**Hidden Markov Model**

1. **TÓM TẮT**

Mô hình Markov ẩn là mô hình thống kê phổ biến để mô hình chuỗi dữ liệu biến đổi nhiều theo thời gian. Trong việc xử lý ngôn ngữ tự nhiên NLP ( Natural Language Processing), HMM đã được ứng dụng với những thành công to lớn trong việc giải quyết các vấn đề như trích thuộc tính của tiếng nói, phân khúc các cụm từ.

1. **GIỚI THIÊU**

Mô hình Markov ẩn là một công cụ thống kê rất mạnh trong việc mô hình hóa các chuỗi có thể sinh ra , hay nói cách khác là các chuỗi mà có thể đặc trưng bởi các chuỗi trạng thái sinh ra các chuỗi quan sát khác nhau .

Mô hình Markov ẩn đã được ứng dụng trong rất nhiều lĩnh vực của xử lý tín hiệu nói chung, và xử lý tiếng nói nói riêng. Và đã ứng dụng thành công trong NLP ( Natural Languages Processing ) như: part-of-speech tagging, phrase chunking, extracting target information from document.

* Định nghĩa của mô hình Markov ẩn là:



* S là tập hợp tất cả các trạng thái:



* V là tập hợp tất cả các quan sát được:



* Q là chuỗi các trạng thái có thể xảy ra, có chiều dài T



* Và tương ứng với nó là chuỗi các quan sát có thể quan sát được



* A là bảng chuyển đổi, chứa những giá trị xác suất chuyển đổi từ trạng thái i sang trạng thái j, và những xác suất chuyển đổi này độc lập với thời gian:



* B là bảng xác suất quan sát, chứa những giá trị xác suất của quan sát k từ trạng thái i, độc lập với thời gian:



* Π là bảng xác suất đầu tiên:



Ta giả sử mô hình Markov ẩn thỏa mãn 2 điều kiện sau:

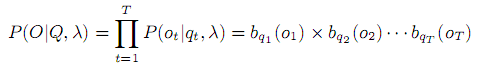
**Thứ 1:** là mô hình Markov first-order, trạng thái hiện tại chỉ phụ thuộc vào trạng thái liền trước nó, đặc trưng cho tính nhớ của mô hình.

**Thứ 2:** quan sát được ở thời điểm t, chỉ phụ thuộc vào trạng thái hiện tại, độc lập với các trạng thái và quan sát được trong quá khứ.

1. **EVALUATION**

Cho một mô hình Markov ẩn và một chuỗi quan sát được O, ta có thể tính được P(*O|λ*), là xác xuất xuất hiện của chuỗi quan sát đó cho bởi mô hình Makov ẩn. Từ đó ta có thể đánh giá chất lượng mô hình khi dự đoán về chuỗi O cho trước và chọn được mô hình thích hợp nhất.

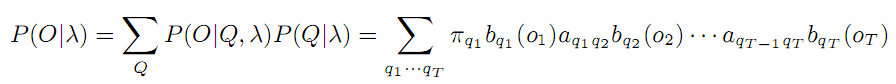
Xác suất chuỗi quan sát O cho chuỗi trang thái Q được tính bởi:



Xác suất của chuỗi trạng thái Q ứng mô hình Markov ẩn là:



Từ đó ta tính được xác suất của chuỗi quan sát O cho bởi HMM là:



Chúng ta có thể tính trực tiếp xác xuất chuỗi O, tuy nhiên lượng phép tính cần dùng là rất lớn.

Do đó, chúng ta có thể sử dụng phương pháp tốt hơn là nhận biết lượng tính toán dư, sau đó đệm chúng để đạt được mục tiêu giảm độ phức tạp trong tính toán. Ta thực hiện đệm bằng biểu đồ mắt cáo cho mỗi bước tính, ta tính giá trị đệm (ký hiệu α)tại mỗi trạng thái bằng cách tổng tất cả trạng thái trước nó. α lúc này là xác suất cua chuỗi O tại mức trạng thái tại thời điểm t.

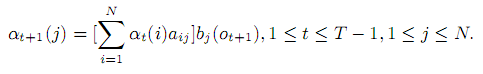


Sau đó ta điền đầy đủ các thông số α vào biểu đồ mắt cáo. Tổng giá trị của cột cuối chính là xác suất chuỗi quan sát được.

