文章编号:1001 - 2486(2004)04 - 0103 - 04

一种针对基因识别的 GHMM 简化算法*

李冬冬,杜耀华,王正志
(国防科技大学机电工程与自动化学院,湖南 长沙 410073)

摘 要:广义隐 Markov 模型是计算机基因识别的一种重要模型,它克服了传统隐 Markov 模型的状态段长成几何分布的缺陷,更加适合于计算机基因识别。其缺点在于计算量大,需要采用有效的简化算法。利用基因的结构特点,在不附加额外限制条件的情况下,提出了一种新的简化算法,其计算复杂度是序列长度的线性函数。对实际生物序列数据的测试结果表明了此简化算法的有效性。

关键词:广义隐 Markov 模型;Viterbi 算法;基因识别
中图分类号:Q61 文献标识码:A

A Simplified Algorithm to GHMM for Gene Finding

LI Dong-dong, DU Yao-hua, WANG Zheng-zhi
(College of Mechatronics Engineering and Automation, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: The generalized hidden Markov model (GHMM) is an important model for computational gene finding. Compared with the traditional hidden Markov model (HMM), GHMM needn't the assumption that the length of each state is geometrical distribution, while it is necessary for HMM. This property is appropriate for computational gene finding. The demerit of GHMM is its high computational complexity, which hinders it from being used practically. According to the characteristic of gene's structure, a novel simplified algorithm is proposed without any additional assumptions, and its computational complexity is linear with the length of sequence. The testing result for biological data demonstrates that the simplified algorithm is effective.

Key words: generalized hidden Markov model; Viterbi algorithm; gene finding

隐 Markov 模型(Hidden Markov Model,HMM)最早应用在语音识别中,并获得了极大的成功[1,2]。对生物序列(DNA 序列、蛋白质序列)而言,由于它与语音信号具有明显的相似性,因此,近年来,人们也开始把 HMM 用于生物序列的分析,包括序列比对、蛋白质二级结构预测、基因预测等等,并取得了一定的成功[3]。但是 HMM 的状态的段长成几何分布,这就限制了它的应用范围。比如对原核生物,其基因编码区的段长分布近似于伽玛分布,而非编码区则是近似于指数分布[4]。为此,人们提出了广义隐 Markov 模型(Generalized Hidden Markov Model, GHMM),其最大的特点在于状态段长可以是任意分布,从而克服了 HMM 中状态段长成几何分布的缺陷,使模型能够更好地贴近实际的系统。GHMM 的优点使得它被广泛用于基因识别[6,7],并且获得了较高的准确率。事实上,迄今为止,基因识别准确率最高的几种软件都是采用的这一模型[4~8]。然而,这一推广的代价是计算量的增加,以至于不能直接在实际的基因识别中应用,而必须使用简化算法。本文根据基因的结构特点,提出了一种新的简化算法,在不添加额外条件的情况下,获得了与序列长度成线性关系的计算复杂度,对实际生物序列数据的测试表明,此算法是有效的。

1 模型描述

计算机基因识别的任务是对一个给定的 DNA 序列,指出其中基因的位置及其完整结构。在 HMM 模型的框架中,这一问题可以形式化地描述如下[4,5]:

* 收稿日期:2004 - 03 - 15
基金项目:军队基础研究项目(JC - 02 - 03 - 021)
作者简介:李冬冬(1974—),男,博士生。

对一个长度为 L 的 DNA 序列 x，定义它的一个状态分解 ϕ，它由一个状态序列 $q = \{q_q, q_2, \cdots, q_n\}$ 和一个辅助的长度序列 $d = \{d_1, d_2, \cdots, d_n\}$ 构成，每个状态 q_i 对应一个长度 d_i，表示状态 q_i 产生一个长度为 d_i 的 DNA 子序列，这些子序列首尾相连正好构成给定的 DNA 序列 x，这里的长度序列满足条件：$L = \sum_{i=1}^{n} d_i$。于是，DNA 序列 x 可以表示成 $x = \{x_1, x_2, \cdots, x_n\}$，其中 x_i 代表由 q_i 产生的子序列。

考察空间 $\Omega = \Phi_L \times X_L$，其中 X_L 是所有长度为 L 的 DNA 序列的集合，Φ_L 是所有长度为 L 的状态分解的集合，对特定的 DNA 序列 $x \in X_L$ 和特定的状态分解 $\phi \in \Phi_L$，其联合分布概率为：

$$P(\phi, x) = \pi_{q1} f_{q1}(d_1) E_{q1}(x_1) \prod_{k=2}^{n} T_{q_{k-1}, q_k} f_{q_k}(d_k) E_{q_k}(x_k) \tag{1}$$

