Interpolated Markov Models for Gene Finding

BMI/CS 776

www.biostat.wisc.edu/bmi776/

Spring 2018

Anthony Gitter

gitter@biostat.wisc.edu

Goals for Lecture

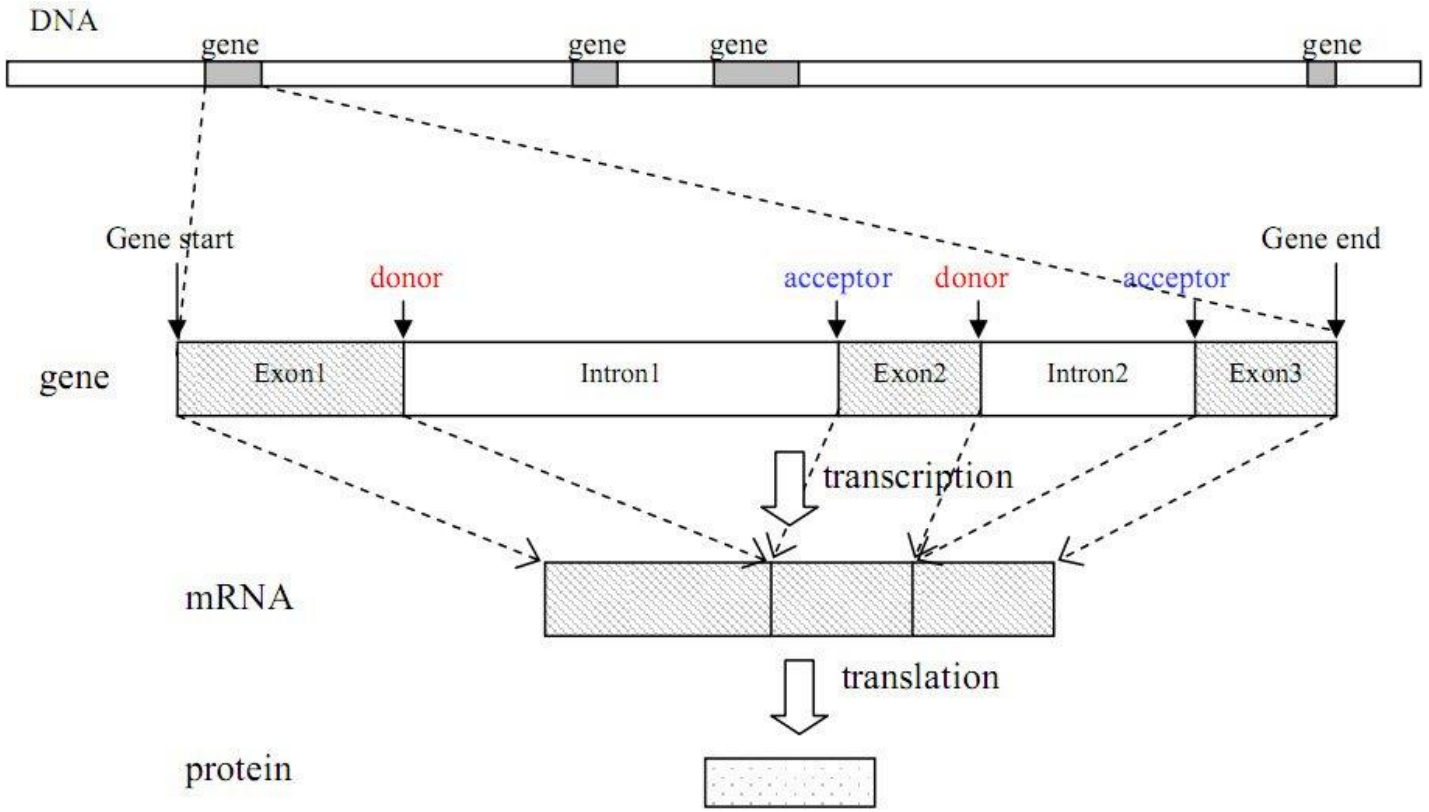
Key concepts

- the gene-finding task
- the trade-off between potential predictive value and parameter uncertainty in choosing the order of a Markov model
- interpolated Markov models

The Gene Finding Task

Given: an uncharacterized DNA sequence

Do: locate the genes in the sequence, including the coordinates of individual *exons* and *introns*



