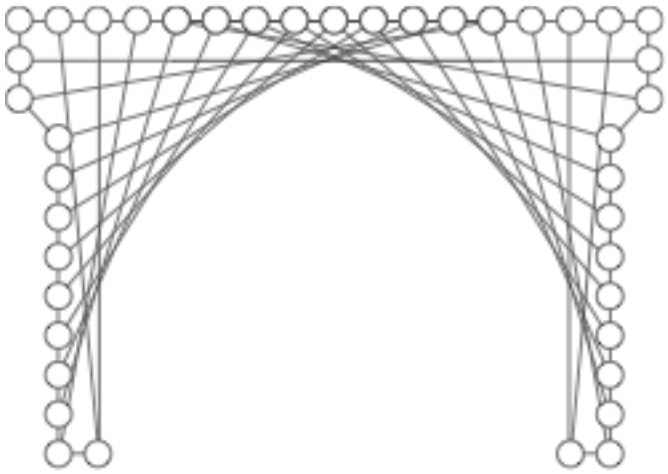
Bioinformatics

Richard Bonneau

Lecture 7: transformational grammars pre RNA structure.



NEW YORK UNIVERSITY
CENTER FOR COMPARATIVE
FUNCTIONAL GENOMICS

COURANT
INSTITUTE

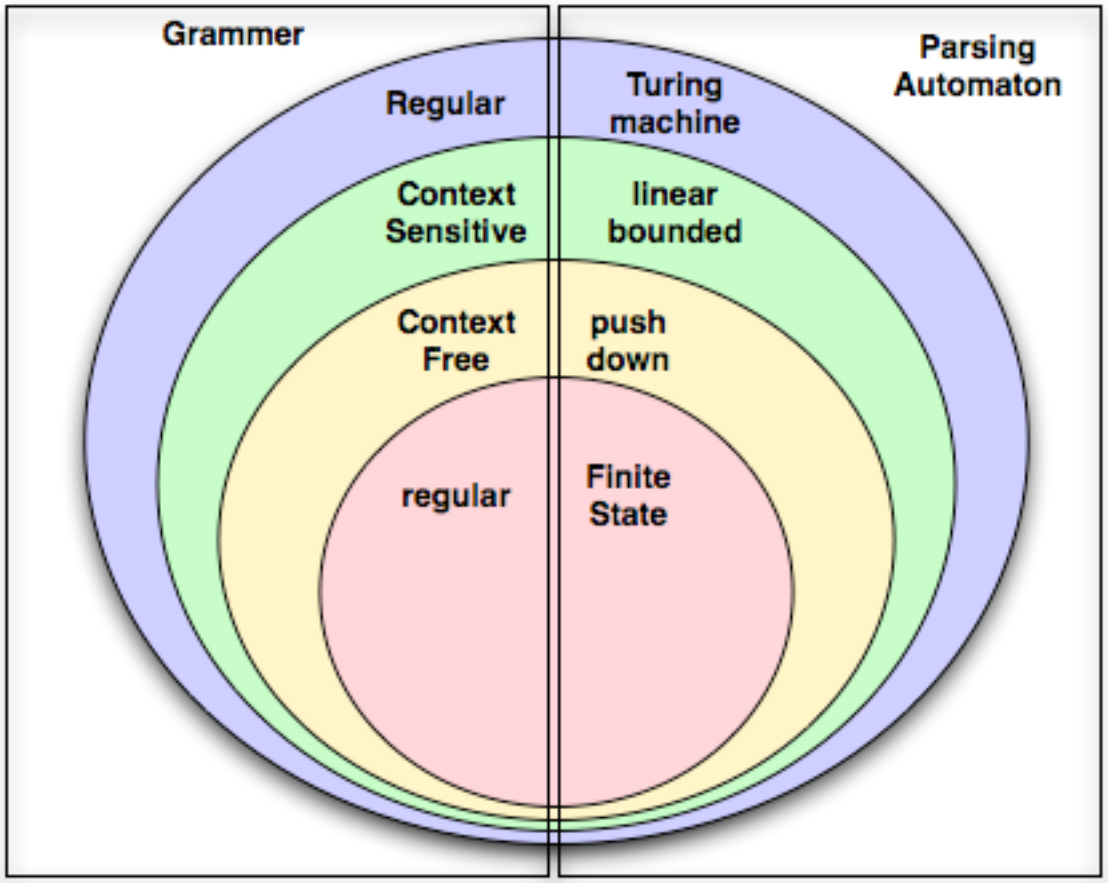
Associated reading.

