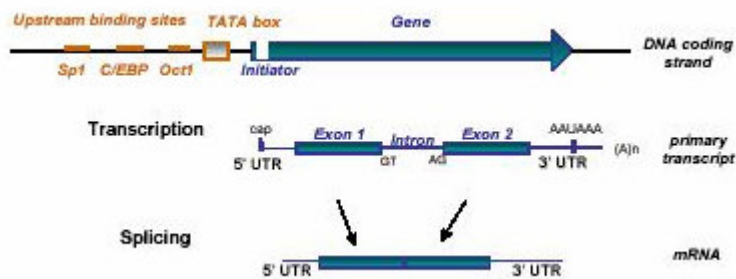


Gene Prediction Review

(1) Biological background

Eukaryotic gene structure

- 1) recognition region (\approx 50 kb)
- 2) transcription initiation site
- 3) 5' UTR untranslated region
- 4) translation initiation
- 5) alternating exon/intron, splice donor and acceptor sites
- 6) translation stop site
- 7) 3' UTR
- 8) polyadenylation signal
- 9) transcription stop site



Prokaryotic gene structure

- 1) recognition region (\approx 50 bp)
- 2) transcription initiation site
- 3) 5' untranslated region
- 4) translation initiation
- 5) coding region
- 6) translation stop site
- 7) 3' untranslated region
- 8) translation stop site

(2) Prokaryotic gene prediction

Geometric distribution

- (1) Consider a process where codons are chosen at random among the 64 possibilities
- (2) Success is defined as choosing a stop codon (this has probability $3/64$)
- (3) Random variable L = number of trials until first success (i.e. $L \cdot 3$ is the length of the sequence)
- (4) A value of $L=u$ where u is much higher than the expected value would indicate that the sequence is likely to be a gene

Note: $P(L=u) = (1-p)^{u-1}p$

(3) Categories of eukaryotic gene prediction

Ab initio gene predictors

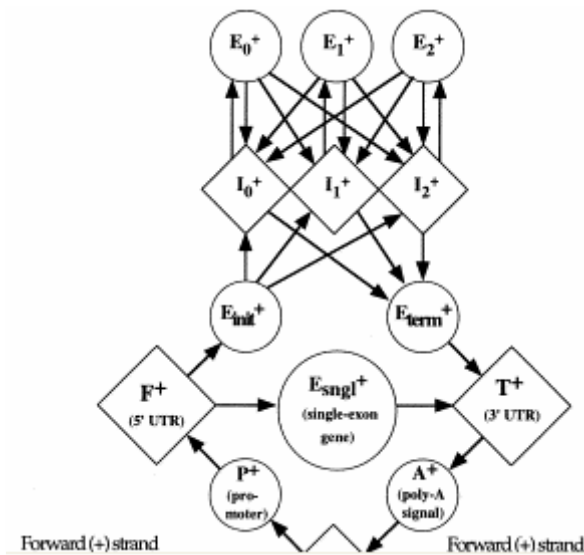
HMMGene, **Genscan**, GlimmerHMM, Genie

Homology based gene predictors

Genomescan, **TWINSKAN**, Shadower

(4) Ab initio gene prediction

Genscan: semi-markov duration hidden markov model (i.e. generalized HMM)



The GHMM given above generates a parse (ϕ) consisting of an ordered set of states, $s = \{s_1, \dots, s_n\}$ with a corresponding set of lengths (durations), $d = \{d_1, \dots, d_n\}$. The generated sequence has final length $L = \sum d_i$.

- (1) An initial state, s_1 , is chosen based on an initial distribution on the states.
- (2) A length, d_1 , is chosen based on the length distribution for state s_1 .
- (3) A sequence segment, w_1 of length d_1 , is generated according to an appropriate sequence generating model for state q_1 .
- (4) The subsequent state, s_2 , is generated based on the transition probabilities from s_1 .
- (5) Steps (2) through (4) are repeated until the sum of the state durations equals or exceeds the desired length L .