Sources of Evidence for Gene Finding

- **Signals**: the sequence *signals* (e.g. splice junctions) involved in gene expression
- **Content**: statistical properties that distinguish protein-coding DNA from non-coding DNA
- **Conservation**: signal and content properties that are conserved across related sequences (e.g. orthologous regions of the mouse and human genome)

Gene Finding: Search by Content

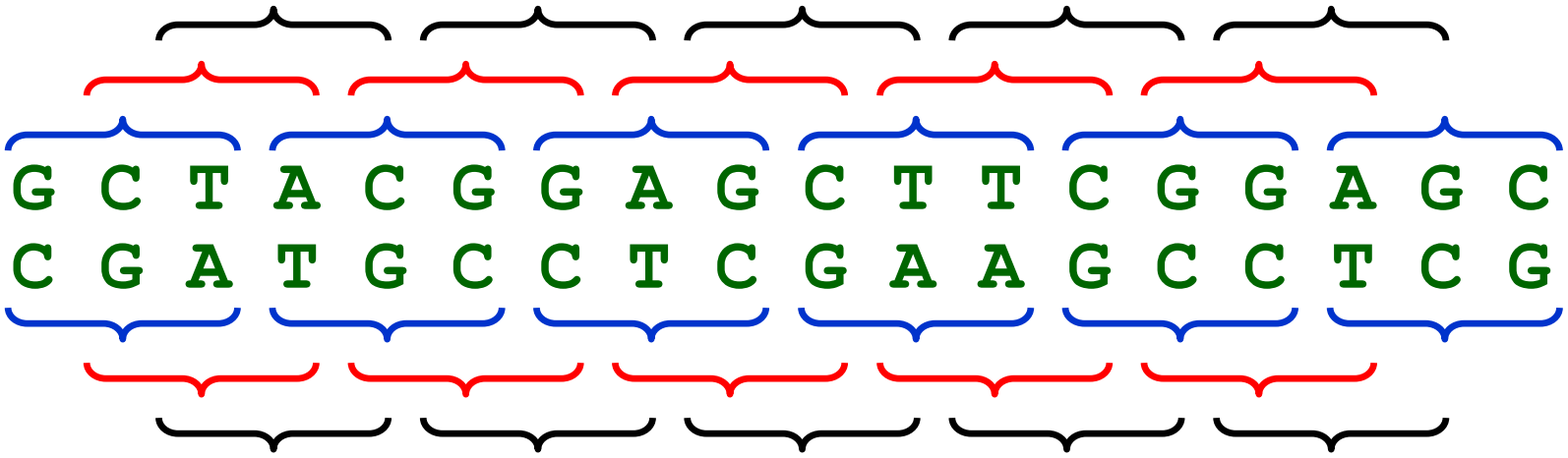
- Encoding a protein affects the statistical properties of a DNA sequence
  - some amino acids are used more frequently than others (Leu more prevalent than Trp)
  - different numbers of codons for different amino acids (Leu has 6, Trp has 1)
  - for a given amino acid, usually one codon is used more frequently than others
    - this is termed *codon preference*
    - these preferences vary by species

Codon Preference in E. Coli

AA	codon	/1000
Gly	GGG	1.89
Gly	GGA	0.44
Gly	GGU	52.99
Gly	GGC	34.55
Glu	GAG	15.68
Glu	GAA	57.20
Asp	GAU	21.63
Asp	GAC	43.26