- Ch. 9 BSA : Preparing for RNA structure prediction
- Berezikov, Cuppen & Plasterk. Approaches to microRNA discovery. NATURE GENETICS SUPPLEMENT. S2 VOLUME 38.JUNE 2006

Hierarchy of transformational grammars

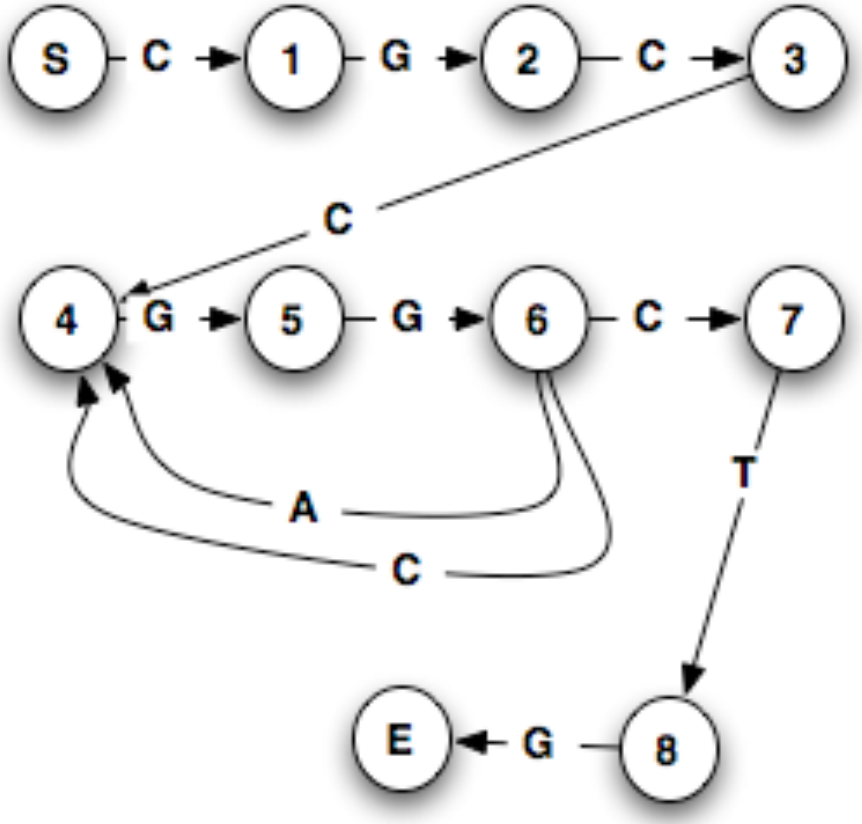
eg: Generate odd number of a terminals:
S -> aT | bS
T -> aS | bT | e

S->aT->aaS->aabS->aabaT ->aaba**e**



Regular:
W->aW
W->a
Context Free:
W->[[a+][W+]] in any combinatior
only one nonterminal on left
Context Sensitive:
a_1Wa_2->a_1[[a+][W+]]a_2
Unrestricted:
a_1Wa_2-> *
Anything goes

A FSA / regular grammar for recognizing a specific sequence with variable repeat



this FSA parses/recognizes:
CG CGG CGG AGG AGG CTG
OR
CG AGG CGG CGG CTG

In rewriting rules:

$S \rightarrow cW_1$ $W_1 \rightarrow gW_2$
$W_2 \rightarrow cW_3$ $W_3 \rightarrow cW_4$
$W_4 \rightarrow gW_5$ $W_5 \rightarrow gW_6$
$W_6 \rightarrow aW_4 | cW_4 | cW_7$
$W_7 \rightarrow tW_8$
$W_8 \rightarrow ge$

Prosite: regular grammars and profiles to detect protein domains and motifs

P - x - [STA] - x - [LIV] - [IVT] - x - [GS] - G - Y - S - [QL] - G

We know the active site is S at position 11.

