MỤC LỤC

CHƯƠNG 1: XÁC ĐỊNH YÊU CẦU BÀI TOÁN

1.1. Tổng quan về bài toán Part-of-Speech Tagging - Gắn nhãn từ loại

1.2. Một số phương pháp gán nhãn từ

1.3. Phân tích bài toán sử dụng mô hình xác suất gán nhãn từ

CHƯƠNG 2: MÔ HÌNH MARKOV ẨN

2.1. Mô hình xác suất gán nhãn từ

2.2. Mô hình Markov ẩn

2.3. Thuật toán Viterbi

CHƯƠNG 3: THIẾT KẾ HỆ THỐNG GÁN NHÃN TỪ

3.1. Xử lý dữ liệu học

3.2. Thiết kế mô hình học máy

3.2.1. Học không giám sát

3.2.2. Học có giám sát

3.3. Thuật toán Viterbi với mô hình tri-gram

3.4. Xử lý từ chưa được học

3.5. Kết quả thực nghiệm

TÀI LIỆU THAM KHẢO

Chương 1
XÁC ĐỊNH YÊU CẦU BÀI TOÁN

1.1. Tổng quan về bài toán Part-of-Speech Tagging - Gắn nhãn từ loại

Gắn nhãn từ loại là quá trình phân loại một từ trong 1 đoạn văn bản dựa trên định nghĩa của từ đó và bối cảnh từ đó được sử dụng.

- Ví dụ: Cho câu: Robot watches videos to learning cooking.

Từ	**Nhãn**
• Robot	Noun
• watches	Verb
• videos	Noun
• to	Preposition
• learn	Verb
• cooking	Noun

Việc gán nhãn từ giúp cho việc thu thập thông tin trong 1 đoạn văn bản trở nên đơn giản hơn và thường được áp dụng trong bước tiền xử lý dữ liệu trong các bài toán xử lý ngôn ngữ tự nhiên. Tuy nhiên, việc gán nhãn từ loại không phải là 1 bài toán đơn giản bởi tiếng Anh sở hữu nhiều từ có các nhãn khác nhau trong những bối cảnh khác nhau.

- Ví dụ: Cho 2 câu
  - Eagle **flies**(Verb) with the dove
  - The **flies**(Noun) and the honey pot

Một cách để phân biệt các từ có nhiều nhãn là sử dụng phương pháp phân tích ngữ nghĩa. Ta có thể suy luận được rằng: Từ ‘flies’ ở câu đầu là Verb vì trước nó là 1 danh từ ‘Eagle’, còn ở câu sau là Noun vì trước nó là từ ‘The’. Nói một cách khác, từ ‘files’ ở câu đầu được gắn nhãn Verb vì xác suất 1 động từ đứng trước một danh từ cao hơn xác suất 1 danh từ đứng trước 1 danh từ, cũng như xác suất 1 danh từ đứng trước từ ‘The’ cao hơn xác suất 1 động từ đứng trước ‘The’. Nhìn theo cách tiếp cận này, ta có thể xây dựng một mô hình gán nhãn dựa trên xác suất để gắn nhãn tự động

cho các từ dựa trên xác suất cao nhất. Các phương pháp dựa trên mô hình xác suất này đã được áp dụng thành công từ vài thập kỷ nay.

1.2. Một số phương pháp gán nhãn từ

- Hiện nay có 3 phương thức gán nhãn từ được sử dụng phổ biến:
  - Gán nhãn dựa trên quy tắc (**Rule-based method**) : Phương pháp này gắn nhãn từ dựa trên một tập các quy tắc được xây dựng sẵn. Phương pháp này rất hiệu quả trong trường hợp không có nhiều tài nguyên học, tuy nhiên khó có thể được áp dụng trong các bài toán phức tạp vì số lượng các quy tắc lớn sẽ dẫn đến sự rối loạn trong quá trình phân loại. Ví dụ về một quy tắc: Một từ đứng trước 'The' và 1 tính từ sẽ được gán nhãn là danh từ.
  - Gán nhãn từ dựa trên xác suất (**Probabilistic method**) : Sử dụng 1 tập dữ liệu các đoạn văn bản đã được gắn nhãn từ trước để tính các xác suất cần thiết và sử dụng những xác suất này để tính toán và gắn nhãn có xác suất cao nhất cho từng từ. Mô hình Markov ẩn hoạt động theo phương pháp này.
  - Gán nhãn từ dựa trên sự biến đổi (**Transformation-based method**) : Phương pháp này là sự kết hợp của 2 phương pháp trên. Phương pháp này gán nhãn có xác suất cao nhất cho từng từ rồi áp dụng một tập các quy tắc đã được xây dựng sẵn lên những từ này để có thể điều chỉnh nhãn nếu bước gán nhãn bằng xác suất không chính xác. Phương pháp này có độ chính xác rất cao bởi khả năng sửa lỗi và khả năng gắn nhãn cho các từ không có trong từ điển có độ chính xác cao hơn các phương pháp khác.