Reading Frames

- A given sequence may encode a protein in any of the six reading frames



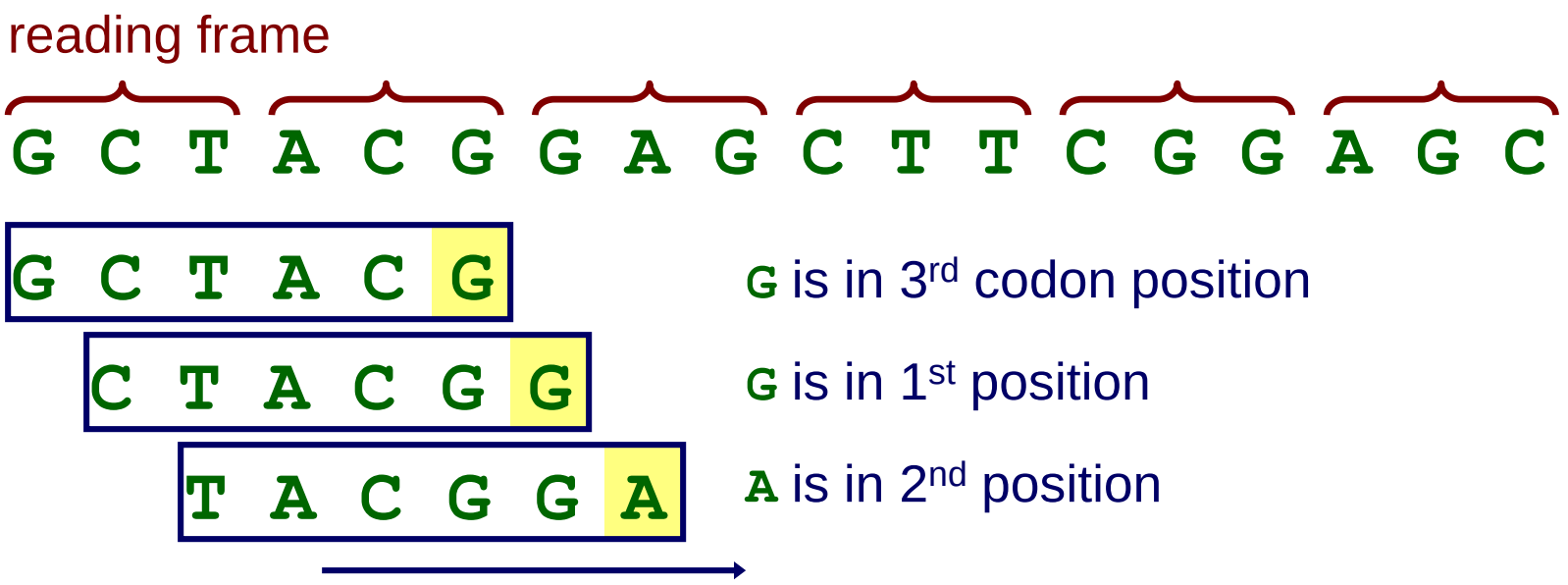
Open Reading Frames (ORFs)

- An ORF is a sequence that
  - starts with a potential start codon
  - ends with a potential stop codon, *in the same reading frame*
  - doesn't contain another stop codon in-frame
  - and is sufficiently long (say > 100 bases)

- An ORF meets the minimal requirements to be a protein-coding gene in an organism without introns

Markov Models & Reading Frames

- Consider modeling a given coding sequence
- For each "word" we evaluate, we'll want to consider its position with respect to the reading frame we're assuming



- Can do this using an inhomogeneous model

Inhomogeneous Markov Model

- **Homogenous Markov model**: transition probability matrix does not change over time or position

- **Inhomogenous Markov model**: transition probability matrix depends on the time or position

Higher Order Markov Models

- Higher order models remember more “history”
- Additional history can have predictive value
- Example:
  - predict the next word in this sentence fragment
    “…you__” (are, give, passed, say, see, too, …?)
  - now predict it given more history