Cutinase pattern

$S \rightarrow pW_1$
$W_1 \rightarrow xW_2$
$W_2 \rightarrow sW_3 \mid tW_3 \mid aW_3$
$W_3 \rightarrow xW_4$
$W_4 \rightarrow lW_5 \mid iW_5 \mid vW_5$
$W_5 \rightarrow vW_6 \mid iW_6 \mid tW_6$
$W_6 \rightarrow xW_7$
$W_8 \rightarrow gW_9 \mid sW_9$
$W_9 \rightarrow gW_{10}$

Prosite: regular grammars and profiles to detect protein domains and motifs

Technical section:

PROSITE methods (with tools and information) covered by this documentation:

TRYPSIN_DOM, PS50240; **Serine proteases, trypsin domain profile** (MATRIX)

Sequences known to belong to this class detected by the profile:	ALL
Other sequence(s) detected in Swiss-Prot:	NONE.

- Domain architecture view of Swiss-Prot proteins matching PS50240

- Retrieve an alignment of Swiss-Prot true positive hits:
  Clustal format, color, condensed view / Clustal format, color / Clustal format, plain text / Fasta format
- Taxonomic tree view of all Swiss-Prot/TrEMBL entries matching PS50240
- Retrieve a list of all Swiss-Prot/TrEMBL entries matching PS50240
- Scan Swiss-Prot/TrEMBL entries against PS50240
- view ligand binding statistics

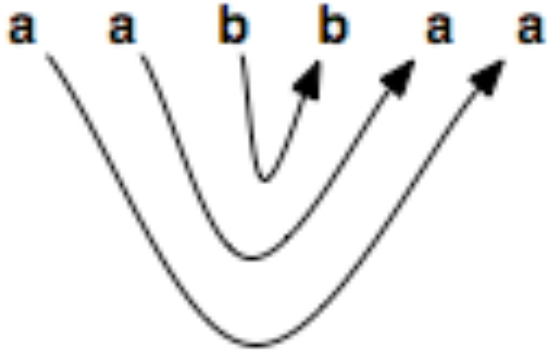
Matching PDB structures: 1A0H 1A0J 1A0L 1A2C ... [ALL]

TRYPSIN_HIS, PS00134; **Serine proteases, trypsin family, histidine active site** (PATTERN)

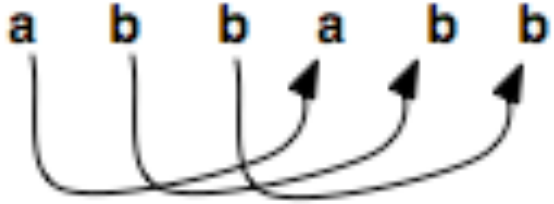
Consensus pattern:	**[LIVM] - [ST] - A - [STAG] - H - C** *H is the active site residue*
Sequences known to belong to this class detected by the pattern:	ALL, except for complement components C1r and C1s, pig plasminogen, bovine protein C, rodent urokinase, a two insect trypsins
Other sequence(s) detected in Swiss-Prot:	18.

Limits of restricted grammars

Position independent (regular grammar)



Palindrome (context free)



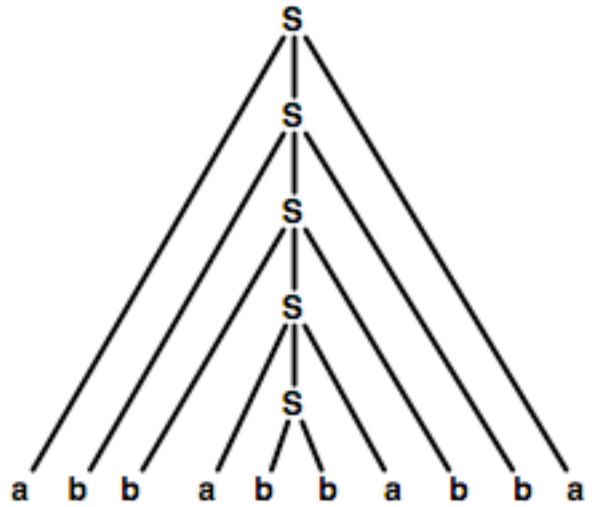
Copy (context sensitive)

Context free grammars

Can address palindromes and some coupling of possitions.