其中，π 为初始时刻系统状态的概率分布，f 为状态产生的子序列长度的概率分布，T 为状态转移概率，E 为状态产生特定子序列元素的概率。基因识别就是对给定的 DNA 序列，寻找一个状态分解 ϕ，它使得如下的条件概率最大：

$$P(\phi \mid x) = \frac{P(\phi, x)}{P(x)} \tag{2}$$

通常地，假设系统产生一个特定的 DNA 序列 x 的概率与其状态分解无关，因此，对(2)式中的条件概率求极大值，等价于对(1)式中的联合分布概率求极大值。这一极大值问题可以采用 Viterbi 算法来求解。

GHMM 与传统的 HMM 的区别在于长度序列 d 的构成：传统的 HMM 中，所有的子序列长度均为 1，这就导致状态段长成几何分布；而 GHMM 中的子序列长度是可变的，它可以是几何分布，也可以是其他的任何分布，从而可以更好地贴近实际的系统。然而，这一改进的代价是计算量的增大[2]：针对传统 HMM 的 Viterbi 算法，其计算复杂度是 $O(N^2 L)$，其中 N 是模型的状态数目；而针对 GHMM 的推广算法，其计算复杂度为 $O(N^2 L^3/2)$。由于一般的 DNA 序列长度在几十 K 个碱基对到几百 K 个碱基对之间，因此 GHMM 的计算量大概是 HMM 的 $10^8 \sim 10^{10}$ 倍。文献[4]中提出可以对状态的段长设置上限，文献[5]则假定某些状态服从几何分布，两者都获得了与序列长度成线性关系的计算复杂度，但也都给基因的结构增加了额外的限制。

从原核生物基因的结构知道，其序列可以看作若干个片断的串联，这些片断对应于 GHMM 中的状态，它们分为三种类型：非编码区、正向链编码区和逆向链编码区，各个片断之间被四类特殊密码子分开。因此，在基因识别的过程中，只要对出现这四类特殊密码子的地方加以计算，就能够获得识别的结果，同时，注意到非编码区的长度分布服从指数分布，其概率能够分解，因而能够实现递推算法。据此，本文提出了一种新的简化方法，在不增加额外限制的条件下，同样获得了与长度成线性的计算复杂度。图 1 是本文中使用的原核生物基因识别模型，它能够同时识别出正向链和逆向链上的基因。

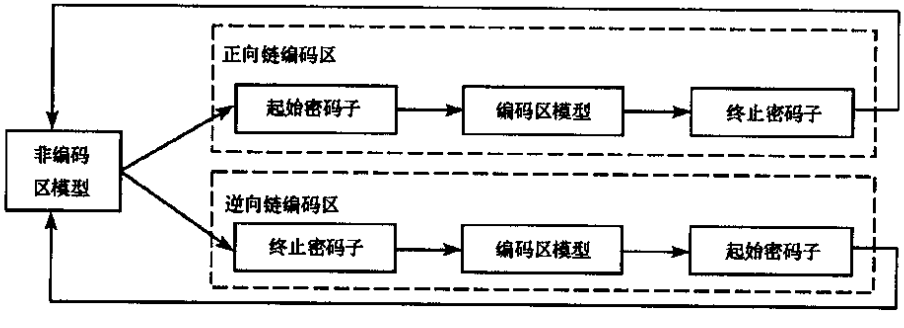

图 1 原核生物基因识别的 GHMM 模型

Fig.1 GHMM for prokaryotic gene finding

2 算法

定义 $V_i(j)$ 为序列的在第 j 个碱基处出现第 i 个状态的最大概率，并定义一个列表 A_j，它包含了所有正好终止于第 j 个碱基处的状态的信息，对其中第 k 个元素而言，它包括起始位置 a_k、结束位置 j、状

态的长度 λ_k、类型 y_k,以及前一个状态的类型 x_k。由此可以写出如下算法:

算法:GHMM 的 Viterbi 算法

1) 初始化:$V_i(0)=1$

2) 循环:对 $j=1,2,\cdots,L-1$,作如下操作:

如果在位置 j 不存在任何一类特殊密码子,直接执行下一次循环;