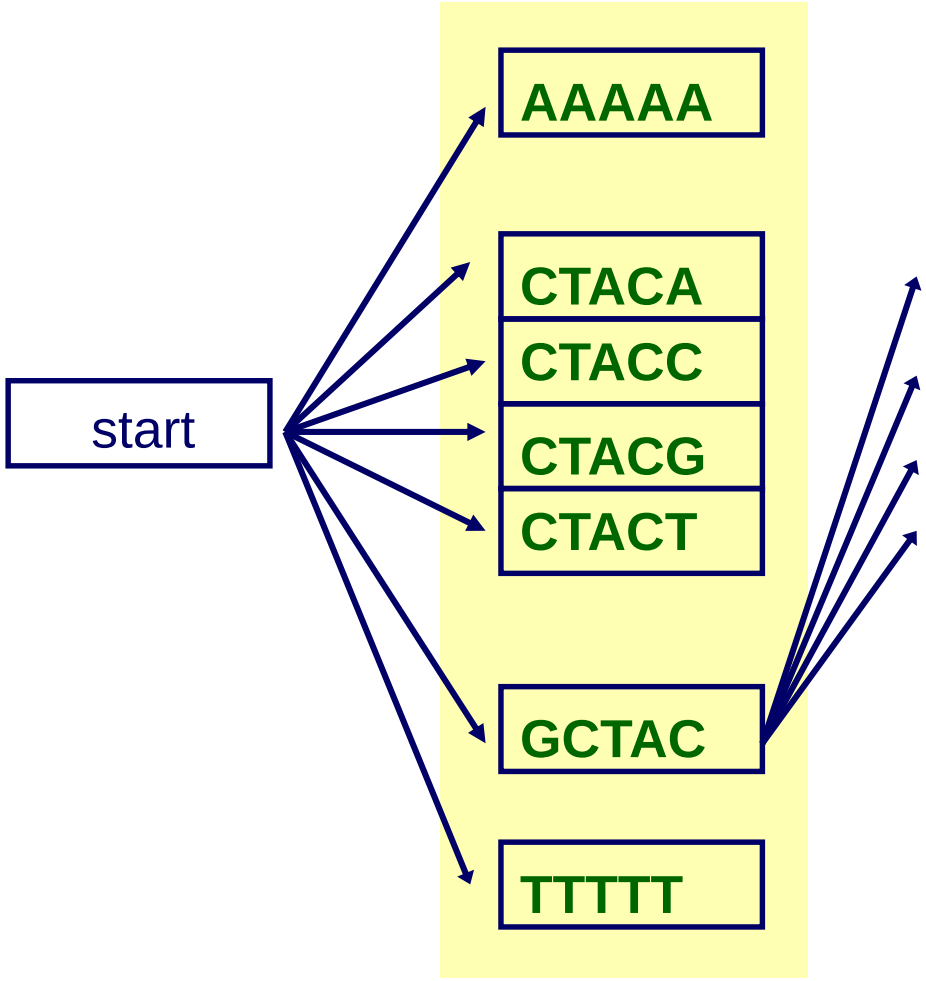
    “…can you___”

    “…say can you___”

    “…oh say can you___”