1.3. Phân tích bài toán gán nhãn từ sử dụng mô hình xác suất

- Phương pháp sử dụng : Mô hình **Markov** ẩn được train bằng tập dữ liệu các đoạn văn bản đã được gán sẵn nhãn.
- Đầu vào : Một đoạn văn bản.
- Đầu ra : Đoạn văn bản đã được gán nhãn.

  Ví dụ : Gán nhãn từ sử dụng dữ liệu nhãn TreeBank làm dữ liệu học

Input	Output
Eagle flies with the dove	Eagle/NNP flies/VBZ with/IN the/DT dove/NN
The flies and the honey pot	The/DT flies/NNS and/CC the/DT honey/NN pot/NN

- **Mô hình xác suất** : Gọi $w_1 \dots w_n$ là n từ trong đoạn văn bản đầu vào. Ta cần tìm các nhãn $q_1 \dots q_n$ sao cho xác suất $P(q_1 \dots q_n | w_1 \dots w_n)$ là cực đại.
  - Sử dụng quy tắc Bayes, ta có

$$P(q_1 \dots q_n | w_1 \dots w_n) = \frac{P(w_1 \dots w_n | q_1 \dots q_n) P(q_1 \dots q_n)}{P(w_1 \dots w_n)}$$

  - Do $P(w_1 \dots w_n)$ không phụ thuộc vào nhãn, ta cần tìm $q_1 \dots q_n$ sao cho xác suất là cực đại.
  - Xét $P(q_1 \dots q_n | w_1 \dots w_n) \approx \prod_{i=1}^{n} P(w_i | q_i)$
  - Đầu vào là một chuỗi : Giá trị của một nhãn phụ thuộc vào các nhãn đứng trước nó

$$P(q_1 \dots q_n) = P(q_1) P(q_2 | q_1) \dots P(q_n | q_{n-1} \dots q_1) = P(q_1) \prod_{i=2}^{n} P(q_i | q_{i-1} \dots q_1)$$

  - Tổng hợp:

$$P(q_1 \dots q_n | w_1 \dots w_n) = P(q_1) \prod_{i=2}^{n} P(q_i | q_{i-1} \dots q_1) \prod_{i=1}^{n} P(w_i | q_i)$$

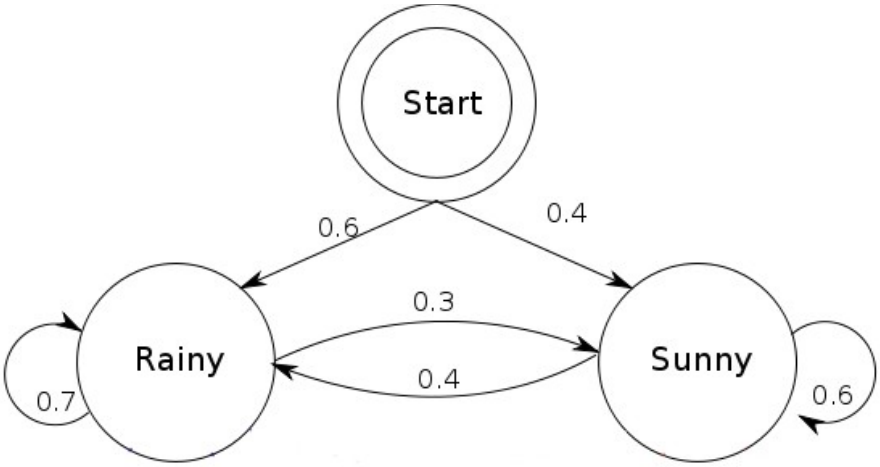
- Mô hình xác suất này chính là dạng tổng quát của mô hình Markov ẩn.

