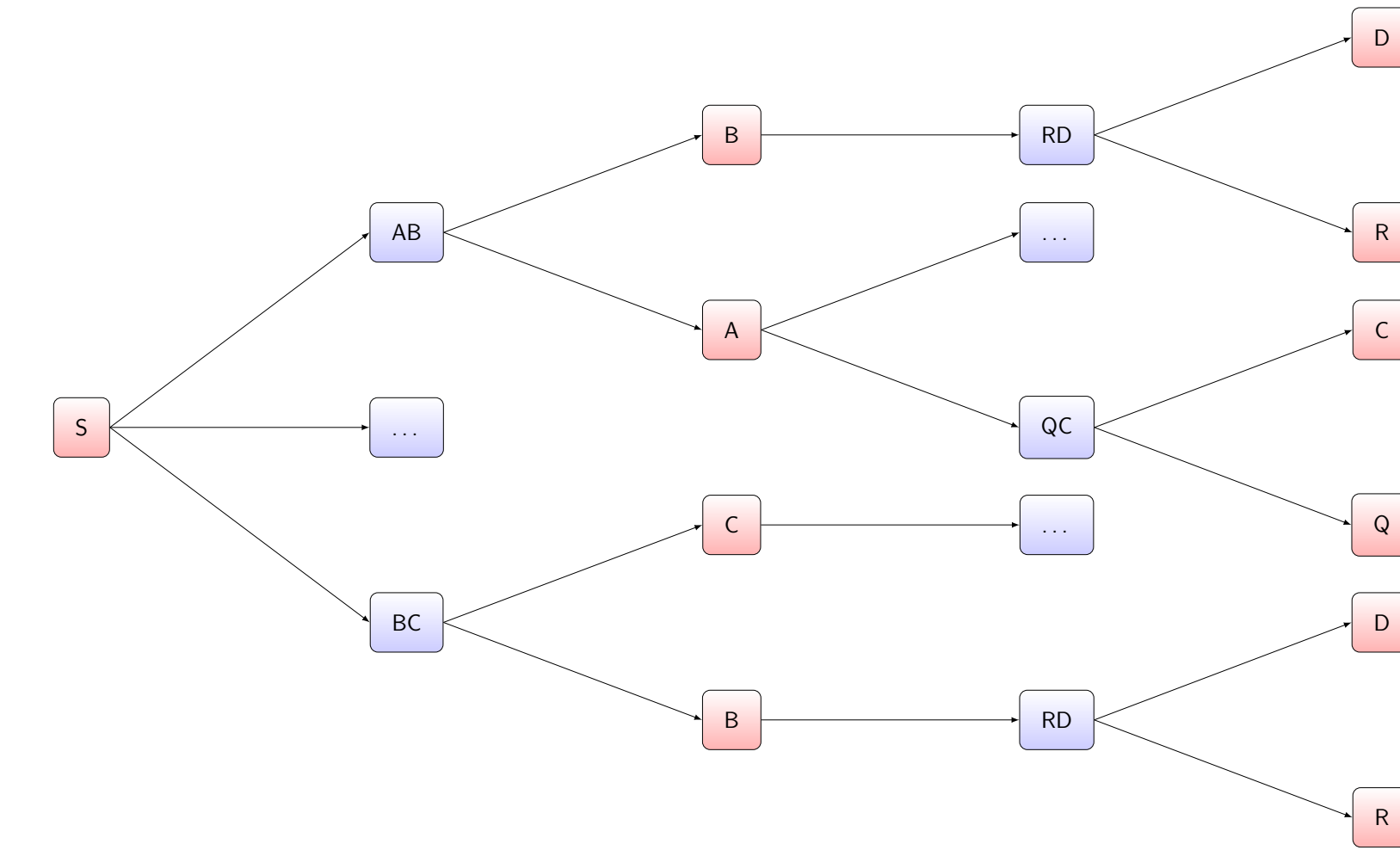
$$\begin{array}{ccc} 2^V \times 2^V & \xrightarrow{\oplus/\otimes} & 2^V \\ \uparrow 1^{-2} \downarrow 1^2 & & \uparrow 1^{-1} \downarrow 1 \\ \mathbb{B}^{|V|} \times \mathbb{B}^{|V|} & \xrightarrow{\boxplus/\boxtimes} & \mathbb{B}^{|V|} \end{array}$$