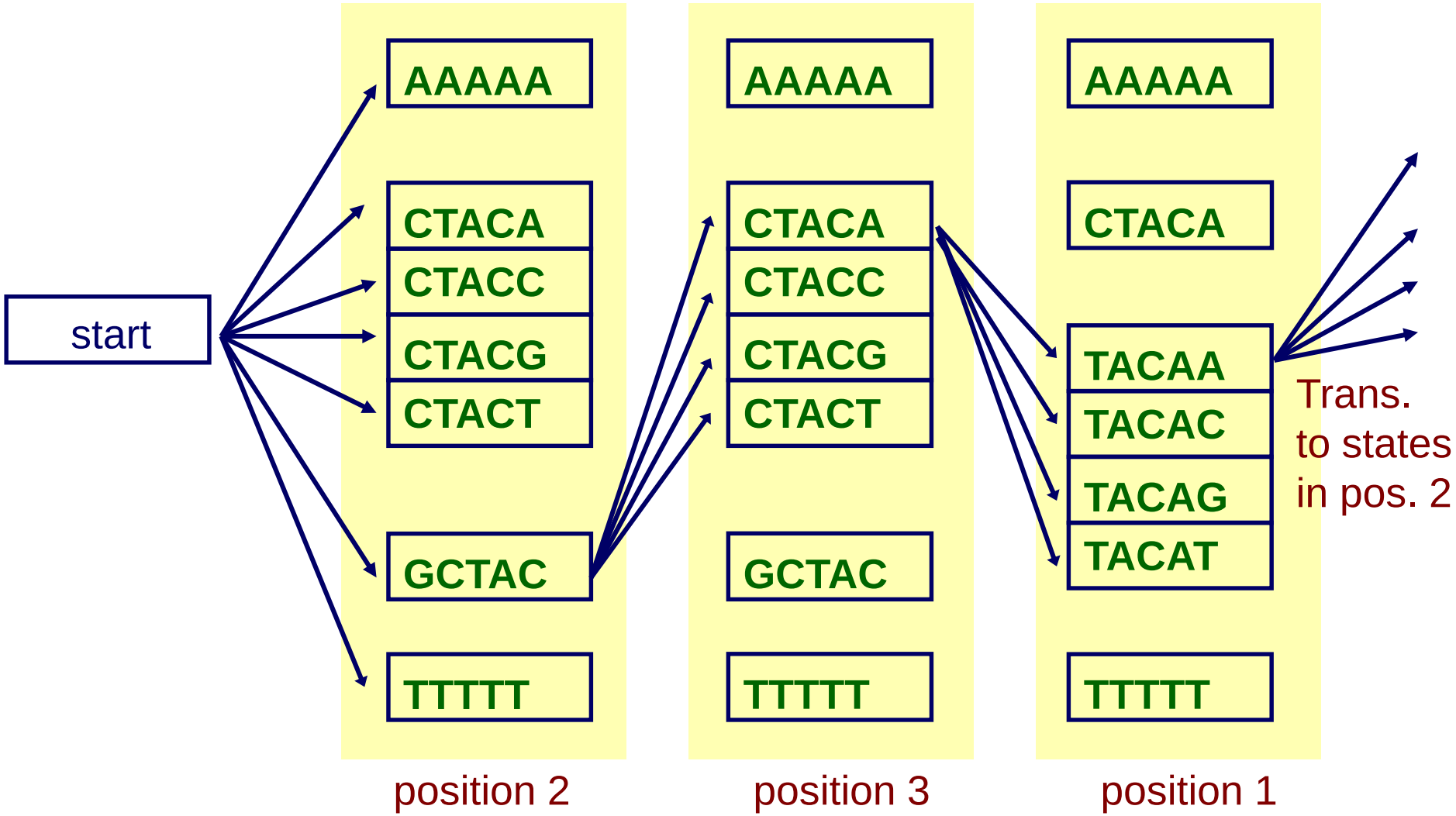
YouTube

A Fifth Order Inhomogeneous Markov Model



$$P(x_i \mid x_{i-5}, \ldots, x_{i-1}, position)$$

A Fifth Order Inhomogeneous Markov Model



Selecting the Order of a Markov Model

- But the number of parameters we need to estimate grows exponentially with the order
  - for modeling DNA we need $O(4^{n+1})$ parameters for an *n*th order model

- The higher the order, the less reliable we can expect our parameter estimates to be
- Suppose we have 100k bases of sequence to estimate parameters of a model
  - for a 2nd order homogeneous Markov chain, we'd see each history 6250 times on average
  - for an 8th order chain, we'd see each history ~ 1.5 times on average

Interpolated Markov Models

- The IMM idea: manage this trade-off by interpolating among models of various orders
- *Simple* linear interpolation:

$$P_{\text{IMM}}(x_i \mid x_{i-n},\ldots,x_{i-1}) = \lambda_0 P(x_i) + \lambda_1 P(x_i \mid x_{i-1}) + \ldots + \lambda_n P(x_i \mid x_{i-n},\ldots,x_{i-1})$$

- where $\sum_i \lambda_i = 1$

Interpolated Markov Models

- We can make the weights depend on the history
  - for a given order, we may have significantly more data to estimate some words than others
- *General* linear interpolation

