Propagation of Syntax Errors in Context-Sensitive Languages using the Lifted Brzozowski Derivative

Anonymous Author(s)

Abstract

Brzozowski (1964) defines the derivative of a regular language as the suffixes that complete a known prefix. In this work, we establish a Galois connection with Valiant's (1975) fixpoint construction in the context-free setting, and further extend their work into the hierarchy of bounded context-sensitive languages realizable by finite CFL intersection. We show how context-sensitive language recognition can be reduced into a tensor algebra over finite fields, drawing a loose analogy to partial differentiation in Euclidean spaces. In addition to its theoretical contributions, our method has yielded applications to incremental parsing, code completion and program repair. For example, we use it to repair syntax errors and perform sketch-based program synthesis, among other language decision problems.

1 Introduction

Recall that a CFG is a quadruple consisting of terminals (Σ), nonterminals (V), productions ($P\colon V \to (V \mid \Sigma)^*$), and a start symbol, ($S$). It is a well-known fact that every CFG is reducible to *Chomsky Normal Form*, $P'\colon V \to (V^2 \mid \Sigma)$, in which every production takes one of two forms, either $w \to xy$, or $w \to t$, where $w, x, y : V$ and $t : \Sigma$. For example, the CFG, $P := \{S \to SS \mid (S) \mid ()\}$, corresponds to the CNF:

$$P' = \left\{ S \to XR \mid SS \mid LR, \quad L \to (, \quad R \to), \quad X \to LS \right\}$$

Given a CFG, $\mathcal{G}' : \langle \Sigma, V, P, S \rangle$ in CNF, we can construct a recognizer $R : \mathcal{G}' \to \Sigma^n \to \mathbb{B}$ for strings $\sigma : \Sigma^n$ as follows. Let 2^V be our domain, 0 be $\varnothing$, $\oplus$ be $\cup$, and $\otimes$ be defined as:

$$x \otimes y := \left\{ W \mid \langle X, Y \rangle \in x \times y, (W \to XY) \in P \right\} \quad (1)$$

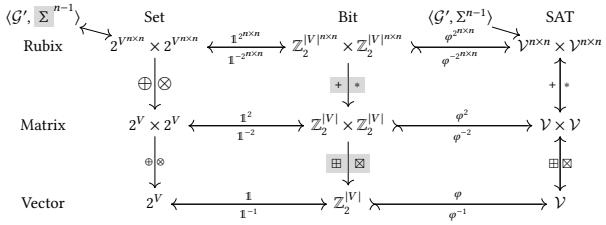
We initialize $\mathbf{M}^0_{r,c}(\mathcal{G}', e) := \{V \mid c = r + 1, (V \to \sigma_r) \in P\}$ and search for a matrix $\mathbf{M}^*$ via fixpoint iteration,

$$\mathbf{M}^* = \begin{pmatrix} \varnothing & \{V\}_{\sigma_1} & \cdots & \mathcal{T} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \{V\}_{\sigma_n} \\ \varnothing & \cdots & \cdots & \varnothing \end{pmatrix} \quad (2)$$

where $\mathbf{M}^*$ is the least solution to $\mathbf{M} = \mathbf{M} + \mathbf{M}^2$. We can then define the recognizer as: $S \in \mathcal{T}? \iff \sigma \in \mathcal{L}(\mathcal{G})?$
Full details of the bisimilarity between parsing and matrix multiplication can be found in Valiant [4] and Lee [3], who shows its time complexity to be $\mathcal{O}(n^\omega)$ where ω is the least matrix multiplication upper bound (currently, $\omega < 2.77$).

2 Method

Note that $\bigoplus_{k=1}^{n} \mathbf{M}_{ik} \otimes \mathbf{M}_{kj}$ has cardinality bounded by $|V|$ and is thus representable as a fixed-length vector using the characteristic function, $\mathbb{1}$. In particular, $\oplus, \otimes$ are defined as $\boxplus, \boxtimes$ (respectively), so that the following diagram commutes:[1]



Where $\mathcal{V}$ is the domain of XOR-SAT expressions, i.e., linear equations over $GF(2)$. Note that while always possible to encode an element of $\mathbb{Z}_2^{|V|}$ into $\mathcal{V}$, φ^{-1} may not exist, as an arbitrary $\mathcal{V}$ might take on zero (i.e., be UNSAT), one, or in general, many values in $\mathbb{Z}_2^{|V|}$. Let us consider two cases, where $\mathbf{M}^*$ is either left- or right-constrained, i.e., $\alpha\ \gamma$, $\gamma\ \alpha$.