- These operators can be lifted into matrices and tensors in the usual manner

A Nested Datatype for BCFLs

We define an algebraic data type $\mathbb{T}_3 = (V \cup \Sigma) \multimap \mathbb{T}_2$ over the type $\mathbb{T}_2 = (V \cup \Sigma) \times (\mathbb{N} \multimap \mathbb{T}_2 \times \mathbb{T}_2)$, is identified by a recurrence relation:

$$L(p) = 1 + pL(p) \quad P(a) = V + aL(V^2P(a)^2)$$



Morally, \mathbb{T}_2 represents an implicit set of possible trees sharing the same root, and \mathbb{T}_3 is a dictionary of possible \mathbb{T}_2 values indexed by possible roots, given by a specific CFG under a porous string. We construct $\hat{\sigma}_r = \Lambda(\sigma_r)$ as follows:

$$\Lambda(s : \underline{\Sigma}) \mapsto \begin{cases} \bigoplus_{s \in \Sigma} \Lambda(s) & \text{if } s \text{ is a hole,} \\ \left\{ \mathbb{T}_2(w, [\langle \mathbb{T}_2(s), \mathbb{T}_2(\varepsilon) \rangle]) \mid (w \rightarrow s) \in P \right\} & \text{otherwise.} \end{cases}$$

We redefine the operations $\oplus, \otimes : \mathbb{T}_3 \times \mathbb{T}_3 \rightarrow \mathbb{T}_3$ over trees as follows:

$$X \oplus Z \mapsto \bigcup_{k \in \pi_1(X \cup Z)} \left\{ k \Rightarrow \mathbb{T}_2(k, x \cup z) \mid x \in \pi_2(X \circ k), z \in \pi_2(Z \circ k) \right\}$$

$$X \otimes Z \mapsto \bigoplus_{(w \rightarrow xz) \in P} \left\{ \mathbb{T}_2(w, [\langle X \circ x, Z \circ z \rangle]) \mid x \in \pi_1(X), z \in \pi_1(Z) \right\}$$

Sampling with replacement

Given a PCFG whose productions indexed by each nonterminal are decorated with a probability vector \mathbf{p} , we define a tree sampler $\Gamma : \mathbb{T}_2 \rightsquigarrow \mathbb{T}$ like so:

$$\Gamma(T) \mapsto \begin{cases} \text{Multi}(\text{children}(T), \mathbf{p}) & \text{if } T \text{ is a root} \\ \langle \Gamma(\pi_1(T)), \Gamma(\pi_2(T)) \rangle & \text{if } T \text{ is a child} \end{cases}$$

This relates to the generating function for the ordinary Boltzmann sampler,

$$\Gamma C(x) \mapsto \begin{cases} \text{Bern}\left(\frac{A(x)}{A(x)+B(x)}\right) \rightarrow \Gamma A(x) \mid \Gamma B(x) & \text{if } C = A + B \\ \langle \Gamma A(x), \Gamma B(x) \rangle & \text{if } C = A \times B \end{cases}$$

however unlike Duchon et al. (2004), our work does require rejection to ensure exact-size sampling, as all trees contained in \mathbb{T}_2 are necessarily the same width.

Sampling without replacement

To sample all trees in a given $T : \mathbb{T}_2$ uniformly without replacement, we construct a modular pairing function $\varphi : \mathbb{T}_2 \rightarrow \mathbb{Z}_{|T|} \rightarrow \text{BTree}$, defined as:

$$\varphi(T : \mathbb{T}_2, i : \mathbb{Z}_{|T|}) \mapsto \begin{cases} \langle \text{BTree}(\text{root}(T)), i \rangle & \text{if } T \text{ is a leaf,} \\ \text{Let } b = |\text{children}(T)|, \\ q_1, r = \langle \lfloor \frac{i}{b} \rfloor, i \pmod{b} \rangle, \\ lb, rb = \text{children}[r], \\ T_1, q_2 = \varphi(lb, q_1), \\ T_2, q_3 = \varphi(rb, q_2) \text{ in} \\ \langle \text{BTree}(\text{root}(T), T_1, T_2), q_3 \rangle & \text{otherwise.} \end{cases}$$

Then instead of sampling trees, we can simply sample integers WOR from $\mathbb{Z}_{|T|}$.

