The Korean Journal of Microbiology, Vol. 39, No. 4, December 2003, p. 207-215

Duration HMM을 이용한 진핵생물 유전자 예측 프로그램 개발

태홍석 · 박기정*

(주)스몰소프트 정보기술연구소

주어진 염기서열에서 단백질로 코딩되는 영역을 예측하는 유전자 구조 예측은 유전자 annotation의 가장 핵심적인 부분으로 유전자 분석 및 유전체 프로젝트 전체에 큰 영향을 준다. 진핵생물의 유전자가 원핵생물의 유전자에 비해 더 복잡한 구조를 가지기 때문에 진핵생물의 유전자 구조 예측 모델 역시 원핵생물에 비해 다양하고 복잡한 모델로 구성되어 있다. 본 연구팀은 duration hidden markov model을 기본형태로 하여 진핵생물의 유전자 구조 예측 프로그램인 EGSP를 개발하였다. 이 프로그램은 각 생명체의 유전자 구조 예측에 필요한 파라메터를 생성하는 학습기능과, 이를 기반으로 핵산 서열을 입력으로 해서 단백질을 코딩하는 부위를 예측하여 출력하는 기능으로 구성되며, 최근의 프로그램들의 추세대로 복수 개 유전자 예측의 기능을 갖추고 있다. EGSP의 학습과 예측에 사용되는 각 파라메터의 전체 성능에 대한 효과 분석 등을 위해 여러 개 signal에 대한 개별 모델이 주는 효과 등을 분석하였다. 진핵생물의 유전자 구조 예측에 가장 많이 연구되는 human dataset을 이용하여 현재 개발된 유전자 구조 예측 프로그램인 GenScan과 GeneID, Morgan 등 보편적으로 사용되는 프로그램들과의 성능을 여러 가지 기준에서 비교한 결과, 본 프로그램이 실용성 있는 수준을 보여주는 것을 확인하였다. 그리고 진핵 미생물인 *Saccharomyces cerevisiae*로 성능을 테스트한 결과 만족할 만한 수준의 성능을 나타내는 것을 알 수 있었다.

Key words □ duration HMM, EGSP, gene prediction, GenScan, training

유전체의 전반적인 구조와 기능을 밝히고자 하는 유전체 프로젝트가 시작된 이후 생명체의 유전체에 대한 많은 연구가 진행되고 있으며 그 결과가 데이터베이스로 저장되고 있다. 유전체 프로젝트의 첫 번째 단계라고 할 수 있는 유전체 염기서열 분석의 비율이 증가하면서 유전체 내의 정확한 유전자 위치를 알아내기 위해 많은 유전자 구조 예측 모델들이 개발되었다. 생물체의 유전체에 존재하는 유전자의 위치를 정확하게 밝혀내는 것은 유전자간의 연관성, 그 유전자로부터 얻어지는 단백질간의 연관성, 그리고 나아가서는 비슷한 유전자들을 가지는 생물종간의 연관성을 밝히는 전체 과정에서 가장 핵심적인 단계로서 매우 중요한 의미를 가진다.

1980년대 초에 Shepherd(19), Fickett(7), 그리고 Staden과 McLachlan(24)에 의한 유전자 구조 예측의 초기연구에서는 아미노산 분포와 codon usage의 경향을 통계적으로 측정해서 genome sequence에 존재하는 단백질의 coding region을 예측하는 방법을 사용했다. 그 후 k-tuple frequencies(4), autocorrelation(13), Fourier spectra(20), purine/pyrimidine periodicity(1), 그리고 local compositional complexity/ entropy(11)등 coding region과 non-coding region에서의 차별적인 구성을 가지는 특성들이 알려지면서, 이러한 구성들의 차이를 이용하여 유전체에 존재하는 coding region의 위치를 예측하는 방법들이 개발되었고, 이와 더불어 진핵생물의 복잡한 유전자 구조를 예측하는 프로그램들이 등장하기 시작했다. 그 중에서 Fickett의 모델에 근거한 TestCode와, neural network 접근방식으로 여러 가지 구성에 대한 통계적 수치를 적용해 염기서열 단편을 coding region과 non-coding region으로 구분한 GRAIL (25)이 가장 널리 사용되었다.

DNA 한쪽 가닥만을 분석하도록 만들어졌던 기존 모델의 문제점을 극복하기 위해, GenMark(2)는 DNA 두 가닥을 동시에 분석하여, 한쪽 가닥의 coding region에 의해서 다른 가닥에서도 그 위치에서 non-coding region임에도 불구하고 coding region처럼 인식되는 'shadow' coding region 문제를 해결하고자 하였다. GenMark는 non-coding region에서는 homogeneous 5th-markov chain, coding region에서는 codon의 위치특이적인 non-homogeneous 5th-markov chain을 DNA 양쪽 가닥에 모두 구성하고, 각 markov chain의 상대적인 score에 따라 coding region을 찾아낸다. GenMark 이후 원핵생물 유전체에 대한 유전자 구조 예측 프로그램으로는 Glimmer(5, 18)가 가장 널리 사용되고 있다. Coding 및 non-coding region에서의 6-tuple의 출현빈도를 측정해서 coding region을 찾는 GenMark와는 달리 Glimmer에서는 interpolated Markov model을 사용하여 8-tuple 또는 그 이하의 길이를 가지는 oligomer의 출현빈도를 측정해서 유전자 구조 예측에 이용하였다.

진핵생물의 유전자는 원핵생물의 유전자보다 구조가 더 복잡하고 유전체 크기에 비해 유전자의 밀도가 원핵생물보다 훨씬 떨어진다. 원핵생물의 유전자 구조가 promoter, start codon, coding region, stop codon, non-coding region등으로 이루어진데 비해 진핵생물의 유전자는 cap, polyA와 같이 전사에 관련된

*To whom correspondence should be addressed.
Tel: 82-42-864-2524, Fax: 82-42-866-9241
E-mail: kjpark@smallsoft.co.kr

signal이 더 존재하며, coding region도 donor, acceptor signal에 의해 exon과 intron으로 나누어진다.

진핵생물에 대한 유전자 구조 예측은 *Caenorhabditis elegans*의 유전자 구조 예측을 위해 gm을 개발한 Fields(8)와 포유동물의 유전자 구조 예측을 연구한 Gelfand(9)에 의해서 시작되었는데, 이들은 입력 염기서열로부터 initial exon과 terminal exon을 포함하는 완성된 구조의 유전자를 예측하는 모델을 구성하였다.