S->aSa|bSb|aa|bb

abaaba
abbabbabba=
(a(b(b(a(bb)a)b)b)a)=

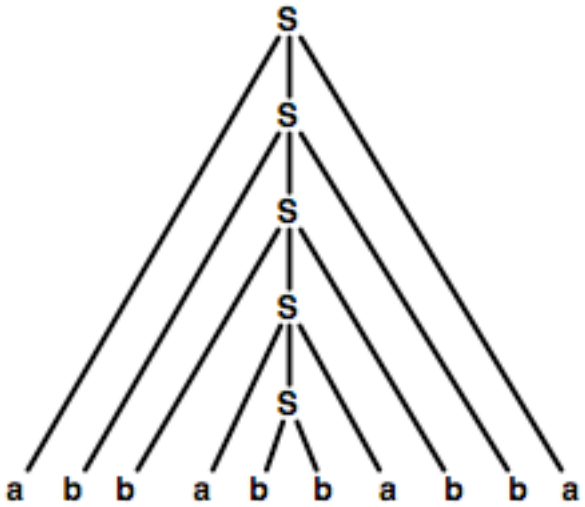


Context free grammars

Can address palindromes and some coupling of possitions.

S->aSa|bSb|aa|bb

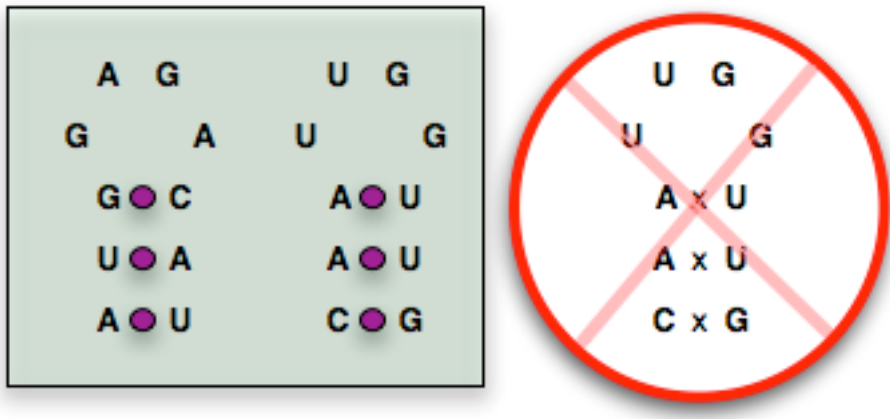
abaaba
abbabbabba=
(a(b(b(a(bb)a)b)b)a)=



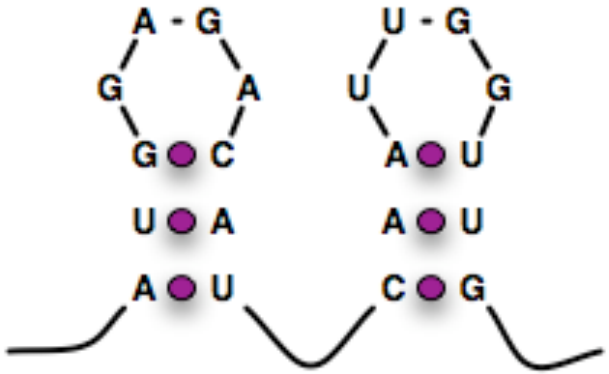
A context free grammar for simple RNA stem-loops

S-> $aW_1u \mid uW_1a \mid gW_1c \mid cW_1g$
W_1-> $aW_2u \mid uW_2a \mid gW_2c \mid cW_2g$
W_1-> $aW_3u \mid uW_3a \mid gW_3c \mid cW_3g$
W_3-> nnnn