Optimal Parse

The parse that optimizes (i.e. maximizes) the following probability is the optimal parse:

$$P(\phi_I | \text{seq}) = P(\phi_I, \text{seq}) / P(\text{seq})$$

In other words, the optimal parse is: $\max_{\phi_I} P(\phi_I, \text{seq})$, where $P(\phi_I, \text{seq})$ equals:

$$\begin{aligned} P(\phi_I, S) = & P(q_1 \text{ is initial state}) * \\ & P(\text{duration of } q_1 \text{ is } d_1) * \\ & P(S(1..d_1) \text{ is generated by state } q_1) * \\ & \text{for } i \text{ from } 2 \text{ through } n \\ & T(q_{i-1}, q_i) * \\ & P(\text{duration of } q_i \text{ is } d_i) * \\ & P(S(L(i-1)..L(i-1)+d_i-1) \text{ is generated by state } q_i) \end{aligned}$$

$$P\{\phi_I, S\} = \pi_{q_1} \cdot f_{q_1}(d_1) \cdot P\{s_1 | q_1, d_1\} \cdot \prod_{k=2}^n T_{q_{k-1}, q_k} \cdot f_{q_k}(d_k) \cdot P\{s_k | q_k, d_k\}$$

Computing the optimal parse in this fashion is just an extension of the viterbi algorithm on non-duration HMMs.

Partition function

The partition function is the fancy name for the probability of a given sequence.

Mathematically, this is written as:

$$P(\text{seq}) = \sum_{\phi_I} P(\phi_I, \text{seq})$$

This probability is computed directly by the forward algorithm (see Durbin et al, chapter 3 for a compact review).

Exon score

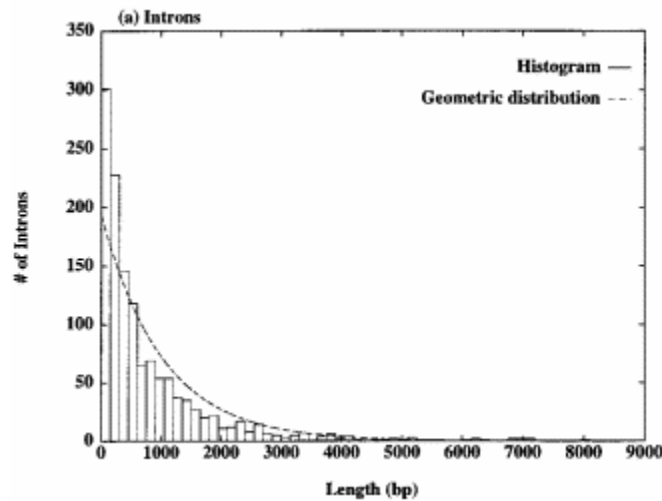
The exon score is a fancy name for the posterior probability of an exonic state being a part of a parse given the input sequence. This probability can be computed directly by the forward-backward algorithm (see Durbin et al, chapter 3 for a compact review).

Training the HMM

The GHMM can be trained by learning transition and state duration distributions based on training data. For example:

- (1) Initial state probabilities are learned from the training data by determining how often arbitrary positions are part of an intron, part of an exon, part of a 3' UTR, etc.
- (2) Transition probabilities are learned from the training data by determining how often a gene is a single-exon gene, the average number of introns per gene, etc.

- (3) State length (i.e. state duration) distributions are set by computing length distributions for various states from the training data. For example, consider the following graph from Burge et al:



The dotted line is the geometric distribution (exponential curve), indicating that GHMM states that represent introns can have their duration length distributions modeled by geometric distributions.

(5) Homology based gene prediction

TWINSKAN: cross-species genomic sequence comparisons

Use the Human-Mouse genome alignment

```

HUMAN:  AACCGCTCGGGACGAGG  (S)
MOUSE:  ACGG TTGGGA CGAG
ALIGN:  -||:|-|:||||-:::|  (C)

```

Consider the probability of generating both S and C

$$P(S, C|Exon) = P(S|Exon) \cdot P(C|Exon)$$

TWINSKAN, as well as ROSETTA and Shadower, take advantage of cross-species comparative genomics. However, other types of homology based predictions exist.

- (1) Spliced alignment techniques for aligning putative EST regions to the genomic DNA (engine: EST_GENOME)
- (2) Updating/correcting the probabilities of GHMM predicted parses by comparing the parses with BLASTX output (engine: GENOMESCAN)