이 후 hierarchical rule을 이용하여 exon의 가능성이 있는 단편에 대해 순위를 계산하는 모델을 사용한 GeneID(10), neural network과 dynamic programming을 혼합한 GeneParser(21, 22), linguistic method를 사용한 GenLang(6), discriminant analysis를 사용한 FGENEH(23), decision tree를 사용한 Morgan(17), generalized hidden markov model을 사용한 Genie(12), 그리고 duration HMM (hidden markov model)을 사용한 GenScan(3)등이 개발되었다. HMM이 응용된 모델은 기계학습 방법 중에서도 비교적 유전자 구조 예측에 높은 효율성을 보여주며, 이와 함께 유전자를 구성하는 5'UTRs, 3'UTRs, exons, introns등의 segment를 state로 두고 주어진 DNA 염기서열을 이들 signal을 포함한 유전자 구조로 해석하는 복잡한 모델이 사용되어, 보다 정교한 예측을 하기 위한 방법들이 개발되고 있다.

HMM은 조건부 확률에 근거해서 바로 이전 상태로부터 현재 상태를 추측해 내는 markov chain의 확장된 모델로서, 음성 인식 분야에 주로 사용되다가 기본적인 통계적 방법으로 활용되고 있다. HMM은 state들과 symbol들을 기본 요소로 가지고 있으며 관찰되는 symbol들의 출현빈도를 측정해서 조건 확률에 따라 실제에 가장 근사한 state들의 구성을 추측해 내는 모델이다. 1990년 초반부터 motif domain 검색이나 promoter prediction등과 같은 생물학을 위한 연구에서 HMM이 적용되기 시작했다 특히, DNA 염기서열과 같이 연속적이고 반복되는 형태를 가진 구조에서 HMM을 구성하기가 용이하다.

HMM이 효율적인 통계적 모델임에도 불구하고 같은 state가 여러 시간동안 지속될 경우 state들의 지속 횟수에 따른 확률분포를 표현하지 못한다. Duration HMM은 symbol들이 연속해서 같은 state를 지속한 횟수를 표현하기 위한 HMM의 응용된 모델이다(16). 즉, state에 대해 특정 시간제한을 둘 수 있어 그 제한 내에서 다른 state로의 transition 없이 동일한 state에 반복적으로 머무를 수 있도록 할 수 있다. Exon이나 intron과 같은 유전자의 각 구성요소들은 일정한 길이 범위에서 분포하기 때문에 길이의 분포확률은 유전자의 구조를 표현하는데 중요한 정보라고 할 수 있다. 본 연구에서는 DNA를 구성하는 각 segment의 길이 분포를 잘 반영할 수 있도록 이 길이 정보를 반영하여 duration HMM을 구성하였다.

본 연구팀은 GenScan에서 사용된 duration HMM과 signal score 계산방식의 기본 모델을 적용하여 진핵생물 유전자 구조 예측 프로그램인 EGSP (Eukaryotic Gene Structure Prediction) 프로그램을 개발하였다. 진핵 유전자 구조 예측에 필요한 각 파라메터 계산을 위한 프로그램들과 이에 기반을 두어 duration HMM으로 유전자 구조를 예측하는 프로그램을 완성하였다.

EGSP의 개발 단계에서, 학습 데이터와 테스트 데이터로는 다른 유전자 구조예측 프로그램들의 개발에 가장 많이 쓰였던 human genome의 염색체 서열을 사용함으로서 다른 프로그램과의 성능 비교를 원활히 하고자 하였고, 진핵 미생물인 *Saccharomyces cerevisiae* 유전체의 서열과 같은 다른 진핵생물에 대해서도 학습 및 예측 프로그램의 효용성을 시험하였다. 이 프로그램의 성능 평가를 위해, 같은 테스트 데이터에서 수행된 GenScan, GeneID 그리고 Morgan의 결과와 상호 비교하였다.

재료 및 방법

EGSP의 전체 구성 및 구조

EGSP는 크게 두 개의 프로그램 그룹으로 구성된다.

첫 번째는 prediction에 사용되는 파라메터를 학습하기 위한 프로그램들이며, 두 번째는 prediction 프로그램들이다. Prediction을 위해 사용하는 파라메터는, GenScan을 비롯한 다수의 알고리즘 분석에 근거하여 설계하였으며, 현재 다음의 파라메터들을 사용하고 있다.

(1) Signal 검색을 위한 Signal model의 matrix
- Start, stop, donor, acceptor, polyA, promoter

(2) 각 state의 initiation probability

(3) 각 state의 length distribution

(4) State transition probability

(5) 각 state 의 segment score 계산
- Homogeneous 5^{th}-markov model : noncoding region
- Nonhomogeneous 5^{th}-markov model : coding region

학습과 prediction을 위한 파라메터 개발을 위한 기본 모델 구축과 시험을 위해 human의 유전체 데이타를 사용하였으므로, 재료와 방법에서는 human 데이타를 대상으로 설명한다. 한편, 실제 미생물에서도 이 프로그램들을 활용하는 예로서, *S. cerevisiae*에 대한 학습 및 prediction을 구축 및 시험하였다.

EGSP의 학습 프로그램

학습 프로그램들을 학습 dataset으로부터 이들 각 파라메터를 학습하는 프로그램들로 구성되는데, 세부 구조는 Fig. 1과 같다. 이 프로그램은 prediction에 사용되는 각 파라메터에 대해 개별 학습 데이터를 생성하는 기능들을 가지고 있는데, 각 파라메터를 학습하기 위한 데이터는 다음과 같이 다양한 소스로부터 구하였다. 이 모델의 구현에 필요한 파라메터를 학습시키기 위해서는 많은 종류의 학습데이터가 필요하다. 길이의 분포와 state들의 개시확률, intergenic region의 homogeneous markov matrix를 학습시키기 위해서 human chromosome 21번을 학습데이터로 사용하였고, homogeneous markov matrix 및 nonhomogeneous markov matrix를 위해 GenBank의 Ref. database human data(15)와 David Kulp에 의해 1992년에 GenBank release 89로부터 만들어진 dataset을 사용하였다. Promoter는 학습데이터로는 EPD (Eukaryotic Promoter Database)(14)의 human promoter를 사용하였고, promoter를 제외한 5종류의 signal은 Kulp의 dataset을 이

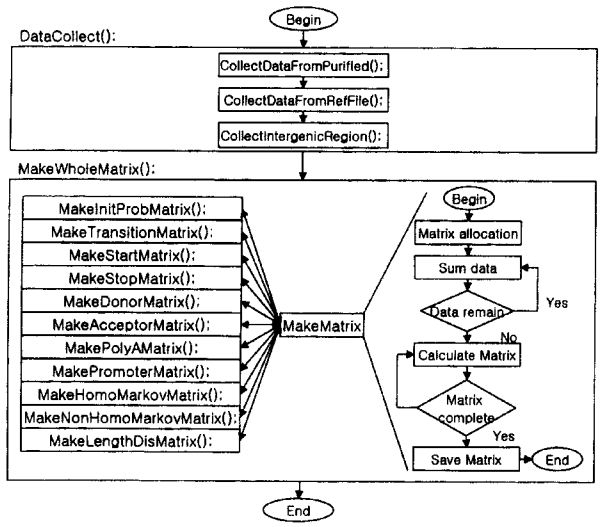

Fig. 1. The detailed structure of EGSP training algorithm.

