

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN



# BÁO CÁO ĐỒ ÁN MÔN HỌC TOÁN ỨNG DỤNG VÀ THỐNG KÊ Chủ đề: Mô hình Markov ẩn

**Giảng viên lý thuyết:** PGS. TS. Nguyễn Đình Thúc  
**Giảng viên hướng dẫn thực hành:**

- Thầy Nguyễn Văn Quang Huy
- Cô Võ Nam Thục Đoan

**Lớp:** 20TN

**Thành viên thực hiện:**

- 20120131 – Nguyễn Văn Lộc
- 20120536 – Võ Trọng Nghĩa
- 20120572 – Nguyễn Kiều Minh Tâm

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 4-5 NĂM 2022

# Lời nói đầu

# Mục lục

<b>1</b>	<b>Lý thuyết mô hình Markov ẩn</b>	<b>3</b>
1.1	Các thành phần của một mô hình Markov ẩn là gì? Chúng khác gì với mô hình Markov? . . . . .	3
1.2	Các giả thiết (assumptions) đặt ra cho mô hình Markov ẩn là gì? Tìm ví dụ các bài toán mà các giả thiết này hợp lý và bất hợp lý. . . . .	3
1.3	Cho một mô hình Markov ẩn với các tham số đã biết, thuật toán tiến trước (forward algorithm) được dùng để xác định độ hợp lý (likelihood) của một chuỗi quan sát (observation). Mô tả và đánh giá độ phức tạp của thuật toán tiến trước. . . . .	4
1.4	Cho một mô hình Markov ẩn với các tham số đã biết, thuật toán Viterbi được dùng để xác định chuỗi trạng thái (state) khả dĩ nhất. Mô tả và đánh giá độ phức tạp của thuật toán Viterbi. . . . .	4

# 1 Lý thuyết mô hình Markov ẩn

Mô hình Markov ẩn (Hidden Markov model – HMM) là một mô hình máy học cổ điển thông dụng trong việc xử lý chuỗi.

## 1.1 Các thành phần của một mô hình Markov ẩn là gì? Chúng khác gì với mô hình Markov?

Một mô hình Markov ẩn có cấu tạo như sau<sup>1,2</sup>:

- $Q = q_1 q_2 \dots q_N$  : tập hợp  $N$  trạng thái (states).
- $A = a_{11} \dots a_{ij} \dots a_{NN}$  : ma trận xác suất chuyển (transition probability matrix)  $A$ ,  $a_{ij}$  là xác suất chuyển từ trạng thái  $i$  sang trạng thái  $j$ ,  $\sum_{j=1}^N a_{ij} = 1, \forall i$ .
- $O = o_1 o_2 \dots o_T$  : chuỗi các quan sát (observations), lấy từ bộ từ vựng (vocabulary)  $V = v_1, v_2, \dots, v_V$ .
- $B = b_i(o_t)$  : ma trận xác suất phụ thuộc trạng thái (emission probabilities), thể hiện xác suất một quan sát  $o_t$  được tạo thành từ trạng thái  $i$ .
- $\pi = \pi_1, \pi_2, \dots, \pi_N$  : phân phối xác suất ban đầu theo trạng thái, có nghĩa là  $\pi_i$  thể hiện xác suất xích Markov bắt đầu ở trạng thái  $i$ . Một số trạng thái  $j$  có thể có  $\pi_j = 0$ , do chúng không thể là trạng thái ban đầu của xích Markov.

So với xích Markov (Markov chains), mô hình Markov ẩn có thêm 2 thành phần là  $O$  – chuỗi các quan sát và  $B$  – ma trận xác suất phụ thuộc trạng thái.

## 1.2 Các giả thiết (assumptions) đặt ra cho mô hình Markov ẩn là gì? Tìm ví dụ các bài toán mà các giả thiết này hợp lý và bất hợp lý.

**Các giả thiết (assumptions) đặt ra cho mô hình Markov ẩn.**

Một mô hình Markov ẩn có 2 giả thiết chính:

- Xác suất của một trạng thái cụ thể chỉ phụ thuộc vào trạng thái ngay trước đó (Markov Assumptions):

$$P(q_i | q_1 \dots q_{i-1}) = P(q_i | q_{i-1}). \quad (1)$$

**1.3** Cho một mô hình Markov ẩn với các tham số đã biết, thuật toán tiến trước (forward algorithm) được dùng để xác định độ hợp lý (likelihood) của một chuỗi quan sát (observation). Mô tả và đánh giá độ phức tạp của thuật toán tiến trước.