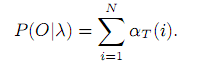
**Các bước thực hiện thuật toán như sau:**

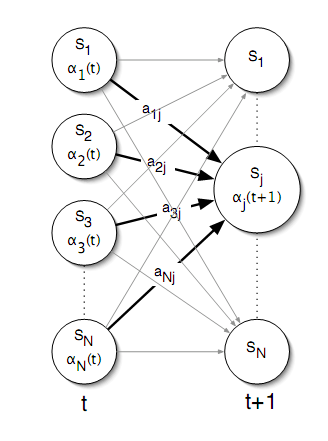
1. Initialisation: (Điền cột đầu tiên):  


2. Industion: (Điền các cột còn lại)



3. Termination: (Tính xác xuất chuỗi O)





Hình trên biểu diễn các phép tính trong 1 bước từ trạng thái thứ t sang trạng thái thứ t+1. Bằng phương pháp sử dụng α để đệm, ta giảm được lượng tính toán từ còn phép tính.

1. **DECODING**

Mục đích của việc decoding là xác đinh được chuỗi trạng thái mà có khả năng đưa ra được chuỗi quan sát cho trước nhiều nhất. Một giải pháp cho vấn đề này là sử dụng thuật toán Viterbi.

Thuật toán Viterbi là một dạng khác của thuật toán biểu đồ mắt cáo, tương tự như thuật toán Forward, ngoại trừ chọn các giá trị xác suất chuyển đổi lớn nhất tại mỗi bước, thay vì tính tổng của chúng.

Đầu tiên, chúng ta phải xác định:

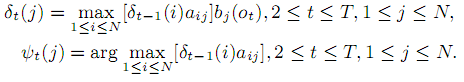


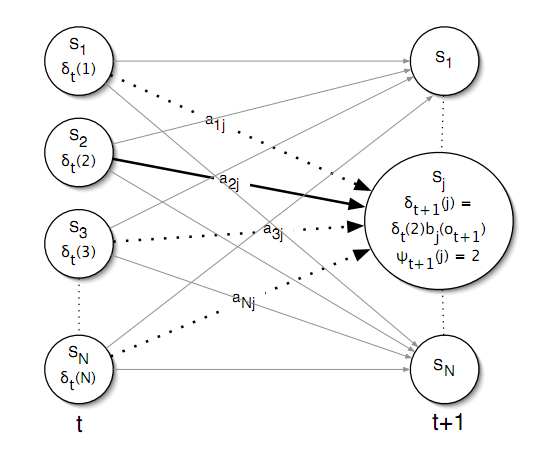
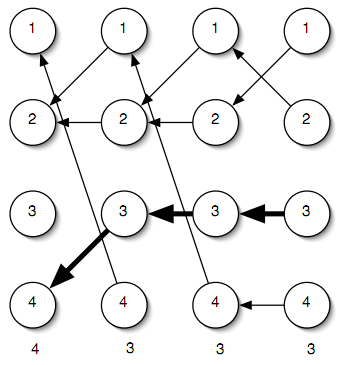
* **Thuật toán Viterbi được thực hiện như sau:**

1. Initialisation:



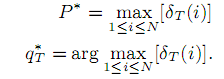
1. Recursion:



Bước Recursion của thuật toán Viterbi Bước Backtracking của thuật toán Viterbi

1. Termination:



1. Optimal state sequence backtracking:



Sau đây ta xét một ví dụ về dự báo thời tiết để hiểu rõ hơn về thuật toán Viterbi

**The Viterbi Algorithm: Thuật toán Viterbi để xác định chuỗi thời tiết có thể xảy ra nhất**

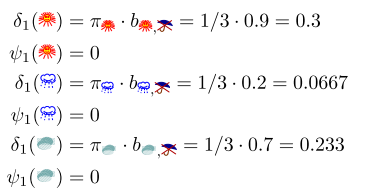
Có 3 loại thời tiết: nắng , mưa , và sương mù . Và ta giả thiết rằng 1 loại thời tiết sẽ xảy ra diễn ra trong cả ngày, không thay đổi vào một khoảng thời gian trong ngày.