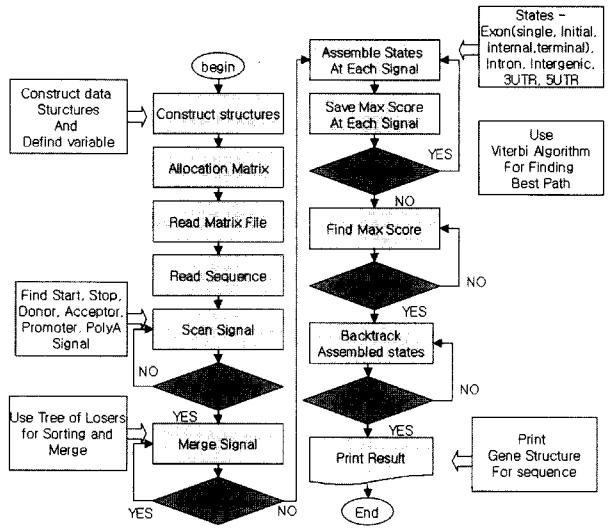

Fig. 2. The detailed structure of EGSP prediction algorithm.

용해서 학습시켰다. 학습프로그램을 구성하는 세부 모듈들의 기능은 Table 1과 같다.

EGSP의 prediction 프로그램

기존의 진핵생물 유전자 구조 분석 알고리즘에 대한 분석에 근거하여, 가장 진보한 알고리즘인 GenScan 알고리즘의 장점을 최대한 반영하여 자체적인 알고리즘을 구성하였다. Fig. 2에서 각 signal에 대한 scan 후에 coding region과 non-coding region에 대한 인식 및 이의 조합을 위해 duration HMM을 사용한 부분이 전체 알고리즘에서 결과에 가장 핵심적인 부분이다. EGSP

Table 1. Moduels and their functions of EGSP training program

DataCollect()	Function : to parse raw data files before traning and to store in *struct* data structures	
	CollectDataFromPurified()	To collect data from GenScan training files Data : start codon, stop codon, donor signal, acceptor signal, poly A, initial exon, internal exon, terminal exon, intron, 5'utr, 3'utr
	CollectDataFromRefFile()	To collect data from human genome mRNA files of GenBank Ref. Data : exon, 5'utr, 3'utr
	CollectIntergenicRegion()	To collect data from human chr. 21 Data : Intergenic region, initiation probability
MakeWholeMatrix()	Function : To train parameters with collected data To store parameters in files	
	MakeInitProbMatrix()	To calculate initiation probabilities with human chr. 21
	MakeTransitionMatrix()	To calculate state transition probabilities
	MakeStartMatrix()	To construct WMM of preceding 6 bases and following 3 bases of start codon
	MakeStopMatrix()	To construct WMM of preceding 3 bases and following 4 bases of stop codon
	MakeDonorMatrix()	To construct WMM of preceding 3 bases and following 4 bases of GT in domor
	MakeAcceptorMatrix()	To construct WMM of preceding 20 bases AG in acceptor
	MakePolyAMatrix()	To construct WMM of 6 bases of Poly A
	MakePromoterMatrix()	To construct WMM of 40 bases of EPD human promoter
	MakeHomoMarkovMatrix()	To construct matrices of intergenic, intron, 5'utr, 3'utr Homogeneous 5th-markov matrix Length distribution matrix
	MakeNonHomoMarkovMatrix()	To construct nonhomogeneous 5th-markov matrix of exon
	MakeLengthDisMatrix()	To construct length distribution matrix for initial exon, internal exon, terminal exon, single exon

prediction 프로그램의 각 세부 단계는 다음과 같다.

(1) Intergenic regoin, promoter, 5'UTR, 3'UTR, single exon, initial exon, terminal exon, internal exon (phase 0, 1, 2), intron (phase 0, 1, 2)을 각각 HMM의 state로 정하고 각 state에 대한 class를 구현한다.

(2) State의 범위를 결정할 수 있는 signal을 정하고 그 구조체를 형성한다. Start codon, stop codon, donor, acceptor, promoter, poly A.

(3) ReadMatrix() 함수에서 State initiation probability, state transition probability, state length distribution probability, signal matrix, segment matrix를 matrix file로 부터 읽는다.

(4) ReadSequence() 함수에서 Gene structure를 prediction할 DNA sequence를 읽는다.

(5) ScanSignal() 함수에서 Signal matrix를 기초로 sequence로부터 모든 signal의 가능성이 있는 부분을 검색한다.

(6) SignalSort() 함수에서 검색된 signal을 위치에 따라 정렬한다.

(7) Dynamic programing을 이용해서 signal 위치에서 가능한 이전 state의 조합을 구성하고, 각 state의 조합에서 가장 score가 가장 높은 조합만을 선택한다.

(8) 모든 signal에서 state의 구성이 종료되면 가장 score가 높은 조합을 Gene structure로 저장한다.

Duration HMM의 구현

본 논문에서 구현한 duration HMM은 intergenic region과 forward/reverse strand 별로 각각 promoter, polyA, 5'UTR, 3'UTR, single exon, initial exon, terminal exon, internal exon(phase 0, 1, 2), intron(phase 0, 1, 2), 27개의 hidden state들과, A, T, G, C, 4개의 symbol로 구성 되었다(Fig. 3).

각 state들 간의 상태전이는 Fig. 3의 화살표의 진행방향으로 일어날 수 있다. 그리고 각 state들 간의 상태전이는 start, stop, donor, acceptor, promoter, polyA, 이렇게 6종류의 signal이 존재하는 위치에서만 가능하다.

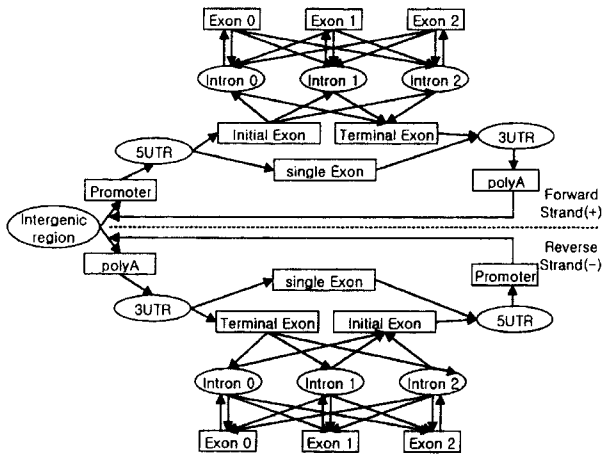

Fig. 3. Duration HMM of EGSP eukaryotic gene structure prediction model. Rectangles and circles represent HMM states and arrows represent state transitions. Every state except 'intergenic' state belongs to a forward strand state group or a reverse strand state group.

