A Tree Sampler for Bounded Context-Free Languages

Breandan Considine

Main Idea

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

Semiring Parsing

Given a CFG $\mathcal{G} = \langle V, \Sigma, P, S\rangle$ in Chomsky Normal Form (CNF), we may construct a recognizer $R_{\mathcal{G}} : \Sigma^n \to \mathbb{B}$ for strings $\sigma : \Sigma^n$ as follows. Let 2^V be our domain, where 0 is $\varnothing$, $\oplus$ is $\cup$, and $\otimes$ be defined as:

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

If we define $\hat{\sigma}_r = \{w \mid (w \to \sigma_r) \in P\}$, 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:

$$M_0 = \begin{pmatrix} \varnothing & \hat{\sigma}_1 & \varnothing & \varnothing \\ & & & \varnothing \\ & & & \hat{\sigma}_n \\ \varnothing & & & \varnothing \end{pmatrix} \Rightarrow \begin{pmatrix} \varnothing & \hat{\sigma}_1 & \Lambda & \varnothing \\ & & & \Lambda \\ & & & \hat{\sigma}_n \\ \varnothing & & & \varnothing \end{pmatrix} \Rightarrow \ldots \Rightarrow M_\infty = \begin{pmatrix} \varnothing & \hat{\sigma}_1 & \Lambda & \Lambda_\sigma^* \\ & & & \Lambda \\ & & & \hat{\sigma}_n \\ \varnothing & & & \varnothing \end{pmatrix}$$

CFL membership is recognized by $R(G', \sigma) = [S \in \Lambda_\sigma^*] \Leftrightarrow [\sigma \in \mathcal{L}(G)]$.

Parsing Dynamics

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

	2^V	$\mathbb{B}^{\lvert V\rvert}$	$\mathbb{B}^{\lvert V\rvert} \to \mathbb{B}^{\lvert V\rvert}$
M_0	$\begin{pmatrix} \{N\} & & \\ & \{N,O\} & \\ & & \{N,O\} \end{pmatrix}$	$\begin{pmatrix} \square\blacksquare\square\square & & \\ & \square\blacksquare\blacksquare\square & \\ & & \square\blacksquare\blacksquare\square \end{pmatrix}$	$\begin{pmatrix} V_{0,1} & & \\ & V_{1,2} & \\ & & V_{2,3} \end{pmatrix}$
M_1	$\begin{pmatrix} \{N\} & \varnothing & \\ & \{N,O\} & \{L\} \\ & & \{N,O\} \end{pmatrix}$	$\begin{pmatrix} \square\blacksquare\square\square & \square\square\square\square & \\ & \square\blacksquare\blacksquare\square & \blacksquare\square\square\square \\ & & \square\blacksquare\blacksquare\square \end{pmatrix}$	$\begin{pmatrix} V_{0,1} & V_{0,2} & \\ & V_{1,2} & V_{1,3} \\ & & V_{2,3} \end{pmatrix}$
M_∞	$\begin{pmatrix} \{N\} & \varnothing & \{S\} \\ & \{N,O\} & \{L\} \\ & & \{N,O\} \end{pmatrix}$	$\begin{pmatrix} \square\blacksquare\square\square & \square\square\square\square & \square\square\square\blacksquare \\ & \square\blacksquare\blacksquare\square & \blacksquare\square\square\square \\ & & \square\blacksquare\blacksquare\square \end{pmatrix}$	$\begin{pmatrix} V_{0,1} & V_{0,2} & V_{0,3} \\ & V_{1,2} & V_{1,3} \\ & & V_{2,3} \end{pmatrix}$

This procedure decides if $\exists \sigma' \in \mathcal{L}(G) \mid \sigma' \sqsubseteq \sigma$ but forgets provenance.

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 $\mathbf{M}^*$ directly
- We simply encode the characteristic function, i.e., $\mathbb{1}_{\subseteq V} : 2^V \to \mathbb{B}^{|V|}$
- $\oplus, \otimes$ are defined as $\boxplus, \boxtimes$, so that the following diagram commutes:

$$\begin{tikzcd} 2^V \times 2^V \arrow[r, "\oplus/\otimes"] \arrow[d, shift left, "\mathbb{1}^2"] & 2^V \arrow[d, shift left, "\mathbb{1}"] \\ \mathbb{B}^{|V|} \times \mathbb{B}^{|V|} \arrow[r, "\boxplus/\boxtimes"'] \arrow[u, shift left, "\mathbb{1}^{-2}"] & \mathbb{B}^{|V|} \arrow[u, shift left, "\mathbb{1}^{-1}"] \end{tikzcd}$$

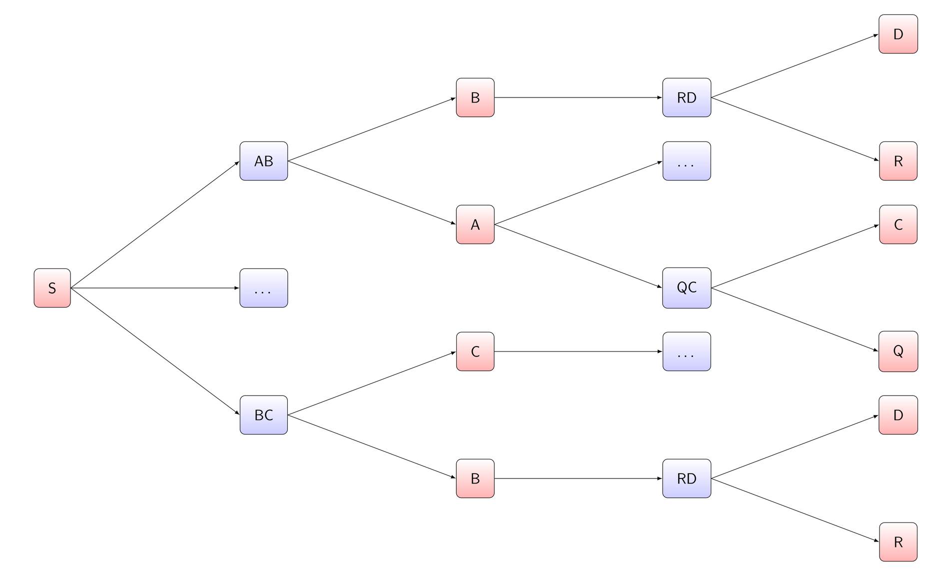
- These operators can be lifted into matrices and tensors in the usual manner

A Nested Datatype for BCFLs

We define a map from nonterminals $\mathbb{T}_3 = (V \cup \Sigma) \rightharpoonup \mathbb{T}_2$ onto the datatype $\mathbb{T}_2 = (V \cup \Sigma) \times (\mathbb{N} \rightharpoonup \mathbb{T}_2 \times \mathbb{T}_2)$, whose inhabitants satisfy the recurrence:

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

Each $\mathbb{T}_2$ consists of a root nonterminal, and a list of distinct products, e.g.,



Morally, $\mathbb{T}_2$ represents an implicit set of possible trees sharing the same root, where $\mathbb{T}_3$ is a dictionary of possible $\mathbb{T}_2$ values indexed by possible roots, given by a specific CFG under a porous string. Instead of $\hat{\sigma}_r$ we initialize using $\Lambda(\sigma_r)$:

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

The operations $\oplus, \otimes : \mathbb{T}_3 \times \mathbb{T}_3 \to \mathbb{T}_3$ are then redefined over trees as follows:

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

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

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:

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

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

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

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 define a modular pairing function $\varphi : \mathbb{T}_2 \to \mathbb{Z}_{|T|} \to \texttt{BTree}$ using the construction:

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

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