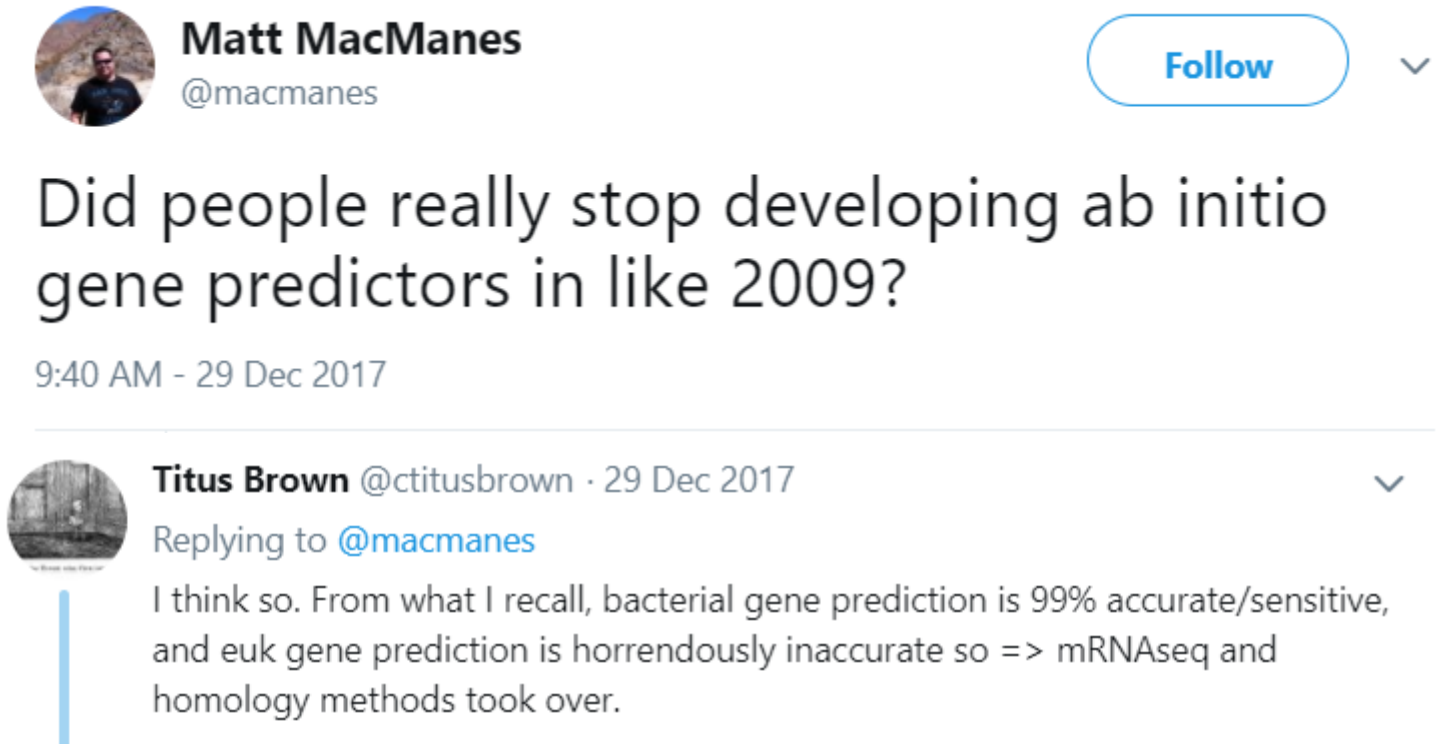
$$P_{\text{IMM}}(x_i \mid x_{i-n},\ldots,x_{i-1}) = \lambda_0 P(x_i)$$
$$+ \lambda_1(x_{i-1}) P(x_i \mid x_{i-1})$$
$$\ldots$$
$$+ \lambda_n(x_{i-n},\ldots,x_{i-1}) P(x_i \mid x_{i-n},\ldots,x_{i-1})$$

λ is a function of the given history

The GLIMMER System

[Salzberg et al., Nucleic Acids Research, 1998]

- System for identifying genes in bacterial genomes
- Uses 8th order, inhomogeneous, interpolated Markov models



IMMs in GLIMMER

- How does GLIMMER determine the λ values?
- First, let's express the IMM probability calculation recursively

$$P_{\text{IMM,n}}(x_i \mid x_{i-n},\ldots,x_{i-1}) =$$
$$\lambda_{\text{n}}(x_{i-n},\ldots,x_{i-1})P(x_i \mid x_{i-n},\ldots,x_{i-1}) +$$
$$[1-\lambda_n(x_{i-n},\ldots,x_{i-1})]P_{\text{IMM,n-1}}(x_i \mid x_{i-n+1},\ldots,x_{i-1})$$

- Let $c(x_{i-n},\ldots,x_{i-1})$ be the number of times we see the history $x_{i-n},\ldots,x_{i-1}$ in our training set

$$\lambda_n(x_{i-n},\ldots,x_{i-1}) = 1 \quad \text{if} \quad c(x_{i-n},\ldots,x_{i-1}) > 400$$

IMMs in GLIMMER

- If we haven't seen $x_{i-n},\ldots,x_{i-1}$ more than 400 times, then compare the counts for the following:

*n*th order history + base	(*n-1*)th order history + base
$x_{i-n},\ldots,x_{i-1},a$	$x_{i-n+1},\ldots,x_{i-1},a$
$x_{i-n},\ldots,x_{i-1},c$	$x_{i-n+1},\ldots,x_{i-1},c$
$x_{i-n},\ldots,x_{i-1},g$	$x_{i-n+1},\ldots,x_{i-1},g$
$x_{i-n},\ldots,x_{i-1},t$	$x_{i-n+1},\ldots,x_{i-1},t$