Signal 검색

Signal은 그 전사관련 signal, 번역관련 signal, splicing signal 이렇게 세 종류를 사용하였다. 전사관련 signal은 promoter와 polyA를 사용하였다. Cap site와 TATA box 부위를 포함해서 promoter를 검색하기 위해 40 base 길이의 2nd-WAM (Weight Array Matrix)을 이용하였다. 2nd-WAM은 위치(i-2)의 염기구성 bi-2, 위치 (i-1)의 염기 구성 bi-1에 의해서 i번째의 염기의 조건부 확률 Pi = (bi | bi-1, bi-2)를 구한다. EPD에서 검색한 Human promoter 서열 -39~0 부위를 학습데이터로 이용해서 promoter 검색에 필요한 matrix를 구성하였다. PolyA (polyadenylation)은 진핵생물의 pre-mRNA 3' 말단에 나타나는 구조로서 일반적으로 AATAAA hexamer의 consensus를 보인다. PolyA를 검색하기위해 GenBank에서 "polyA_signal"로 annotation 된 염기서열들을 학습데이터로 이용해서 6 base들에 대한 WMM (Weight Matrix Model)을 구성하였다.

번역관련 signal로 start signal과 stop signal을 사용하였다. Start signal의 검색을 위해서는 start codon 이전의 6 base와 이후의 3 base, stop signal의 검색을 위해서는 stop codon 이전의 3 base와 이후의 6 base에 대한 WMM을 각각 구성하였다.

Splicing signal로 donor와 acceptor를 사용하였다. Donor의 검색을 위해서 donor의 보존서열인 GT 이전의 3 base와 이후의 4 base에 대해 WMM을 구성하였다. Acceptor signal 검색을 위해 AG 상류 20 base까지에 대해 WAM을 구성하였다.

State의 길이분포

State의 개수가 L개이고 시간의 길이가 N인 duration HMM에서 Viterbi 알고리즘을 이용해 최적화 된 경로를 찾을 경우 $O(L_2N_3)$의 time complexity를 가진다. 길이가 긴 유전체의 검색에 이러한 time complexity는 적당하지 못하다. 하지만 각 state들의 최소길이와 최대길이에 대해 제한하게 되면 길이가 길어짐에 따라 계산시간이 기하급수적으로 늘어나는 것을 방지할 수 있다.

유전자 구조 예측에서 유전체를 구성하는 state들의 길이 분포는 중요한 정보 중 하나이다. 예를 들어 학습데이터에 의하면 single exon의 최소 길이는 9 base이고 최대 길이는 7400 base이고 9 base 에서 400 base 사이에서는 Gauss distribution에 가까운 분포를 보이는 반면 intron의 경우 exponential distribution을 보인다. EGSP에서는 이러한 분포를 함수로 표현하지 않고 실질적인 분포에 따라 길이분포확률을 적용하였다. 학습데이터를 일정한 길이 단위에 따라 구간별로 나눈 후, 그 구간에 속하는 학습데이터의 개수를 전체 학습데이터로 나누어서 그 구간에 대한 길이분포확률을 할당하였다(Fig. 4).

최적의 Gene Structure 구성

EGSP는 입력된 matrix들과 염기 서열로부터 6가지의 signal을 각각 검색하게 되고 일정한 score 이상이 되는 염기서열을 각

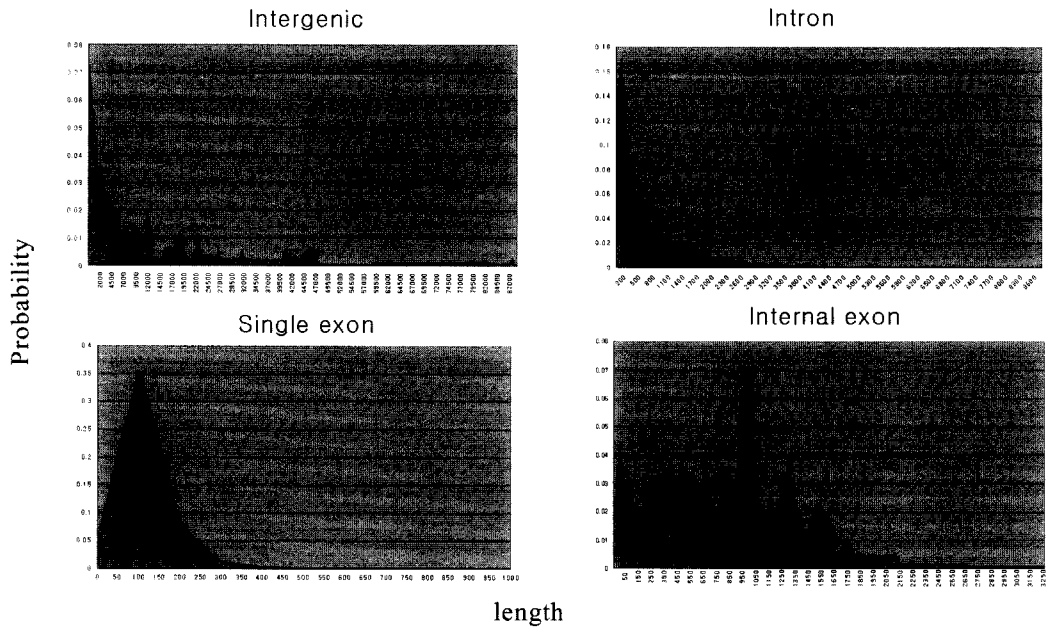

Fig. 4. Length distributions of intergenic region, intron, internal exon and single exon.

signal의 후보로서 저장하게 된다. 각 signal의 후보들이 선택되면 각 signal의 위치에서 가능한 state의 조합을 구성하고 Viterbi 알고리즘을 이용해서 가장 높은 score를 가지는 state의 구성을 찾는다(Fig. 5).

위치 t에서의 state i일 경우의 score는 아래의 (식 1)과 같이 계산한다.

(식 1)

for ($1 \le t \le T$)

$$r(t, i) = MAX\left\{ \pi_i^* \cdot F_i(S_{1,t}) \right\}^* L_i(t),\ MAX(t_{prev}, k)$$

$$\left\{ r(t_{prev}, k)^* F_k(S_{t_{prev}, t})^* L_k(t - t_{prev})^* M(k, i) \right\}$$

$r(t, i)$= t 에서 state i일 경우에 가지는 최대 score.

t = 염기서열 내의 base 위치.