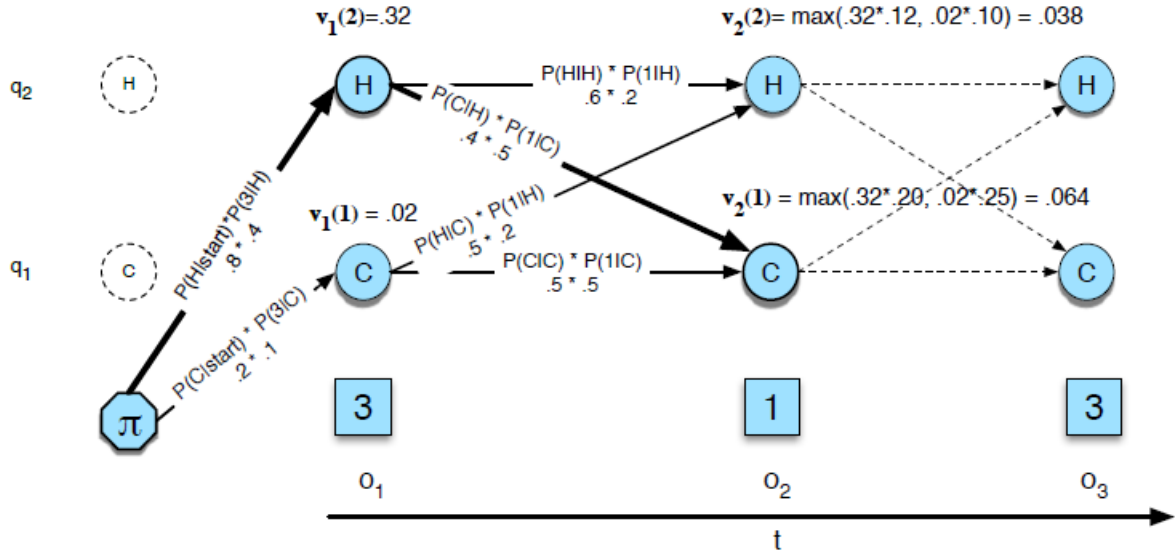
**1.4** Cho một mô hình Markov ẩn với các tham số đã biết, thuật toán Viterbi được dùng để xác định chuỗi trạng thái (state) khả dĩ nhất. Mô tả và đánh giá độ phức tạp của thuật toán Viterbi.

Với mỗi mô hình chứa những biến ẩn (hidden variables), chẳng hạn như mô hình Markov ẩn, việc quyết định chuỗi biến nào là nền tảng (underlying source) của một (vài) chuỗi quan sát gọi là **giải mã** (decode). Công việc này được thực hiện bởi một **decoder**.

**Giải mã (Decoding):** cho một mô hình Markov ẩn  $\lambda = (A, B)$  và một chuỗi các quan sát  $O = o_1, o_2, \dots, o_T$ , tìm chuỗi các trạng thái  $Q = q_1 q_2 q_3 \dots q_T$  có khả năng xảy ra nhất.

Ta có thể đưa ra chuỗi tốt nhất như sau: với mỗi chuỗi trạng thái ẩn có thể, ta có thể sử dụng thuật toán tiến trước và tính độ hợp lý của chuỗi quan sát có được từ chuỗi trạng thái ẩn đó. Sau đó, ta có thể chọn chuỗi với độ hợp lý cao nhất. Dễ thấy từ phần trước rằng ta không thể làm theo cách này vì số lượng chuỗi trạng thái có thể là hàm mũ.

Thay vào đó, thuật toán giải mã phổ biến nhất là **thuật toán Viterbi** (Viterbi algorithm). Giống như thuật toán tiến trước, Viterbi là một thuật toán quy hoạch động sử dụng dynamic programming trellis. Viterbi cũng thể hiện một biến thể của quy hoạch động, thuật toán minimum edit distance.



Hình 1: Viterbi trellis dùng cho việc tìm đường đi tốt nhất qua các không gian trạng thái ẩn cho việc ăn kem. Các trạng thái ẩn được đóng khung trong vòng tròn, các quan sát được đóng khung hình vuông. Các hình tròn nét đứt màu trắng thể hiện. Hình trên thể hiện quá trình tính  $v_i(j)$  cho 2 trạng thái ở 2 bước. Quá trình tính toán trong mỗi ô sử dụng công thức (3)  $v_t(j) = \max_{1 \leq N-1} v_{t-1} a_{ij} b_j(o_t)$ . Xác suất trong mỗi ô được tính theo công thức (2)  $v_t(j) = P(q_0, q_1, \dots, q_{t-1}, o_1, o_2, \dots, o_t, q_t = j | \lambda)$ .