Bây giờ giả thiết rằng bạn bị nhốt trong một căn phòng trong nhiều ngày, và bạn được hỏi về thời tiết ở bên ngoài. Và chỉ có một vài bằng chứng bạn có là việc một người bạn đem bữa ăn đến cho bạn là có mang một cây dù  hay không . Bạn không biết thời tiết như thế nào khi bạn bị khóa.

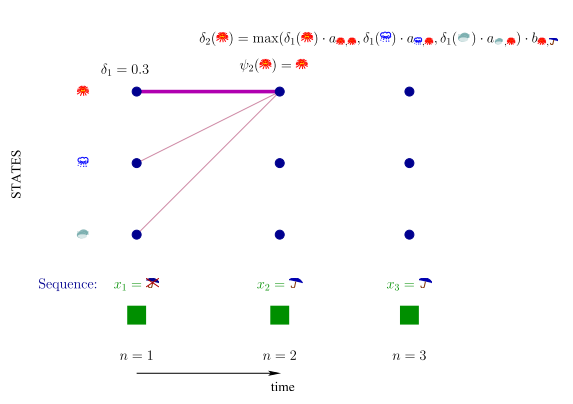
Vào ba ngày đầu tiên bạn quan sát thấy người bạn như sau: .

Tìm xác suất lớn nhất chuổi thời tiết sử dụng thuật toán Viterbi. (giả thiết răng xác suất ưu tiên là như nhau vào ngày 1)

1. Tính giá trị ban đầu ( Initialisation ):



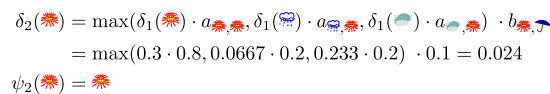
Các công thức trên được tính tương tự như bước 1 ở trên.



Hình 5: Thuật toán Viterbi để tìm chuỗi thời tiết có thể xảy ra nhất. Ta tìm thấy đường dẫn đến trạng thái trời nắng ở thời điểm n= 2.

2. Đệ quy (Recursion ):

Chúng ta tính khả năng tới trạng thái  từ tất cả 3 trạng thái trước, và chọn cái có khả năng xảy ra nhất với

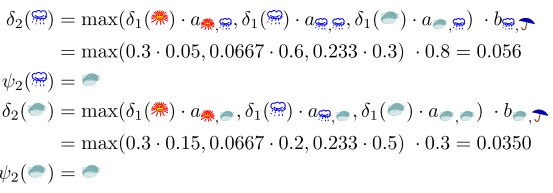


Các công thức trên được tính như bước 2 ở trên.

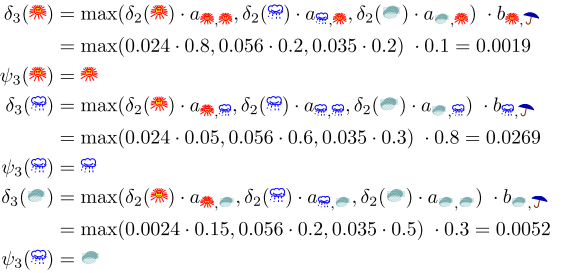
Tức là trong hình vẽ tìm đường dẫn mà dẫn đến likelihood lớn nhất xét về likelihood tốt nhất ở thời điểm trước và bước chuyển tiếp từ nó. Rồi nhân với likelihood hiện tại được cho bởi trạng thái hiện tại. Và kết quả là sẽ tìm thấy đường dẫn tốt nhất.