T = 염기서열 전체 길이.

i = t 위치에서의 임의의 state ($1 \le i \le N$).

N = state의 개수.

k = state i 이전에 올 수 있는 state k.

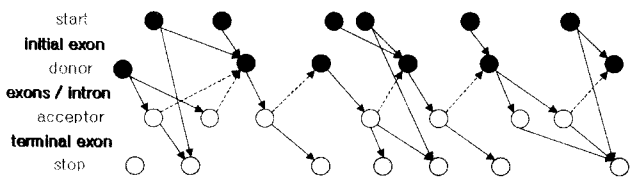

Fig. 5. The dynamic programming algorithm for optimal gene structure evaluation.

$F_i(S_{t_{prev}, t}) = t_{prev}$ 에서 t 까지의 염기서열이 state i에 속할 때 나타나는 확률 값을 반환.

$L_i(t - t_{prev})$ = state i가 $t - t_{prev}$의 길이를 가질 확률 값을 반환.

$M(k, i)$ = state k에서 state i로 transition 될 확률 값을 반환

이 식은 t의 값이 입력 염기서열 전체의 길이인 T에 도달할 때까지 계속되고, T에서 가장 높은 score를 가지는 state i로부터 back-tracking을 통해 최적의 gene structure를 구성한다.

결과 및 고찰

EGSP 프로그램의 사용자 환경과 인터페이스

파라메터를 생성하는 프로그램의 인터페이스는 윈도우즈 프로그램으로 간단히 구성되어 있으며, 내부적으로 각 파라메터 생성 프로그램이 학습을 수행하게 되어, 각 파라메터 파일들을 출력하게 된다. 파라메터 파일이 만들어지면, prediction 프로그램은 이 파일로부터 파라메터를 입력받아 사용하게 된다.

Preidction 프로그램의 인터페이스는 웹을 통해 구현하였다. 사용자로부터 사용될 matrix의 종류와 유전자 구조예측에 사용될 염기서열을 입력받으면, 유전자 구조 예측의 계산결과를 GenScan이나 여타의 gene prediction 프로그램과 같은 형태로 보여준다. 위치정보에 대한 리스트의 출력과 함께 예측된 유전자 각각에 대해 단백질 서열로 변형 했을 때의 서열도 출력한다.

Signal 검색의 모델들

이 모델에서 state들의 상태전이는 signal의 위치에서만 이루어지므로 signal의 정확한 위치를 찾는 것이 무엇보다 중요하다고 할 수 있다. Signal은 그 메카니즘에 따라 전사관련 signal, 번역

관련 signal, splicing signal 이렇게 세 종류로 나눌 수 있다. 그 중에서도 전사관련 signal인 promoter와 splicing signal인 donor, acceptor는 그 위치를 정확히 예측하기 위한 연구가 개별적으로 진행 되고 있을 만큼 생물학적인 중요성이 크다고 할 수 있다.

전사관련 signal인 promoter와 polyA는 유전자의 구조와 intergenic region의 경계가 되는 지점이다. Promoter는 DNA로부터 RNA가 전사되는 정도를 조절하는 중요한 부위로서 다양한 조절인자가 promoter에 영향을 주어서 RNA의 전사를 조절한다. 유전자에 따라서 그 발현을 조절하는 인자가 다르기 때문에 promoter의 구조도 유전자에 따라 다르지만 어느 정도 유사성을 나타내는 부분이 있다. 전사가 시작되는 전사개시부위에 cap site가 존재하고, 전사개시부위의 상류 35-base 부위에 TATA box라는 T, A가 많은 부위가 존재한다. 하지만 TATA box도 현재 밝혀진 전체 promoter의 70% 가량만이 가지고 있기 때문에 이 부위를 이용해 promoter를 검색하는데 많은 어려움이 있다.

번역관련 signal인 start signal과 stop signal은 유전자구조에서 coding 부위와 non-coding 부위를 경계 짓는 signal이다. Start codon은 항상 ATG 3 base로 구성되어 있고, stop codon은 TAG, TGA, TAA 이렇게 세 종류의 codon을 보인다. 두 signal 모두 동일한 염기서열을 보이는 다른 부위와의 구분에 큰 특징을 보이지 않기 때문에 정확한 signal의 검색이 어렵다.

Splicing signal인 donor와 acceptor는 exon과 intron의 경계가 되는 signal이다. Donor와 acceptor는 각각 intron의 5' 말단과 3' 말단에 존재하는 부위로 두 signal의 정확한 위치를 찾기 위해 많은 연구가들이 splicing에 대한 연구를 진행하였다. GT 주변에 존재하는 염기들의 구성이 상호의존적이라는 분석에 근거하여 GenScan에서는 MDD (Maximal Dependence Decomposition)와 같이 염기들의 상호의존성을 반영한 모델을 사용하였다. 이 논문에서는 donor의 검색을 위해서 donor의 보존서열인 GT 이전의 3 base와 이후의 4 base에 대해 WMM을 구성하였다. Acceptor signal은 염기들의 상호의존성이 donor와 같이 강하지 않기 때문에 MDD와 같은 모델이 적당하지 않다. Acceptor는 보존서열 AG와 상류에서 약한 상동성을 보이기 때문에 AG 상류 20 base까지 WMM과 WAM을 구성하였다. EGSP에서는 AG 상류 20 base에 대해 WAM을 구성하고 acceptor signal을 검색하였다.

Signal scan 모델별 성능 비교

Acceptor에 대한 모델로 pyrimidin-rich 지역을 포함하는 보전서열 AG 이전의 20 base에 대해서 WMM과 2nd-WAM을 구현하여 비교하였다.

Fig. 6(a)는 WMM과 2nd-WAM의 sensitivity (true positive signal/annotated signal)를 비교한 그래프이다. 학습데이터의 annotated acceptor에 대해서 두 모델을 각각 적용하여 score를 계산하고 logodds score값이 cutoff value이상 되는 score를 가지는 true positive acceptor의 비율을 계산하였다. 그래프는 cutoff value를 -5~5 까지 변화를 주었을 때 acceptor의 sensitivity 변화를 나타낸다. WAM이 WMM에 비해서 높은 sensitivity를 나타낸다. EGSP에서는 acceptor를 위한 모델로 2nd-WAM을 최종적으로 채택하였다.