否则

根据位置 j 处的密码子类型,向前搜索,构造列表 A_j;

计算:

$$V_i(j)=\max_{k\in A_j,y_k=i}\{V_{x_k}(a_k-1)T_{x_k,y_k}f_{y_k}(d_k)E_{y_k}(x_{a_k,j})\} \tag{3}$$

3) 终止:

$$V_i(L)=\max_{k\in A_L,y_k=i}\{V_{x_k}(a_k-1)T_{x_k,y_k}f_{y_k}(d_k)E_{y_k}(x_{a_k,L})\} \tag{4}$$

其中的 $x_{a_k,j}$ 表示一段子序列,起始点为 a_k,终止点为 j,其余各项与(1)式中的项含义相同。

此算法的复杂度集中于第二步。由于终止密码子不能编码任何氨基酸,而起始密码子则可以编码氨基酸,从而在基因的开放阅读框架(Open reading frame,ORF)中,可能出现起始密码子,而不能出现终止密码子。因此,在构造列表 A_j 的时候,如果是搜索编码区的起点,那么一旦搜索到一个终止密码子的时候,搜索就结束了。因此,搜索的长度(当前位置 j 与上一个终止密码子之间的距离)和结果(A_j 中的元素数目)都是常数,不会随着序列的长度而变化。这样,在每一次循环中的计算次数都与序列长度 L 无关,整个算法的复杂度就是序列长度 L 的线性函数。然而,如果要搜索的是非编码区的起点,由于对非编码区不存在类似于编码区的限制,因此,现在需要搜索第 j 个位置以前所有的碱基对,从而 A_j 中的元素与 j 成正比,而对其中的每个元素,当采用公式(3)计算其 $V(j)$ 值的时候,还要对 A_j 中的所有元素执行一次运算,因而整个算法的复杂度将是序列长度 L 的立方函数。

正如有研究者曾经指出的[4],非编码区的长度近似服从指数分布,对大肠杆菌基因组序列数据的分析也验证了这一结论(数据未给出)。另外,在 GHMM 框架下,状态产生子序列元素的概率本身就是可分解的。在这两个前提条件下,下述结论成立:

结论 非编码区产生子序列的概率是可分解的,即 $P_N(x_{a,c})=CP_N(x_{a,b})P_N(x_{b+1,c})$,其中 $1\leqslant a<b<c\leqslant L$,$x_{x,y}$ 表示从第 x 个碱基到第 y 个碱基的一段子序列,下标 N 表示讨论的状态是非编码区,C 是常数。

证明:

由于 $P_N(x_{a,c})=f_N(|x_{a,c}|)E_N(x_{a,c})$,这里的 $|x_{x,y}|$ 代表子序列 $x_{x,y}$ 的长度,$f_N(\cdot)$ 是非编码区的长度分布函数,它是一个指数函数。$E_N(\cdot)$ 是非编码区状态产生子序列元素的概率,它是可以分解的。即:

$$f_N(|x_{a,c}|)=f_N(|x_{a,b}|)M_{N,N}f_N(|x_{b+1,c}|) \tag{5a}$$

$$E_N(x_{a,c})=E_N(x_{a,b})E_N(x_{b+1,c}) \tag{5b}$$

其中的 $M_{N,N}$ 是指数分布 $f_N(\cdot)$ 中参数的倒数,对特定的基因组,它是一个常数。这两个结论来自于我们的前提条件。将这两式代入 $P_N(\cdot)$ 的定义式,得:

$$\begin{aligned}P_N(x_{a,c})&=\{f_N(|x_{a,b}|)M_{N,N}f_N(|x_{b+1,c}|)\}\{E_N(x_{a,b})E_N(x_{b+1,c})\}\\&=\{f_N(|x_{a,b}|)E_N(x_{a,b})M_{N,N}f_N(|x_{b+1,c}|)E_N(x_{b+1,c})\\&=P_N(x_{a,b})M_{N,N}P_N(x_{b+1,c})\end{aligned} \tag{6}$$

故结论成立。证毕。

注意到在前面的算法中,(3)式中的 $T_{N,N}$(即 T_{x_k,y_k} 的下标均为 N 时)代表非编码区内部的转移概率,与(6)式相比较可以看到,$M_{N,N}$ 实际上就是 $T_{N,N}$,但由于 $M_{N,N}$ 不存在转移概率的含义,我们称之为转移系数。

