Constrained Decoding

Manu Bhat Arnav Dandu Kanghee Park

March 27, 2025

0.1 Introduction

0.2 Algorithms

Definition 1 (Finite-State Automata)**.** todo

Definition 2 (Context-Free Grammar)**.** todo

Definition 3 (Finite-State Transducer)**.** todo

Definition 4 (Pushdown Automata)**.** todo

Definition 5 (Checker)**.** todo

Definition 6 (ConstrainedDecoding)**.**

```
Algorithm: ConstrainedDecoding
Input: Model $M$, Checker $C$, Tokenized prompt $x$
$\mathcal{V} := M.\texttt{vocabulary}$
repeat
    $m := C(x; \mathcal{V})$
    $\mathit{logits} := M(x)$
    $t_{\text{next}} := \texttt{sample}(\texttt{applyMask}(m, \mathit{logits}))$
    $x := x.\texttt{append}(t_{\text{next}})$
until $t_{\text{next}} \neq \texttt{EOS}$
return $x$
```

Definition 7 (Partial Lexer)**.** todo

Definition 8 (Lexing Transducer)**.** todo

Definition 9 (BuildLexingFST)**.** todo

Theorem 10 (Lexing Transducer Equivalent to `Lex`)**.** *Let* $\mathcal{T}_A = (Q, \Sigma, \Gamma, q_0, \delta, F)$ *be a lexing transducer for the lexer specification* $\{(\mathcal{A}^i, T^i)\}_i$*. Then*

$$q_0 \xrightarrow{w:T_1 \ldots T_k}{}^{*} q' \in \delta^* \quad \textit{if and only if} \quad \texttt{Lex}(w) = (T_1 \ldots T_k, w_r)$$

for some $w_r \in \Sigma^*$ *and* $q' \in Q$ *such that* $q_0 \xrightarrow{w_r:\epsilon}{}^{*} q'$*.*

Definition 11 (LanguageModel)**.** todo

Definition 12 (Detokenizer)**.** todo

Definition 13 (BuildDetokenizingFST)**.** todo

Definition 14 (Parser)**.** todo

Definition 15 (PreprocessParser)**.** todo