Extended SCFGs for LTR Identification

M.L. Souza
University of California Berkeley
Biophysics

August 22, 2011

We consider slightly extended stochastic context-free grammars for use in parsing languages of limited repeats.

1 Overview

Background

2 Extending SCFGs

Consider grammar $G = (N,\ S,\ T,\ P,\ \psi)$
where N are non-terminal symbols, $S \in N$ the start symbol, T terminals, P a set of production rules, and ψ the probability distribution over the production rules P.

2.1 Production Rules

In the following, let $n, m \in T$ be a terminal symbols, and $x,\ x' \in T^*$ be strings.
We form a superset of "RNA normal form" for SCFGs (Reference? Durbin?):

1. Bifurcation: $L \rightarrow R\ M$
2. Pass-through: $L \rightarrow R$
3. Left emission: $L \rightarrow nR$
4. Right emission: $L \rightarrow Rn$
5. Paired emission: $L \rightarrow mRn$
6. Terminal emission: $L \rightarrow n$
7. Null emission: $L \rightarrow \epsilon$

Adding an additional rule:

8. Repeat emission: $L \rightarrow R_{\text{rep}}(M)$

Where $R_{\text{rep}}(M)$ is a distinguished non-terminal in which: $L \rightarrow R_{\text{rep}}(M) \Leftrightarrow L \rightarrow xMx'$
With $x,\ x' \in T^*$, and x' an approximate-repeat of x, to be made more precise below.

2.1.1 Repeat Emissions

Non-terminals such as $R_{\text{rep}}(M)$ are an embedded constrained linear indexed grammar defined by the following rules:

1. $R_{\text{rep}} \to X[]$
2. $X[\sigma] \to nX[\sigma\ n]$
3. $X[\sigma] \to X'[\sigma]$
4. $X'[\sigma n] \to X'[\sigma]m$
5. $X'[] \to M$

The above rules define a grammar capable of generating the language:

$$L_{\text{rep}} = \left\{xmx' \mid x, x', m \in T^*\right\}$$

With $|w| = |w'|$ and each terminal w_i dictating the probability of emitting symbol w'_i for $0 \leq i \leq |w|$, and m denoting the substring generated by the non-terminal M.
I.e. it generates repetitions of precisely the same length with pointwise mutations.

We will show that parsing of an extended SCFG having repeat emissions as above can be performed in $\mathrm{O}(N^4)$ time, where N is the length of the input string.

We can extend the repeat grammar to allow insertions and deletions by introducing the following production rules:

6. $X'[\sigma n] \to X'[\sigma]$ (Popping a symbol off the stack; corresponds to a deletion)
7. $X'[\sigma] \to X'[\sigma]m$ (A right-emission without stack modification; corresponds to an insertion.)

An important property of this grammar is that the growing stack for non-terminal X is exactly the substring which the grammar emitted.

2.2 Parsing

We now consider an extension of the CYK algorithm to determine the maximum-likelihood parse for a given input string s.

2.2.1 Recursive Definition

We first give a recursive definition for each element of matrix $C \in \mathbb{R}^{|s|} \times \mathbb{R}^{|s|} \times N$.
Let $C(i, j-i+1, S) = \max_{\substack{\text{parse trees } \pi \\ \text{deriving } x_{i \cdots j}}} P(\pi)$ be defined as follows:

$$C(i, j-i+1, S) = \max \begin{cases} \max\limits_{R,M} \max\limits_{0 \leq k \leq j-i} C(i,k,R)C(i+k, j-k, M)P(L \to RM) \\ \max\limits_{R} C(i, j-i, R)P(L \to R) \\ \max\limits_{R} P(L \to nR)C(i+1, j-(i+1), R) \\ \max\limits_{R} C(i, j-1, R)P(L \to Rn) \\ \max\limits_{R} P(L \to nRm)C(i+1, j-(i+2), R) \\ P(L \to x_i) \\ P(L \to \epsilon) \\ f(i,j) \text{ defined below} \end{cases}$$

We consider $f(i, j)$ for two cases, with and without rules 6 & 7 of the repeat SLIG defined above.

For the grammar including pointwise mutations, insertions, and deletions (including rules 6 and 7):

In the grammar of pointwise mutations, the repetitive strings are necessarily the same length and so we simplify parsing by one degree of freedom.

$$f(i,j) = \max_{M} \max_{0 \leq k \leq \lfloor \frac{j-i}{2} \rfloor} (LHS)C(c-k, k, M)(RHS)P(L \to R_{\text{rep}}(M))P(X'[] \to M)$$

Where $c = \lfloor \frac{i+j}{2} \rfloor$, and:

$$LHS = \prod_{l=i}^{c-k-1} P(X[\sigma] \to x_l X[\sigma x_l])$$

$$RHS = \prod_{l=i}^{c+k+1} P(X'[\sigma x_l] \to X'[\sigma] x_l)$$

2.2.2 Pseudocode Implementation

adsf

2.3 Training

Each repeat non-terminal $R_{\text{ltr}}(M)$ is parametrized by .

References