Where nnnn represents any 4-mer



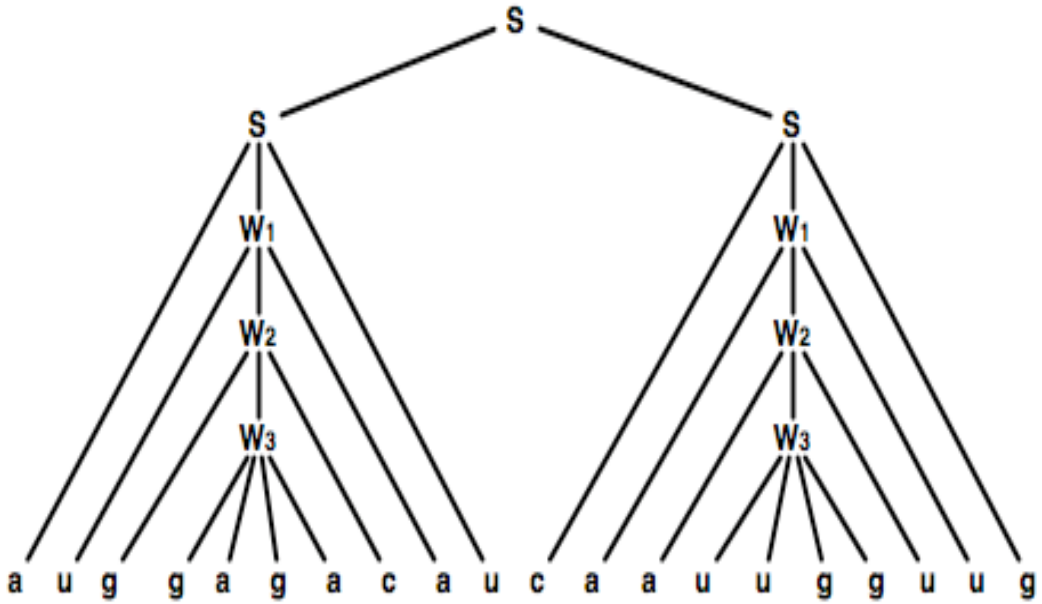
Context free grammars

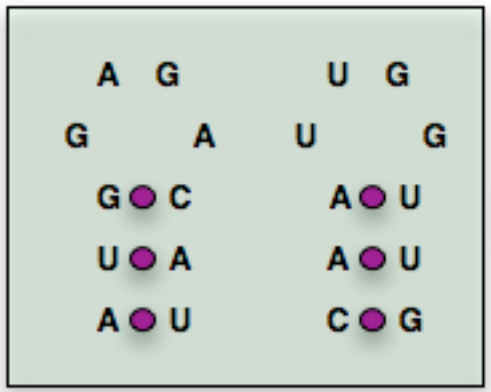


A context free grammar for simple RNA stem-loops

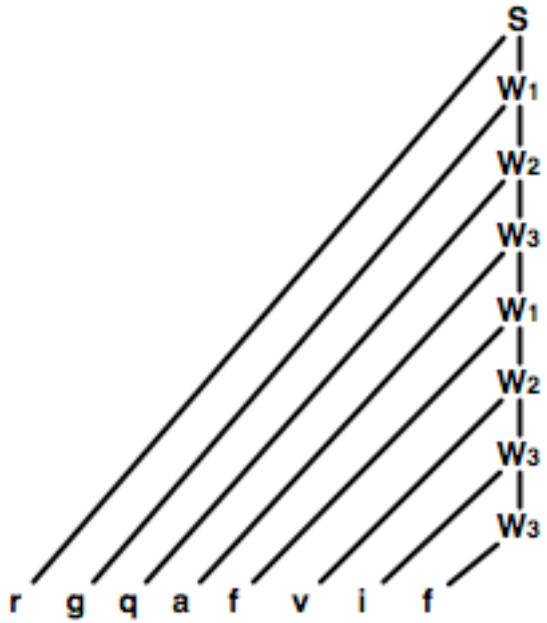
S-> $aW_1u \mid uW_1a \mid gW_1c \mid cW_1g$
W_1-> $aW_2u \mid uW_2a \mid gW_2c \mid cW_2g$
W_1-> $aW_3u \mid uW_3a \mid gW_3c \mid cW_3g$
W_3-> nnnn

Where nnnn represents any 4-mer





Parse trees for regular grammars are less exciting



We can also represent any alignment of a sequence to a prosite pattern (a regular grammar) as a parse tree

The sequence RGQAFVIF matching / parsed by / aligned to
The RNP-1 motif:
[RK]-G-{EDRKHPCG}-[AGSCI]-[FY]-[LIVA]-x-[FYM]

Where "{}" indicate NOT-match

Context free grammars- Push Down automaton

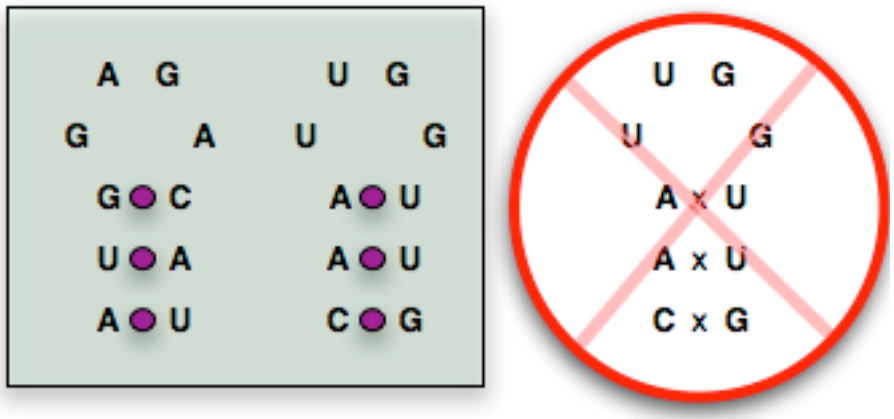
Given a sequence how do we test to see if there is a valid parse of the sequence given the rewrite rules

Does : GCCGCAAGGC fit?