Valiant's $\otimes$ operator, which solves for the set of productions unifying known factors in a binary CFG, implies the existence of a left- and right-quotient, which yield the set of nonterminals that may appear to the right- or left-side, respectively, of known factors in a ternary configuration.

Left Quotient

$$\frac{\partial f}{\partial \vec{x}} = \left\{ y \mid (w \to xy) \in P \right\}$$

x	w
	y

Right Quotient

$$\frac{\partial f}{\partial \vec{y}} = \left\{ x \mid (w \to xy) \in P \right\}$$

x	w
	y

The left quotient coincides with the derivative operator first proposed by Brzozowski [2] and Antimirov [1] over regular languages, lifted into the context-free setting (our work).

Let $\mathcal{V}$ represent 2^V. Assuming the root is known, (e.g., S), the operator $\frac{\partial S}{\partial \vec{x}} : (\vec{V} \to S) \to \vec{\mathcal{V}}$ is a dependently-typed function which returns the unknown RHS factor. We may also define a gradient operator, $\vec{\nabla} S : (\vec{\mathcal{V}} \to S) \to \vec{\mathcal{V}}$, which tracks the partials with respect to multiple unknown factors.

If the root itself is unknown, we can define an operator, $\mathcal{H}_{\mathcal{W} \subseteq \mathcal{V}} : (\vec{\mathcal{V}} \times \vec{\mathcal{V}} \times \mathcal{W}) \to (\vec{\mathcal{V}} \times \vec{\mathcal{V}} \to \mathcal{W})$, which tracks second-order partial derivatives for all roots in $\mathcal{W}$. Unlike

[1] Hereinafter, we use gray highlighting to distinguish between expressions containing only constants from those which may contain free variables.

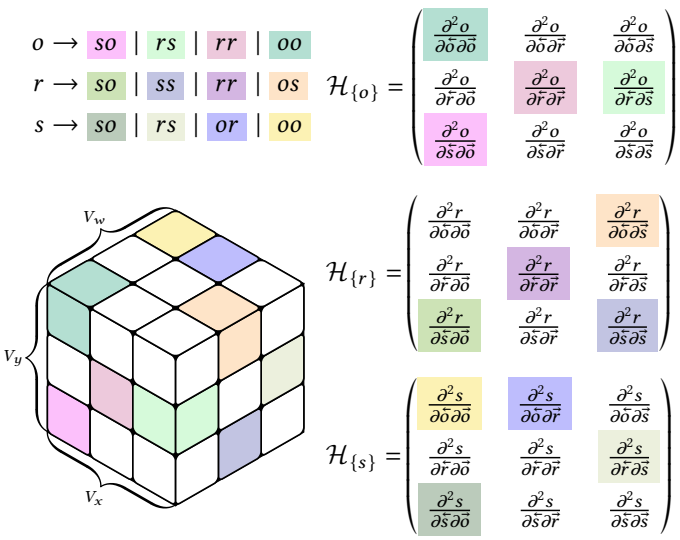
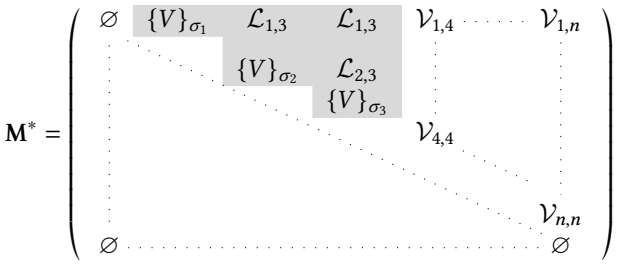


Figure 1. CFGs are witnessed by a rank-3 binary tensor, whose nonzero entries indicate CNF productions. Derivatives in this setting effectively condition the parse tensor. By backpropagating $\mathcal{H}$ across upper-triangular entries of $\mathbf{M}^*$, we constrain the superposition of admissible parse forests.