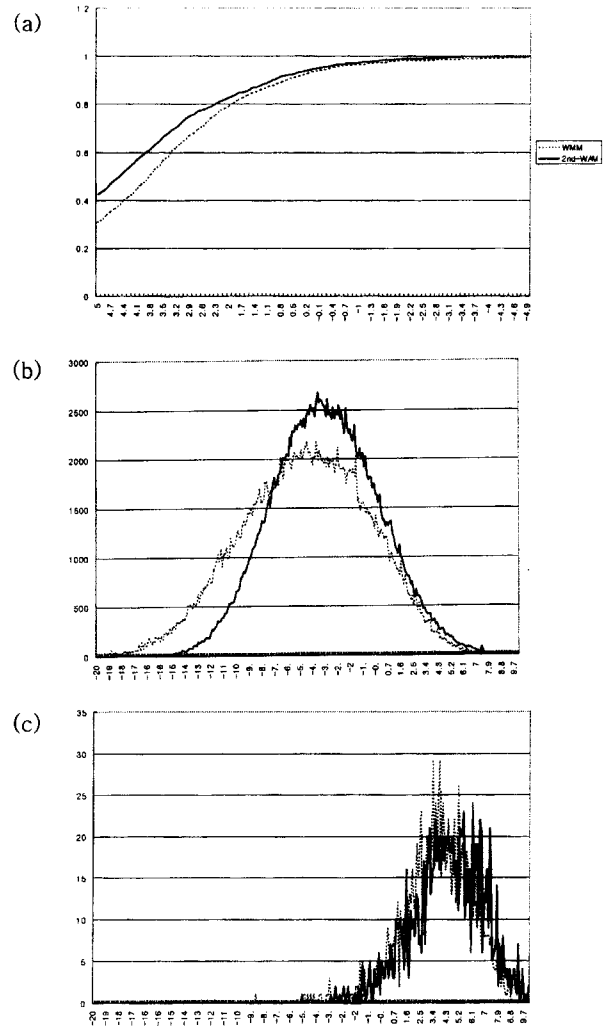

Fig. 6. Sensitivity analysis of models for scanning acceptor signal.

Fig. 6(b)는 학습데이터에 존재하는 모든 AG의 이전 20 base에 대해 WMM과 WAM의 score를 각각 계산한 후 logodds score에 따른 개수를 그래프로 나타낸 것이고, Fig. 6(c)는 실제 acceptor signal에 대한 logodds score별 개수를 나타낸다. 이 그림들에서 알 수 있듯이 WAM이 WMM에 비해서 true acceptor의 검색에 더 유용하다.

Promoter signal은, EPD에 저장되어 있는 human promoter 1867개의 sequence로부터 40(-39~0) base를 추출하여 WMM, 2nd-WAM, 2nd-markov model을 구현하고, 10개의 집합으로 random하게 나누어서 cross validation test를 하였다. Promoter 역시 acceptor과 같은 방법으로 sensitivity를 측정하였다.

Flg. 7 는 model별로 cutoff score의 변화에 따른 sensitivity를 나타낸다. 전반적으로 2nd-WAM이 다른 두 모델에 비해서 비교적 sensitivity가 높게 나타났다. EGSP에서는 promoter를 위한 모델로 2nd-WAM을 최종적으로 채택하였다.

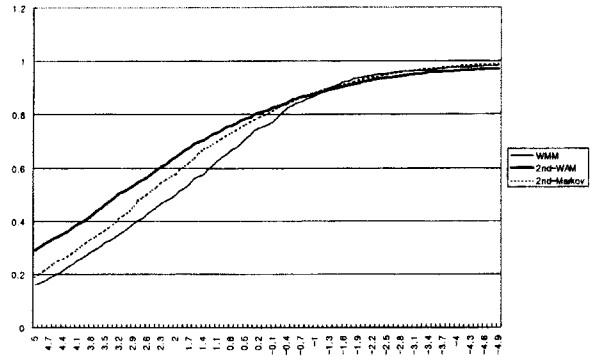

Fig. 7. Sensitivity analysis of models for scanning promoter signal.

EGSP 최종 결과의 성능 비교

Kulp가 Genie(21)의 학습을 위해 GenBank release 95로부터 만든 데이터 셋을 테스트 데이터로 이용하였다. 이 데이터 셋은 http://www.fruitfly.org/seq_tools/datasets/Human/에서 다운로드 받을 수 있다. 이 데이터 셋에서 EGSP의 학습데이터 셋과 중복되는 부분을 제외한 나머지(single exon gene 208개, mutiple exon 210개)를 테스트 데이터로 이용하였다. EGSP와의 상호 비교를 위한 프로그램으로 GenScan, GeneID, 그리고 Morgan을 사용하였다.

Table 2는 네 가지 prediction 프로그램들의 각 signal에 대한 정확성을 비교한다. 각 signal에 대한 정확성은 annotation 된 exon들과 prediction한 exon들의 양 말단을 비교함으로서 구해내었다. Annotation과 prediction의 구분이 명확한 signal인 start, stop, donor, acceptor만을 비교하였다. 이 비교자료는 exon의 종류에 따른 유전자 구조 예측 프로그램들의 정확성을 나타낸다. EGSP를 비롯한 네 개의 프로그램 모두 start와 stop signal의 정확성이 donor와 acceptor signal에 비해서 떨어지는 것으로 나타났다(Table 2). 이는 유전자 구조 예측 프로그램들이 initial exon과 terminal exon을 찾는데 어려움을 가지고 있다는 것을 의미한다. GenScan의 donor가 다른 결과에 비해 정확한 결과를 보이는데 이 결과는 donor를 찾기 위한 모델로 GenScan에서 사용된 MDD가 가장 효과적임을 나타내며, 이러한 모델의 추가를 통해 EGSP가 개선 될 것으로 예상된다.

Table 2. Performance comparison of EGSP prediction program with other gene prediction programs using signal hit analysis

program	signal	TP	FP	FN	Sn	Sp
EGSP	start	159	271	256	0.38	0.36
	stop	92	24	326	0.22	0.79
	donor	697	969	441	0.61	0.41
	accept	680	698	469	0.59	0.49
Morgan	start	172	235	245	0.41	0.42
	stop	143	264	275	0.34	0.35
	donor	761	1188	413	0.64	0.39
	accept	734	1211	377	0.66	0.37
GeneID	start	154	102	259	0.37	0.60
	stop	233	79	185	0.55	0.74
	donor	881	329	244	0.78	0.72
	accept	895	351	269	0.76	0.71
GenScan	start	298	126	116	0.71	0.70
	stop	326	118	92	0.77	0.73
	donor	1014	451	138	0.88	0.69
	accept	1008	455	148	0.87	0.68

Sn = TP/Annotated signals
Sp = TP/Predicted signals
TP = number of true positives

Table 3은 네 프로그램들이 prediction한 exon들을 annotation된 exon들과 비교한 결과를 보여준다. 이 table에서 exact match는 annotation된 exon이 prediction한 exon의 양 말단과 정확히 일치하는 exon을 의미하고, partial match는 한쪽만이 정확히 일치하는 exon을, overlap은 양쪽 모두 일치하지 않지만 서로 겹쳐진 exon을 의미한다. 비교결과 GenScan이 annotated exon중 78%를 찾아내었고, EGSP는 42%를 찾아내었다. EGSP는 GenScan보다는 낮은 정확성을 보이지만, Morgan에 비해서는 높은 정확성을 보였다. 한편 total match로 볼 때 EGSP는 70.4%로 현 단계에서도 실용적으로 사용할 수 있음을 보여준다. Morgan과의 비교에서 total match가 Morgan이 높은 match 비율을 가지지만, 정확성이 떨어지는 결과를 보이고 있음을 볼 수 있고, 이는 Table 2의 signal hit analysis를 통해 명백히 나타나고 있다.