A context free grammar for simple RNA stem-loops

S-> $aW_1u \mid uW_1a \mid gW_1c \mid cW_1g$
W_1-> $aW_2u \mid uW_2a \mid gW_2c \mid cW_2g$
W_1-> $aW_3u \mid uW_3a \mid gW_3c \mid cW_3g$
W_3-> nnnn

Where nnnn represents any 4-mer



Push Down automaton

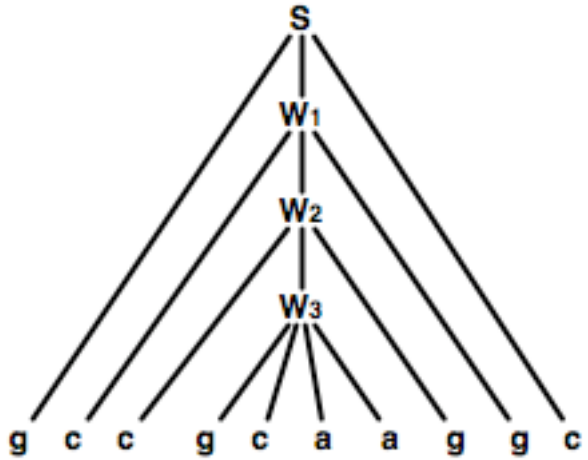
A context free grammar for simple RNA stem-loops

$S \rightarrow aW_1u \mid uW_1a \mid gW_1c \mid cW_1g$
$W_1 \rightarrow aW_2u \mid uW_2a \mid gW_2c \mid cW_2g$
$W_2 \rightarrow aW_3u \mid uW_3a \mid gW_3c \mid cW_3g$
$W_3 \rightarrow nnnn$

Where nnnn represents any 4-mer

Input:	Stack:	Operation:
GCCGCAAGGC	S	Pop S, push g1c
GCCGCAAGGC	g1c	pop g, accept g, R on input
G**C**CGCAAGGC	1c	pop 1, push c2g
G**C**CGCAAGGC	c2gc	pop c, accept c, R on input
GC**C**GCAAGGC	2gc	pop 2, push c3g
GC**C**GCAAGGC	c3ggc	pop c, accept c, R on input
GCC**G**CAAGGC	3ggc	pop 3, push nnnn
GCC**G**CAAGGC	nnnnggc	pop g, accept g, R on input
…	…	…several acceptances…
GCCGCAAGGC	c	
GCCGCAAGGC	empty	accept whole string

Push Down automaton



A context free grammar for simple RNA stem-loops

S-> $aW_1u \mid uW_1a \mid gW_1c \mid cW_1g$
W_1-> $aW_2u \mid uW_2a \mid gW_2c \mid cW_2g$
W_2-> $aW_3u \mid uW_3a \mid gW_3c \mid cW_3g$
W_3-> nnnn (in n is popped from stack accept a,c,g OR t on input at current position)

Input:	Stack:	Operation:
GCCGCAAGGC	S	Pop S, push g1c
GCCGCAAGGC	g1c	pop g, accept g, R on input
G**C**CGCAAGGC	1c	pop 1, push c2g
G**C**CGCAAGGC	c2gc	pop c, accept c, R on input
GC**C**GCAAGGC	2gc	pop 2, push c3g
GC**C**GCAAGGC	c3ggc	pop c, accept c, R on input
GCC**G**CAAGGC	3ggc	pop 3, push nnnn
GCC**G**CAAGGC	nnnnggc	pop g, accept g, R on input
…	…	…several acceptances…
GCCGCAAGGC	c	
GCCGCAAGGC	empty	accept whole string

Stochastic Grammars

Turn profile into regular grammar:

Seq:	1	2	3	… L
A	.2	.1	1.0	
C	.2	.1	0	
G	.6	.1	0	
T	.0	.7	0	

W_1-> aW_2 | cW_2 | gW_2 | tW_2

P : 0.2 0.2 0.6 0.0

BUT:

1. *how do we learn these things from data !????*
2. *How do we align to SCFG and SRG ?????*

A context free stochastic grammar

W_1-> aW_2u | uW_2a | gW_2c | cW_2g

P : 0.2 0.2 0.1 0.5

HMMs are stochastic regular grammars (why? … 0th order)

Can you show that any order Markov chain can be written as a stochastic regular grammar?