在此结论下,如果要搜索非编码区的起点,当搜索碰到下一个起始密码子 S 的时候(设它的位置为

$m, m < j$),搜索过程结束。因为在此之前已经计算得到了 $V_N(m)$,假设还有另一个非编码区起点(设起始位置为 $n, n < m$),那么从该起点起计算得到的 $V_N(j)$ 为:

$$\begin{aligned} V_N^n(j) &= V_N(n-1) M_{N,N} f_N(|x_{n,j}|) E_N(x_{n,j}) \\ &= V_N(n-1) M_{N,N} f_N(|x_{n,m}|) E_N(x_{n,m}) M_{N,N} f_N(|x_{m+1,j}|) E_N(x_{m+1,j}) \\ &\leqslant V_N(m) M_{N,N} f_N(|x_{m+1,j}|) E_N(x_{m+1,j}) \\ &= V_N^m(j) \end{aligned} \tag{7}$$

可见,不需要对 S 之前的序列进行搜索,这就保证了对非编码区进行搜索时,其搜索的范围和结果都是常数(与序列长度无关)。因此,最终算法的复杂度将只是序列长度的线性函数。

3 数据验证与结论

为了验证算法的有效性,采用大肠杆菌的全基因组原始数据对算法做了测试。数据来自 GenBank 134.0 版,其访问号为 AE000111 至 AE000510,共 400 条记录,4 639 221 个碱基对,4 288个基因。本文中的算法计算得到的结果如表 1 所示,作为对比,表中还包括 GeneMark.hmm 对 E.Coli 数据的测试结果[4]。

表 1 基因预测的结果

Tab.1 The result of gene finding

Method	S_n	S_p	Exact Pred.	3'-end Pred.	Total Pred.
Method of this paper	97.16%	97.42%	2876(67%)	1221(28.5%)	4438
GeneMark.hmm	…	…	2483(58%)	1592(37%)	4440

注:S_n 和 S_p 分别表示核酸水平的敏感性和特异性,其计算以核酸为单位。Exact Pred.表示预测完全正确(5'端和 3'端边界都预测正确)的编码区数目,3'-end Pred.表示只有 3'端边界预测正确的编码区数目,Total Pred.表示程序预测出来的全部编码区数目。这三者的计算以编码区为单位。表中的…符号表示原始文献中没有相应的数据。

从计算结果可以看到,本文的算法具有较高的预测正确率。与同是采用 GHMM 作为模型框架的 GeneMark.hmm 相比,本文的简化算法的预测正确率有较大的提高,这证明本文中提出的算法是有效的。

需要指出的是,对真核生物基因而言,虽然其结构要复杂得多,但本文中的方法仍然能够应用,只是需要作一些相应的修改。这种扩展是本文下一步的主要工作。

参考文献:

[1] Rabiner L R, Juang B H. An Introduction to Hidden Markov Models[J]. IEEE ASSP Magazine, 1986, 3(1): 4 – 16.

[2] Rabiner L R. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition[J]. Proceedings of the IEEE, 1989, 77(2): 257 – 285.

[3] Durbin R, Eddy S, Krogh A, Mitchison G. Biological Sequence Analysis[M]. Cambridge University Press, 1998.

[4] Lukashin A V, Borodovsky M. GeneMark.hmm: New Solutions for Gene Finding[J]. Nucleic Acids Research, 1998, 26(4): 1107 – 1115.

[5] Burge C, Karlin S. Prediction of Complete Gene Structure in Human Genomic DNA[J]. Journal of Molecular Biology, 1997, 268(1): 78 – 94.

[6] Krogh A. Two Methods for Improving Performance of a HMM and Their Application for Gene Finding[P]. In T. Gaasterland, et al., ed., Proc. of Fifth Int. Conf. on Intelligent Systems for Molecular Biology, Menlo Park, CA: AAAI Press, 1997: 179 – 186.

[7] Kulp D, Haussler D, Reese M G, Eeckman F H. A Generalized Hidden Markov Model for the Recognition of Human Genes in DNA[P]. Proc. Conf. on Intelligent Systems in Molecular Biology, Menlo Park, CA: AAAI Press, 1996: 134 – 142.

[8] Rogic S, Mackworth A K, Ouellette F B F. Evaluation of Gene-Finding Programs on Mammalian Sequences[J]. Genome Research, 2001, 11(5): 817 – 832.