- Use a statistical test to assess whether the distributions of x_i depend on the order

IMMs in GLIMMER

*n*th order history + base	(*n-1*)th order history + base
$x_{i-n},\ldots,x_{i-1},a$	$x_{i-n+1},\ldots,x_{i-1},a$
$x_{i-n},\ldots,x_{i-1},c$	$x_{i-n+1},\ldots,x_{i-1},c$
$x_{i-n},\ldots,x_{i-1},g$	$x_{i-n+1},\ldots,x_{i-1},g$
$x_{i-n},\ldots,x_{i-1},t$	$x_{i-n+1},\ldots,x_{i-1},t$

- Null hypothesis in χ^2 test: x_i distribution is independent of order
- Define $d = 1 - pvalue$
- If d is small we don't need the higher order history

IMMs in GLIMMER

- Putting it all together

$$\lambda_n(x_{i-n},\ldots,x_{i-1}) = \begin{cases} 1 & \text{if } c(x_{i-n},\ldots,x_{i-1}) > 400 \\ d \times \dfrac{c(x_{i-n},\ldots,x_{i-1})}{400} & \text{else if } d \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

where $d \in (0,1)$

IMM Example

- Suppose we have the following counts from our training set

ACGA	25	**CGA**	100	**GA**	175
ACGC	40	**CGC**	90	**GC**	140
ACGG	15	**CGG**	35	**GG**	65
ACGT	20	**CGT**	75	**GT**	120
	100		300		500

χ^2 test: $d = 0.857$ χ^2 test: $d = 0.140$

$\lambda_3(\mathbf{ACG}) = 0.857 \times 100/400 = 0.214$

$\lambda_2(\mathbf{CG}) = 0$ (d < 0.5, $c(\mathbf{CG}) < 400$)

$\lambda_1(\mathbf{G}) = 1$ ($c(\mathbf{G}) > 400$)

IMM Example (Continued)

- Now suppose we want to calculate $P_{\text{IMM},3}(T \mid ACG)$

$$P_{\text{IMM},1}(T \mid G) = \lambda_1(G)P(T \mid G) + \big(1-\lambda_1(G)\big)P_{\text{IMM},0}(T)$$
$$= P(T \mid G)$$

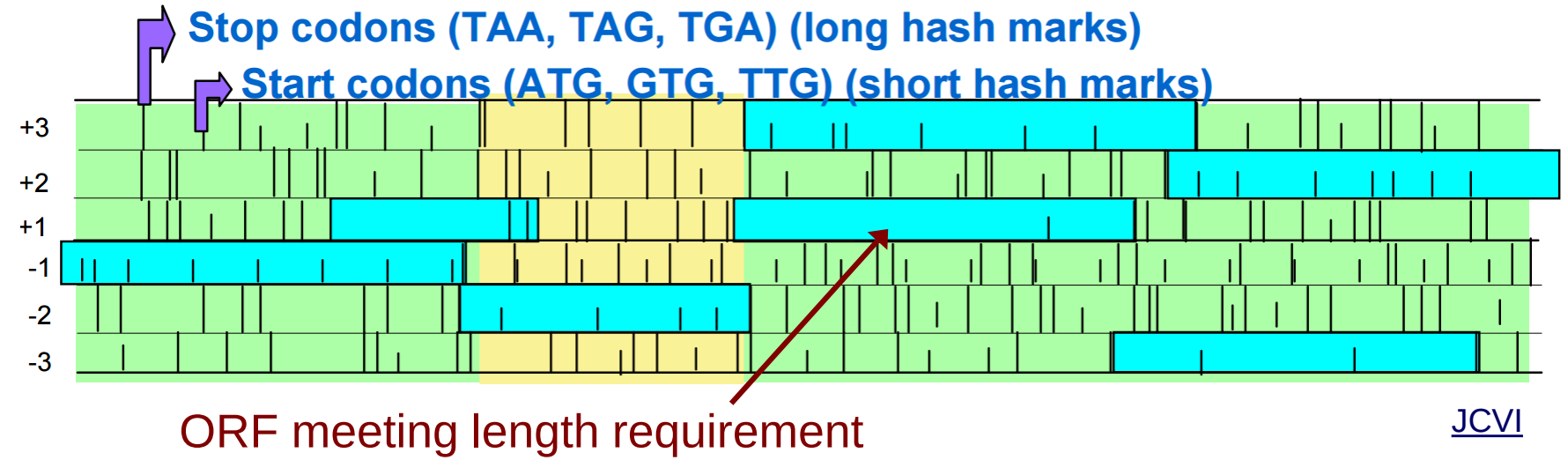
$$P_{\text{IMM},2}(T \mid CG) = \lambda_2(CG)P(T \mid CG) + \big(1-\lambda_2(CG)\big)P_{\text{IMM},1}(T \mid G)$$
$$= P(T \mid G)$$