Likelihood là được tính trong δ, và trạng thái trước đó có thể nhất là trong ψ. Xem hình 5

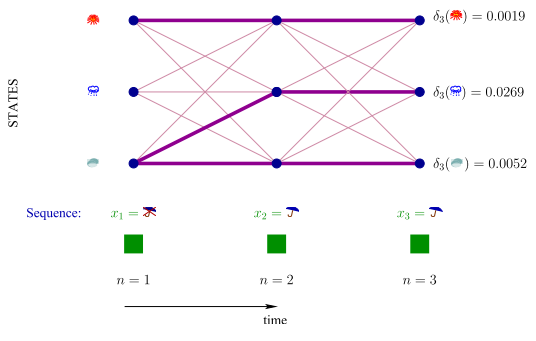
Thực hiện tương tự với trạng thái trời  và :



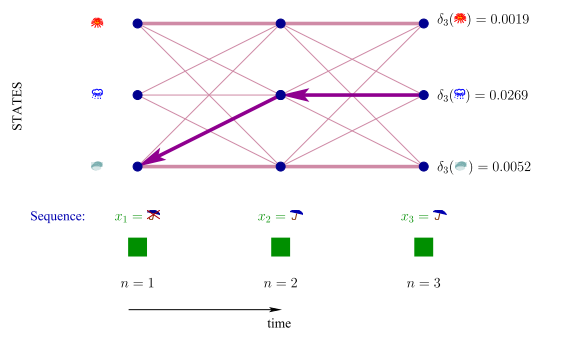
n=3:



Cuối cùng, chúng ta sẽ thu được đường dẫn kết thúc có thể xảy ra nhất ở mỗi trạng thái của mô hình. Xem hình 6



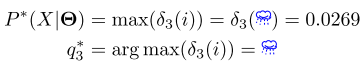
Hình 6: Thuật toán Viterbi để tìm chuỗi thời tiết có khả năng xảy ra nhất tại n= 3.



Hình 7: Thuật toán Virtebi tìm ra chuổi thời tiết có thể xảy ra nhất.

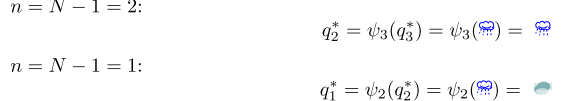
3. Termination:

Con đường có thể xảy ra nhất đã được xác định, bắt đầu bằng cách tìm ra trạng thái cuối cùng của chuổi có thể xảy ra nhất.



4. Backtracking:

Chuỗi trạng thái tốt nhất có thể đọc từ vector ψ. Xem hình 7



Theo cách đó thì chuỗi thời tiết có thể xảy ra nhất là:



1. **LEARNING**

Cho trước 1 hệ thống các mẫu của một quá trình, chúng ta có thể đánh giá các thông số của mô hình Makov ẩn λ = (A, B, π) sao cho chũng thể hiện quá trình đó một cách tối ưu. Có 2 phương pháp thông thường để thực hiện, tùy thuộc vào dạng của mẫu ví dụ cho trước, đó là training có giám sát và training không giám sát.

Nếu mẫu ví dụ có cả đầu vào và đầu ra thì ta thực hiện training có giám sát, với đầu vào là chuỗi quan sát được còn đầu ra là chuỗi trạng thái. Nếu mẫu ví dụ chỉ có đầu vào thì ta chỉ có thể training không giám sát bằng cách “đoán” các thông số của mô hình để đạt được chuỗi quan sát đã cho.

Trong bài này ta chỉ thảo luận về training có giám sát, còn training không giám sát với thuật toán Baum -Wetch đã được trình bày trong [6].

Giải pháp đơn giản nhất để thiết lập mô hình Makov ẩn có thông số *λ*là sử dụng một loạt mẫu ví dụ. Tiêu biểu cho phương pháp này là giải pháp PoS tagging.

Ta mô tả 2 nhóm:

là nhóm tag, tương đương nhóm trạng thái của HMM

là nhóm word, tương đương nhóm quan sát của HMM

Để xác định các thông số mô hình trên ta dùng đánh giá khả năng cực đại (MLE – Maximum Likelihood Estimation) từ chuỗi quan sát và chuỗi trạng thái tương ứng.

Ma trận chuyển đổi được tính:



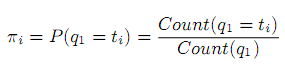
là số lần chuyện trạng thái từ sang và là số lần xuất hiện.

Ma trận quan sát được tính bởi:



là số lần quan sát được ứng với và là số lần xuất hiện.

Thông số π cho bởi:



Trong thực tế khi tính toán các thông số ta cần các biện pháp làm mượt để tránh số đếm được là 0 và tăng hiệu năng của mô hình với các dữ liệu không xuất hiện trong mẫu ví dụ dùng training.