진핵 미생물에서의 EGSP 성능을 평가하기 위해 *S. cerevisiae*

Table 3. Performance comparison of EGSP prediction program with other gene prediction programs using exon hit analysis

program	# of predicted exons	exact match			# of partial match	# of overlap	total match		
		#	Sn(%)	Sp(%)			#	Sn(%)	Sp(%)
EGSP	1760	666	42.6	37.8	316	117	1099	70.4	62.4
Morgan	2354	631	40.4	26.8	508	130	1269	81.3	53.9
Geneid	1512	939	60.1	62.1	280	66	1285	82.4	84.9
GenScan	1893	1214	77.8	64.1	204	21	1439	92.2	76.0

Total number of annotated exons = 1560
Sn = (# of match / # of annotated exons)
Sp = (# of match / # of predicted exons)
In comparison of a predicted exon with the corresponding annotated exon, 'exact match' means that the starting and the ending sites are exactly matched, 'partial match' means that either the starting or the ending sites is exactly matched, and 'overlap' means that neither the starting nor the ending sites is exactly matched but the compared exons are overlapped with each other.

를 사용하여 테스트를 수행하였다. *S. cerevisiae*는 진핵생물 중에서도 하등한 형태를 보이는 미생물로서 비교적 결과에 대한 평가가 원활히 수행될 수 있다. 진핵생물의 표준적인 유전자 구조를 가지는 human에 비해, *S. cerevisiae*의 경우는 원핵생물의 유전자 구조와 매우 유사한 구조를 보이고 있으며, 기존의 annotation된 정보로부터 promoter나 polyA 및 intron 등에 대한 정보를 거의 구할 수 없다. 따라서, 학습과 prediction 프로그램에서는 표준 구조에서 많이 수정된 단순화된 모델을 사용하였고, promoter에 대해서는 다른 유전체에 대한 정보를 활용하였다. Table 4는 *S. cerevisiae* 16개의 chromosome중 7개의 chromosome에 대한 예측 결과로서, 하나의 chromosome을 테스트하기 위해 다른 6개의 chromosome으로 학습하는, cross validation 방법을 통하여 유전자 구조 예측을 수행한 결과를 보여준다. 테스트 결과, EGSP는 각 chromosome에 대해 근사한 결과를 보였으며, 이 테스트를 통하여 유전자 구조예측 프로그램으로서 높은 성능을 가질 수 있음을 확인하였다.

추후 개선방향

유전자 구조 예측 프로그램은 현 단계에서, 복잡한 단계와 다수의 모듈 구성으로 인해, 서열 분석에 대한 다양한 노하우와 기술을 필요로 하고 있는 분야이며, 기술적 장벽이 높고 상업화의 진전으로 인해 활용을 위한 경제적인 대가를 차츰 많이 필요로 하고 있다.

EGSP는 국내 기술로 개발된 최초의 진핵생물에 대한 유전자 예측 프로그램이다. 다른 서열 분석 프로그램에 비해 유전자 예측 프로그램은 기술 장벽이 높아 실용성 있는 프로그램의 개발이 어려운 상황이었고, 이러한 장벽이 더욱 높아지고 있는 상황이다. 본 프로그램은 현재 기존의 일부 프로그램보다는 더 나은 성능을 보이고 있으며, 최고 성능의 프로그램에 비해서는 다소 낮은 성능으로 자체 평가되었다. 성능 향상을 위한 후속 연구에서는, 각 모듈의 결과를 추적하는 프로그램의 개발을 통하여, 각 단계에서 prediction의 성능이 변화되는 것을 추적하고 단계별 모듈을 개선하여, 훨씬 개선된 성능을 가진 프로그램을 완성할 수 있을 것으로 기대한다. *S. cerevisiae*에 대한 유전자 구조예측 성능 테스트에서도 이러한 점을 유추할 수 있다. *S. cerevisiae*에 대한 유전자의 구조 연구가 많이 진행되지 않아 promoter와 polyA에 대한 정보가 극히 부족하기 때문에 수정된 모델을 이용하여 유전자 구조예측을 수행하였다. 비교적 양호한 결과를 구할 수 있었지만, 진핵생물 유전자 구조의 중요한 정보인 intron과 함께 promoter와 polyA에 대한 정보가 풍부하다면 보다 완성도 높은 프로그램이 개발될 수 있다.

각 signal의 후보 위치에서 최적화된 state의 구성을 계산하기 위해서 initiation probability, transition probability, length distribution, segment probability, signal probability 이렇게 5가지 계산 모듈에 대해 각각 log odds score 계산 방식을 적용했다. 이 계산 방식은 각 signal의 위치에서 직관적인 비교 값을 제공하지만, 전체적인 score의 비율의 계산에 어려움을 준다. 그에 비해 GenScan에서 사용되는 전체적인 score의 비교방식에서는 서로 다른 계산 모듈에 대한 score의 직관적인 비교가 어렵다. 추후에는 두 방식을 적절하게 조화시켜 나갈 예정이다.

GenScan의 경우 dynamic programing을 통한 global score계산 방식에 의해 입력 염기서열의 길이가 길어질수록 정확성이 떨어지는 모습을 보이는데 EGSP에서는 global score 계산방식과 local score 계산방식의 조합방식을 추가할 것이다. 그리고 state들의 segment probability를 계산할 때 segment내에서의 5th markov score 분포에 따라 state 별로 다른 weight를 둘 필요가 있음을 알게 되었다. 이와 같은 모듈의 추가를 통해 유전자 위치 분석에 조금 더 정확한 예측 결과를 기대할 수 있을 것이다.

GenScan을 비롯한 최근 프로그램들이 다양한 기계학습을 시험하여 개발되어 오고 있으나, 현재 사용을 위한 라이센스를 필수적으로 지불해야 하고, 그 대가가 차츰 부담스럽게 되어 가고 있는 실정이다. 한편, 자체적으로 수행하는 생명체에 적합한 gene prediction 환경을 구축하기 위해서는, 자체 개발에 대한 기술력이 매우 결정적인 기술요소가 된다. EGSP은 국내 기술로 개발된 프로그램으로서, 이러한 시스템 개발에 활용될 수 있고, 추후 개발되는 유전자 예측 프로그램의 비교용으로 유용하게 활용할 수 있을 것이다. EGSP의 prediction 프로그램은 http://218.48.131.86/~hstae/EGSP/index.html 에서 사용 및 시험할 수 있다.