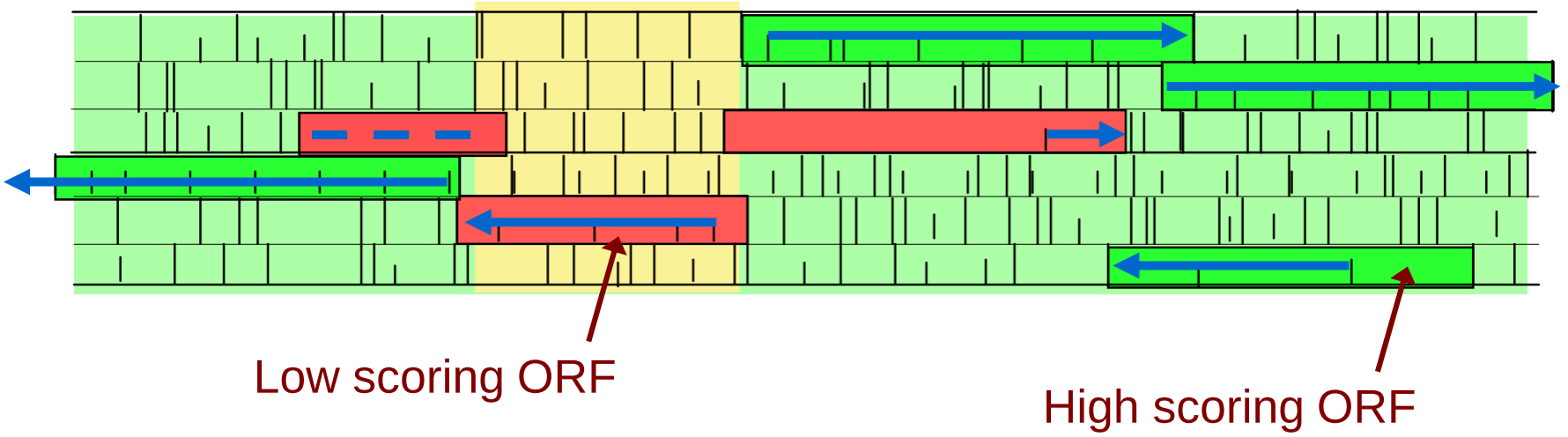
$$P_{\text{IMM},3}(T \mid ACG) = \lambda_3(ACG)P(T \mid ACG) + \big(1-\lambda_3(ACG)\big)P_{\text{IMM},2}(T \mid CG)$$
$$= 0.214 \times P(T \mid ACG) + (1-0.214) \times P(T \mid G)$$
$$= 0.214 \times 0.2 + (1-0.214) \times 0.24$$

Gene Recognition in GLIMMER

- Essentially ORF classification
- For each ORF
  - calculate the probability of the ORF sequence in each of the 6 possible reading frames
  - if the highest scoring frame corresponds to the reading frame of the ORF, mark the ORF as a gene
- For overlapping ORFs that look like genes
  - score overlapping region separately
  - predict only one of the ORFs as a gene

Gene Recognition in GLIMMER





GLIMMER Experiment

- 8^{th} order IMM vs. 5^{th} order Markov model
- Trained on 1168 genes (ORFs really)
- Tested on 1717 annotated (more or less known) genes

GLIMMER Results

	TP	FN	FP & TP?
Model	Genes found	Genes missed	Additional genes
GLIMMER IMM	1680 (97.8%)	37	209
5th-Order Markov	1574 (91.7%)	143	104

The first column indicates how many of the 1717 annotated genes in *H.influenzae* were found by each algorithm. The 'additional genes' column shows how many extra genes, not included in the 1717 annotated entries, were called genes by each method.

- GLIMMER has greater sensitivity than the baseline
- It's not clear whether its precision/specificity is better