differential calculus on smooth manifolds, partials in this calculus do not necessarily commute depending on the CFG. By allowing the matrix $\mathbf{M}^*$ to contain bitvector variables representing holes in σ, we obtain a set of multilinear equations whose solutions exactly correspond to the set of admissible repairs and their corresponding parse forests. Specifically, the repairs coincide with holes in the superdiagonal $\mathbf{M}^*_{r+1=c}$, and the parse forests occur along upper-triangular entries $\mathbf{M}^*_{r+1<c}$. In the case depicted below, $\mathbf{M}^*$ is left-constrained, although the holes may (in general) appear anywhere in σ:



We also need the constraint that no two conflicting nonterminals may be active at any given time… TODO: describe uniqueness constraint

2.1 Context-Sensitive Reachability

It is well-known that the family of CFLs is not closed under intersection. For example, consider $\mathcal{L}_\cap := \mathcal{L}(\mathcal{G}_1) \cap \mathcal{L}(\mathcal{G}_1)$:

$$P_1 := \{\ S \to LR,\quad L \to ab \mid aLb,\quad R \to c \mid cR\ \}$$
$$P_2 := \{\ S \to LR,\quad R \to bc \mid bRc,\quad L \to a \mid aL\ \}$$

Note that $\mathcal{L}_\cap$ generates the language $\{\ a^d b^d c^d \mid d > 0\ \}$, which according to the pumping lemma is not context free. In our formalism, we encode finite intersections of CFLs $\bigcap_{i=0}^{c} \mathcal{L}(\mathcal{G}_i)$ as a prism with upper-triangular matrices joined to each rectangular face, like the fletches of an arrow. As $c \to \infty$, this shape approximates a right circular cone whose symmetric axis intersects equivalent V_i's in each CNF unit production $w \to t$, and base perimeter represents $S_i \in \mathcal{T}_i$. Equations of this form are equiexpressive with the family of CSLs realizable by intersecting a finite collection of CFLs.

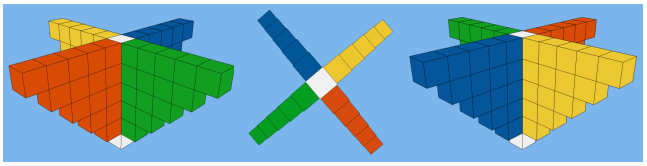


Figure 2. Orientations of a $\bigcap_{i=1}^{4} \mathcal{L}(\mathcal{G}_i) \cap \Sigma^6$ configuration.

More precisely, we intersect all terminals $\Sigma_\cap := \bigcap_{i=1}^{c} \Sigma_i$, then for each $t \in \Sigma_\cap$, we construct an equivalence class $E(t, \mathcal{G}_i) = \{w_i \mid (w_i \to t) \in P_i\}$, and glue them all together:

$$\bigwedge_{t \in \Sigma_\cap} \bigwedge_{j=0}^{c-2} \bigwedge_{i=0}^{|\sigma|-1} E(t, \mathcal{G}_j) \equiv_{\sigma_i} E(t, \mathcal{G}_{j+1}) \tag{3}$$

Together with the uniqueness constraint this ensures that at least one representative of each equivalence class is included… (TODO).

Although emptiness for CSLs is, in general, undecidable, unsatisfiability corresponds to emptiness of bounded CSLs. Anyhow, since we are working with bounded-width CSLs, everything collapses down to finite languages, which are always closed. Thus, Bar-Hillel and other elegant constructions become trivial when so restricted. So we can use it to decide impossible substrings and other decision problems which are typically intractable in more general settings.

3 Applications

This technique can be used to repair syntax errors in programming languages.

References

[1] Valentin Antimirov. 1996. Partial derivatives of regular expressions and finite automaton constructions. Theoretical Computer Science 155, 2 (1996), 291–319.

[2] Janusz A Brzozowski. 1964. Derivatives of regular expressions. Journal of the ACM (JACM) 11, 4 (1964), 481–494. http://maveric.uwaterloo.ca/reports/1964_JACM_Brzozowski.pdf

[3] Lillian Lee. 2002. Fast context-free grammar parsing requires fast boolean matrix multiplication. Journal of the ACM (JACM) 49, 1 (2002), 1–15. https://arxiv.org/pdf/cs/0112018.pdf

[4] Leslie G Valiant. 1975. General context-free recognition in less than cubic time. Journal of computer and system sciences 10, 2 (1975), 308–315. http://people.csail.mit.edu/virgi/6.s078/papers/valiant.pdf