Table 4. Performance analysis of EGSP prediction program for *Saccharomyces cerevisiae* chromosomes using exon hit analysis

chromosomes	# of annotated exons	# of predicted exons	exact match			# of partial match	# of overlap	total match		
			#	Sn(%)	Sp(%)			#	Sn(%)	Sp(%)
chr01	100	122	53	53.0	43.4	23	11	87	87.0	71.3
chr02	421	436	237	56.2	54.3	60	59	356	84.6	81.6
chr03	164	174	91	55.4	52.2	30	23	144	87.8	82.7
chr04	793	801	456	57.5	56.9	139	79	674	85.0	84.1
chr05	292	318	171	58.5	53.7	45	35	251	86.0	78.9
chr06	131	149	75	57.2	50.3	22	15	112	85.5	81.8
chr07	549	571	307	55.9	53.7	85	80	472	86.0	82.6
sum	2450	2571	1390	56.7	54.0	404	302	2096	85.5	81.5

Meanings of Sn, Sp, 'exact match', 'partial match', and 'overlap' are the same as those in Table 3.

참고문헌

1. Arques, D.G. and C.J. Michel. 1990. Periodicities in coding and noncoding regions of the genes. *J. Theor. Biol.* 143, 307-318.
2. Borodovsky, M. and J. McIninch. 1993. GENMARK: parallel gene recognition for both DNA strands. *Comp. Chem.* 17, 123-134.
3. Burge, C. and S. Karlin. 1997. Prediction of complete gene structures in human genomic DNA. *J. Mol. Biol.* 268, 78-94.
4. Claverie, J.M. and L. Bougueleret. 1986. Heuristic informational analysis of sequences. *Nucl. Acids Res.* 14, 179-196.
5. Delcher, A.L., D. Harmon, S. Kasif, O. White, and S.L. Salzberg. 1999. Improved microbial gene identification with GLIMMER. *Nucl. Acids Res.* 27, 4636-4641.
6. Dong, S. and D.B. Searls. 1994. Gene structure prediction by linguistic methods. *Genomics* 23, 540-551.
7. Fickett, J.W. 1982. Recognition of protein coding regions in DNA sequences. *Nucl. Acids Res.* 10, 5503-5518.
8. Fields, C.A. and C.A. Soderlund. 1990. gm: A practical tool for automating DNA sequence analysis. *Comp. Appl. Biosci.* 6, 263-270.
9. Gelfand, M.S. and M.A. Roytberg. 1993. Prediction of the intron-exon structure by a dynamic programming approach. *BioSystems* 30, 173-182.
10. Guigo, R., S. Knudsen, N. Drake, and T. Smith. 1992. Prediction of gene structure, *J. Mol. Biol.* 226, 141-157.
11. Konopka, A.K. and J. Owens. 1990. Complexity charts can be used to map functional domains in DNA. *Genet. Anal. Tech. Appl.* 7, 35-38.
12. Kulp, D., D. Haussler, M.G. Reese, and F.H. Eeckman. 1996. A Generalized hidden markov model for the recognition of human genes in DNA. *ISMB-96.* 134-142.
13. Michel, C.J. 1986. New statistical approach to discriminate between protein coding and non-coding regions in DNA sequences and its evaluation. *J. Theor. Biol.* 120, 223-236.
14. Périer, R.C., T. Junier, and P. Bucher. 1998. The eukaryotic promoter database EPD. *Nucl. Acids Res.* 26, 353-357.
15. Pruitt, K.D. and D.R. Maglott. 2001. RefSeq and LocusLink: NCBI gene-centered resources. *Nucl. Acids Res.* 29, 137-140.
16. Rabiner, L.R. 1989. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proc. IEEE.* 77, 257-285.
17. Salzberg, S., A.L. Delcher, K.H. Fasman, and J. Henderson. 1998. A Decision Tree System for Finding Genes in DNA. *J. Comp. Biol.* 5, 667-680.
18. Salzberg, S.L., A.L. Delcher, S. Kasif, and O. White. 1998. Microbial gene identification using interpolated Markov models. *Nucl. Acids Res.* 26, 544-548.
19. Shepherd, J.C.W. 1981. Method to determine the reading frame of a protein from the purine/pyrimidine genome sequence and ist possible evolutionary justification. *Proc. Natl. Acad. Sci. USA.* 78, 1596-1600.
20. Silverman, B.D. and R. Linsker. 1986. A measure of DNA periodicity. *J. Theor. Biol.* 118, 295-300.
21. Snyder, E.E. and G.D. Stormo. 1995. Identification of protein coding regions in genomic DNA. *J. Mol. Biol.* 248, 1-18.
22. Snyder, E.E. and G.D. Stormo. 1993. Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks. *Nucl. Acids Res.* 21, 607-613.
23. Solovyev, V.V., A.A. Salamov, and C.B. Lawrence. 1994. Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames. *Nucl. Acids Res.* 22, 5156-5163.
24. Staden, R. and A.D. McLachlan. 1982, Codon preference and its use in identifying protein coding regions in long DNA sequences. *Nucl. Acids Res.* 10, 141-156.
25. Uberbacher, E. and J. Mural. 1991. Locating protein coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc. Natl. Acad. Sci. USA.* 88, 11261-11265.

(Received November 3, 2003/Accepted December 8, 2003)

ABSTRACT: A Eukaryotic Gene Structure Prediction Program Using Duration HMM

Hongseok Tae and Kiejung Park* (Information and Technology Institute, SmallSoft Co., Ltd., Daejeon 305-811, Korea)

Gene structure prediction, which is to predict protein coding regions in a given nucleotide sequence, is the most important process in annotating genes and greatly affects gene analysis and genome annotation. As eukaryotic genes have more complicated stuructures in DNA sequences than those of prokaryotic genes, analysis programs for eukaryotic gene structure prediction have more diverse and more complicated computational models. We have developed EGSP, a eukaryotic gene structure program, using duration hidden markov model. The program consists of two major processes, one of which is a training process to produce parameter values from training data sets and the other of which is to predict protein coding regions based on the parameter values. The program predicts multiple genes rather than a single gene from a DNA sequence. A few computational models were implemented to detect signal pattern and their scanning efficiency was tested. Prediction performance was calculated and was compared with those of a few commonly used programs, GenScan, GeneID and Morgan based on a few criteria. The results show that the program can be practically used as a stand-alone program and a module in a system. For gene prediction of eukaryotic microbial genomes, training and prediction analysis was done with *Saccharomyces* chromosomes and the result shows the program is currently practically applicable to real eukaryotic microbial genomes.