Hình 1 cho ta một ví dụ về Viterbi trellis về tìm chuỗi trạng thái ẩn tốt nhất cho chuỗi quan sát 3 1 3. Ý tưởng là xử lý chuỗi quan sát từ trái sang phải, điền vào những trellis. Mỗi ô trong trellis,  $v_t(j)$ , thể hiện xác suất mô hình Markov ẩn ở trạng thái  $j$  sau khi trải qua  $t$  quan sát và đã duyệt qua các trạng thái có thể  $q_1, q_2, \dots, q_{t-1}$ , với tự động hóa  $\lambda$  biết trước. Giá trị mỗi ô  $v_t(j)$  được tính đệ quy bằng cách sử dụng đường đi có thể xảy ra nhất dẫn tới ô đó. Mỗi ô thể hiện xác suất được tính bằng công thức sau:

$$v_t(j) = \max_{q_1 \dots q_{t-1}} P(q_1 \dots q_{t-1}, o_1, o_2 \dots o_t, q_t = j | \lambda). \quad (2)$$

Cần lưu ý rằng ta lấy đại diện con đường có thể xảy ra nhất bằng cách lấy giá trị lớn nhất của các chuỗi trạng thái trước đó  $\max_{q_1 \dots q_{t-1}}$ . Giống như các thuật toán quy hoạch động khác, thuật toán Viterbi điền giá trị vào mỗi ô bằng phương pháp đệ quy. Biết rằng ta đã tính xác suất của mỗi trạng thái ở thời điểm  $t-1$ , ta tính xác suất Viterbi bằng cách tìm mở rộng (extensions) có thể nhất của con đường dẫn đến ô hiện tại. Với mỗi trạng thái  $q_j$  cho trước ở thời điểm  $t$ , giá trị  $v_t(j)$  được tính như sau:

$$v_t(j) = \max_{i=1}^N v_{t-1}(i) a_{ij} b_j(o_t). \quad (3)$$

3 nhân tử trong công thức (3) được sử dụng cho việc mở rộng con đường trước đó để tính xác suất Viterbi ở thời điểm  $t$ :

$v_{t-1}(i)$  : xác suất con đường Viterbi (previous Viterbi path probability) từ bước trước đó.

$a_{ij}$  : xác suất chuyển (transition probability) từ trạng thái  $q_i$  trước đó sang trạng thái  $q_j$  hiện tại.

$b_j(o_t)$  : độ hợp lý (state observation likelihood) của quan sát  $o_t$  từ trạng thái  $j$ .

Mã giả của thuật toán Viterbi như sau:

---

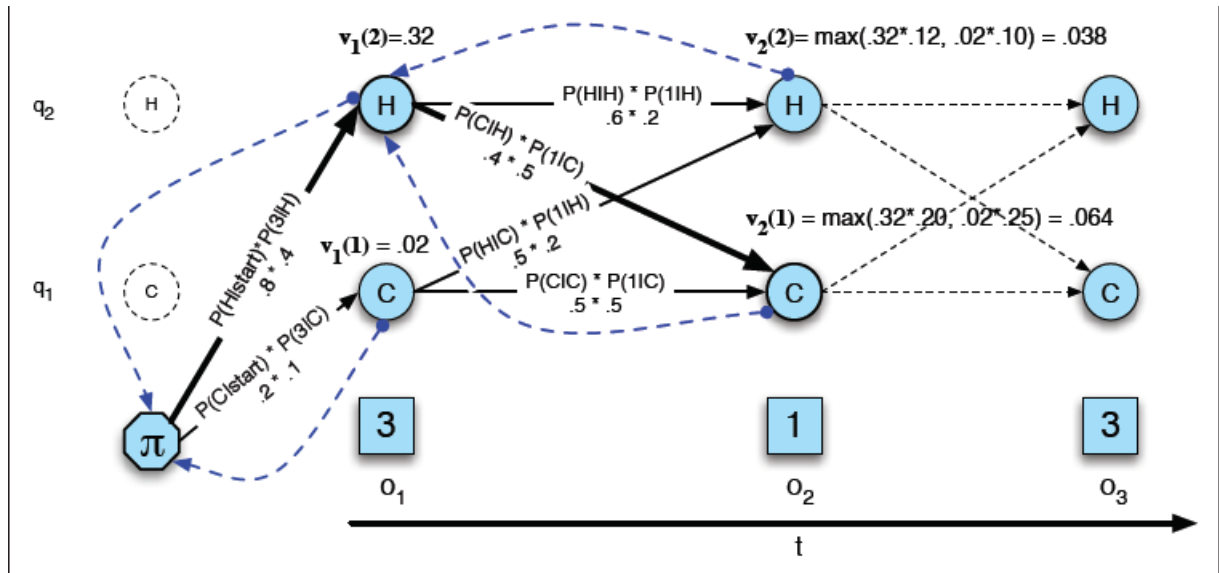
**Algorithm 1** Thuật toán Viterbi

---

```
1: function VITERBI(observation of len  $T$ , state_graph of len  $N$ )
2:   tạo một ma trận xác suất đường đi  $viterbi[N, T]$ 
3:                                     ▷ khởi tạo
4:   for mỗi trạng thái  $s$  from 1 to  $N$  do
5:      $viterbi[s, 1] \leftarrow \pi_s * b_s(o_1)$ 
6:      $backpointer[s, 1] \leftarrow 0$ 
7:                                     ▷ đệ quy
8:   for mỗi bước  $t$  from 2 to  $T$  do
9:     for mỗi trạng thái  $s$  from 1 to  $N$  do
10:       $viterbi[s, t] \leftarrow \max_{s'=1}^N (viterbi[s', t-1] * a_{s',s} * b_s(o_t))$ 
11:       $backpointer[s, t] \leftarrow \arg \max_{s'=1}^N (viterbi[s', t-1] * a_{s',s} * b_s(o_t))$ 
12:                                     ▷ Kết thúc
13:    $best\_path\_prob \leftarrow \max_{s=1}^N (viterbi[s, T])$ 
14:    $best\_path\_pointer \leftarrow \arg \max_{s=1}^N (viterbi[s, T])$ 
15:    $best\_path \leftarrow$  con đường bắt đầu ở trạng thái  $best\_path\_pointer$ ,
16:   , đi theo  $backpointer[]$  đến các trạng thái return  $best\_path, best\_path\_prob$ 
```

---

Thuật toán Viterbi tương tự như thuật toán tiến trước, ngoại trừ việc thuật toán này lấy **giá trị lớn nhất** của các xác suất đường đi (path probabilities) trước đó trong khi thuật toán tiến trước lấy **tổng**. Ngoài ra, thuật toán Viterbi có thêm một thành phần so với thuật toán tiến trước: **backpointer**. Lý do cho sự khác biệt này là trong khi thuật toán tiến trước cần cung cấp một độ hợp lý quan sát (observation likelihood), thuật toán Viterbi cần cung cấp xác suất và chuỗi trạng thái hợp lý nhất. Ta tính chuỗi trạng thái tốt nhất này bằng cách lưu lại vết (keep track) của đường đi từ các trạng thái ẩn dẫn tới mỗi trạng thái, giống như trong hình 2, và sau đó, quay lại tìm đường đi tốt nhất tới trạng thái đầu (Viterbi backtrace).



Hình 2: Viterbi backtrace

Ta có định nghĩa của thuật toán Viterbi như sau:

1. Khởi tạo:

$$v_1(j) = \pi_j b_j \quad 1 \leq j \leq N$$

$$bt_1(j) = 0 \quad 1 \leq j \leq N$$

2. Đệ quy:

$$v_t(j) = \max_{i=1}^N (v_{t-1}(i) a_{ij} b_j(o_t)) \quad 1 \leq j \leq N, 1 < t \leq T$$

$$bt_t(j) = \arg \max_{i=1}^N (v_{t-1}(i) a_{ij} b_j(o_t)) \quad 1 \leq j \leq N, 1 < t \leq T$$

3. Kết thúc:

$$\text{Xác suất tốt nhất: } P^* = \max_{i=1}^N (v_T(i))$$

$$\text{Điểm bắt đầu của con đường tốt nhất: } q_T^* = \arg \max_{i=1}^N (v_T(i))$$



## Tài liệu tham khảo

- [1] Nguyễn Tiến Dũng. *Mô hình Markov ẩn*. URL: <https://jurgendn.github.io/mathematics/2020/06/16/Hidden-Markov-Model>.
- [2] Daniel Jurafsky and James H. Martin. “Hidden Markov model”. In: *Speech and Language Processing*. 3rd ed. Stanford, 2021. Chap. Appendix A. URL: [https://web.stanford.edu/%5Csim\\$jurafsky/slp3/A.pdf](https://web.stanford.edu/%5Csim$jurafsky/slp3/A.pdf).