

A Word Sampler for Well-Typed Functions

Breandan Mark Considine

January 4, 2026

Formal languages & type theory

$$\underbrace{\sigma \in \mathcal{L}(G) \Leftrightarrow \exists V. V \Rightarrow_G^* \sigma}_{\text{membership / parse tree}} \quad \Leftrightarrow \quad \underbrace{\exists \tau. (\Gamma \vdash e : \tau)}_{\text{type checking / proof tree}}$$

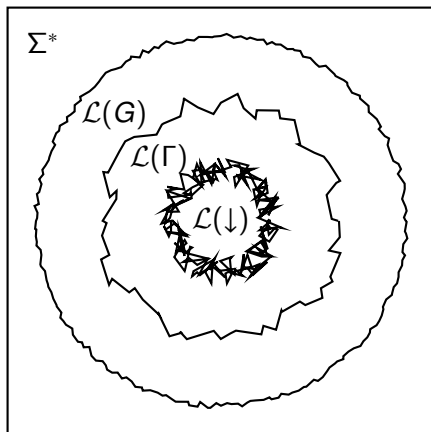
$$\underbrace{(W \rightarrow XZ) \in P}_{\text{grammar production}} \quad \Leftrightarrow \quad \underbrace{\frac{\Gamma \vdash x : X \quad \Gamma \vdash z : Z}{\Gamma \vdash xz : W}}_{\text{typing judgment}}$$

$$\underbrace{\mathcal{L}(G) \neq \emptyset \Leftrightarrow \exists \sigma. S \Rightarrow_G^* \sigma}_{\text{non-emptiness / generation}} \quad \Leftrightarrow \quad \underbrace{\exists e. (\Gamma \vdash e : \tau)}_{\text{type inhabitation / synthesis}}$$

Goal: Given a set of typing judgements and a typing context (Γ) , design a grammar, G , such that $\forall \sigma \in \mathcal{L}(G) \cap \Sigma^{<n} \exists \tau. \Gamma \vdash \sigma : \tau$ and furthermore, $\forall \sigma \in \Sigma^{<n}. \Gamma \vdash \sigma : \tau \implies \sigma \in \mathcal{L}(G)$.

Programming language [in]approximability

- ▶ Σ^* : all words over Σ
- ▶ $\mathcal{L}(G)$: syntactically valid
- ▶ $\mathcal{L}(\Gamma)$: type-safe programs
- ▶ $\mathcal{L}(\downarrow)$: halting programs
- ▶ Most LLMs: $\sigma \leftarrow \Sigma^*$
- ▶ Typesafe: $\sigma \leftarrow \mathcal{L}(\Gamma)$
- ▶ Tighter approximations require ever-increasing expressive power
- ▶ Volumes are not to scale



High-level grammar embedding recipe

- ▶ Fix a finite type universe \mathbb{T} and an ambient global context Γ
- ▶ Decorate vanilla nonterminals with a typing annotation, $E[\tau]$
- ▶ Each typing judgment becomes a schema for constructing a family of synthetic productions each instantiated by valid $\tau : \mathbb{T}$

Syntax:
$$\frac{\Gamma \vdash e_1 : \tau_1, \dots, e_m : \tau_m \quad \Phi(\Sigma, \tau_1, \dots, \tau_m) : \tau}{(E[\tau] \rightarrow \Phi(\Sigma, \tau_1, \dots, \tau_m)) \in P_\Gamma}$$

Names:
$$\Gamma \vdash e : \tau \Rightarrow (E[\tau] \rightarrow \mathbf{e}) \in P_\Gamma$$

Functions:
$$\frac{\Gamma \vdash f : (\tau_1, \dots, \tau_k) \rightarrow \tau}{(E[\tau_1] \rightarrow \mathbf{f} \ (E[\tau_1] \ , \ \dots \ , \ E[\tau_k] \)) \in P_\Gamma}$$