Inside algorithm

First we rewrite our SCFG in Chomsky normal form:

Two allowable rule types:

$W_v \rightarrow W_y W_z$

$W_v \rightarrow a$

Eg.

$W \rightarrow aWa \Rightarrow$

$W_0 \rightarrow W_{left} W_{right}$

$W_{left} \rightarrow a$

$W_{right} \rightarrow W_0 W_{left}$

Inside algorithm

$$\mathbf{W} = W_1, W_2, ..., W_M$$

$$init : i = 1...L, v = 1...M$$

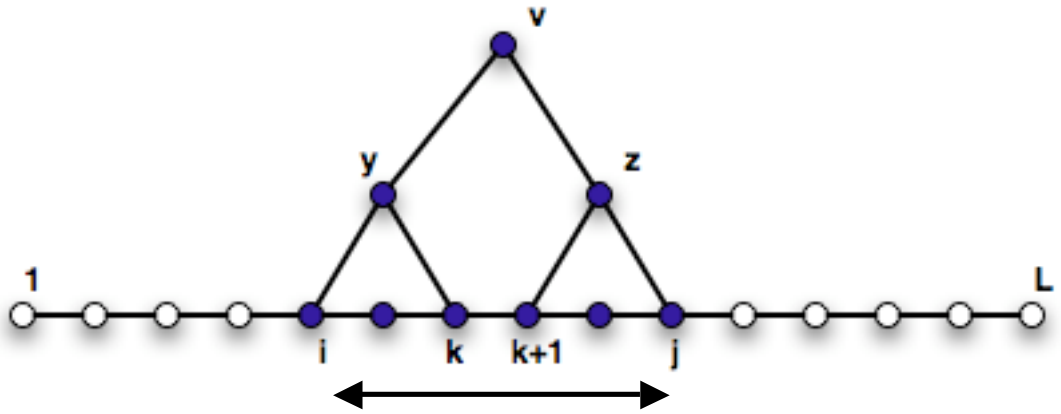
$$\alpha(i,i,v) = e_v(x_i)$$

$$iteration : i = 1...L-1, j = i+1...L, v = 1...M$$

$$\alpha(i,j,v) = \sum_{y=1}^{M} \sum_{z=1}^{M} \sum_{k=i}^{j-1} \alpha(i,k,y)\alpha(k+1,j,z)t_v(y,z)$$

$$Term :$$

$$P(x \mid \theta) = \alpha(1,L,1)$$



Parameter estimation

We estimate expected number of times each state (c(v) for W_v) is used and the number of times Wv -> WyWz for all v and v->y,z

This is an EM procedure similar to the EM procedure used to parameterize HMMs (Baum-Welch)

1. We estimate c(v) and c(v->y,z)
2. We use estimates to calc $t_v(y,z)$ and $e_v(a)$ [update θ]
3. Inside algorithm
4. Outside algorithm
5. -Repeat #1 if $P(x|\theta)$ has improved
   -Converge if $P(x|\theta)$ not changed by threshold

Estimating model parameters with results from inside and outside algorithm.

$$c(v) = \frac{1}{P(x|\theta)} \sum_{i=1}^{L} \sum_{j=1}^{L} \alpha(i,j,v)\beta(i,j,v)$$

$$c(v \to y,z) = \frac{1}{P(x|\theta)} \sum_{i=1}^{L-1} \sum_{j=i+1}^{L} \sum_{k=i}^{j-1} \beta(i,j,v)\alpha(i,k,y)\alpha(k+1,j,z)t_v(y,z)$$

SCFG -> HMM

GOAL	HMM	SCFG
Opt alignment	Viterbi	CYK (max)
$P(x\|\theta)$	Forward	Inside (sum)
Parameters	Forward-backward	Inside-Outside
memory	$O(LM)$	$O(L^2M)$
time	$O(LM^2)$	$O(L^3M^3)$

Next week's reading

- Ch. 10 BSA : RNA structure prediction…last BSA chapter :-(
- Sabbi Lall. Current Biology 16, 460-471.
-