CHƯƠNG 2
MÔ HÌNH MARKOV ẨN

Mô hình Markov ẩn là **mô hình xác suất chuỗi** với chức năng : với một chuỗi các đơn vị đầu vào (từ, chữ cái, …), tính phân phối xác suất của chuỗi này trên một tập hợp chuỗi các trạng thái tương ứng với chuỗi đầu vào và chọn chuỗi các trạng thái có xác suất cao nhất để gắn nhãn cho chuỗi đầu vào. Nói một cách khác, mô hình thực hiện việc ánh xạ chuỗi đầu vào với chuỗi các trạng thái tương ứng sao cho xác suất các trạng thái là cao nhất.

2.1. Chuỗi Markov

Là một chuỗi các trạng thái có liên kết với nhau cùng với xác suất chuyển đổi giữa các trạng thái đó. Những trạng thái trong chuỗi Markov đều có thể **quan sát được**, vì thể chuỗi Markov thường được sử dụng để biểu diễn một chuỗi các sự kiện ngẫu nhiên dưới dạng đồ thị xác suất.



Ảnh 1: Chuỗi Markov cho trạng thái thời tiết

- Một chuỗi Markov bao gồm các thành phần:
  - $Q=q_1,q_2,...,q_m$: một tập gồm M trạng thái
  - $A=a_{1,1},a_{1,2},...,a_{m,1},...,a_{m,m}$ Ma trận xác suất chuyển đổi trạng thái, trong đó $a_{i,j}$ là xác suất chuyển đổi từ trạng thái i đến j Tính chất :

$\sum_{j=1}^{m} a_i \quad \forall i=\overline{1,M}$: Tổng xác suất chuyển đổi từ một trạng thái sang các trạng thái khác tính cả chính nó luôn bằng 1.

- $\pi_1, \pi_2, \ldots, \pi_m$: Xác suất khởi điểm của các trạng thái, trong đó π_i là xác suất chuỗi Markov sẽ bắt đầu ở trạng thái i .

Trong chuỗi Markov, xác suất của một trạng thái tại một thời điểm phụ thuộc vào các một số lượng nhất định các trạng thái đứng trước nó. Những mô hình xác suất có tính chất này được gọi là mô hình **n-gram**, với n là số lượng các trạng thái trong quá khứ mà trạng thái hiện tại phụ thuộc. Ở dạng tổng quát, xác suất một trạng thái trong chuỗi Markov được tính theo công thức:

$$P(q_i) = \prod_{i=2}^{m} P(q_i | q_{i-1} \ldots q_{i-n}) \quad \text{với} \quad m > n$$

trong đó n là số trạng thái q_i phụ thuộc. Chuỗi Markov trên được gọi là $g-order$ Markov. Chuỗi **first-order** Markov có $n=1$, đồng nghĩa với việc xác suất một trạng thái tại một thời điểm chỉ phụ thuộc duy nhất vào trạng thái đứng trước nó.

$$P(q_i | q_{i-1} \ldots q_{i-n}) = P(q_i | q_{i-1})$$

Với chuỗi Markov như hình trên với hai trạng thái {Sunny , Rainy}, ta có thể mô hình hóa được các trạng thái thời tiết dưới dạng mô hình xác suất. Ví dụ với bài toán : Cho xác suất trời nắng, trời mưa trong ngày một. Giả sử rằng mỗi một ngày chỉ có một trạng thái thời tiết, tính xác suất trời nắng trong ngày thứ hai và trời mưa trong ngày thứ ba. Với các xác suất khởi điểm và xác suất chuyển đổi giữa trời nắng và trời mưa, bài toán được giải quyết với chuỗi Markov như sau :

- Gọi x_i là trạng thái thời tiết trong ngày i . Xác suất cần tính :

$$P(x_2 = Sunny, x_3 = Rainy)$$

- Ta có :

$$P(x_2 = Sunny, x_3 = Rainy)$$

$= P(x_1 = Sunny, x_2 = Sunny, x_3 = Rainy) +$

$P(x_1 = Rainy, x_2 = Sunny', x_3 = Rainy)$

- Đối với chuỗi first-order Markov, ta có công thức :

$P(x_1 = Sunny, x_2 = Sunny, x_3 = Rainy)$

$= \pi_{Sunny} P(Sunny|Sunny) P(Rainy|Sunny)$

$= 0.4 * 0.6 * 0.4 = 0.096$

$P(x_1 = Rainy, x_2 = Sunny, x_3 = Rainy)$

$= \pi_{Rainy} P(Sunny|Rainy) P(Rainy|Sunny)$

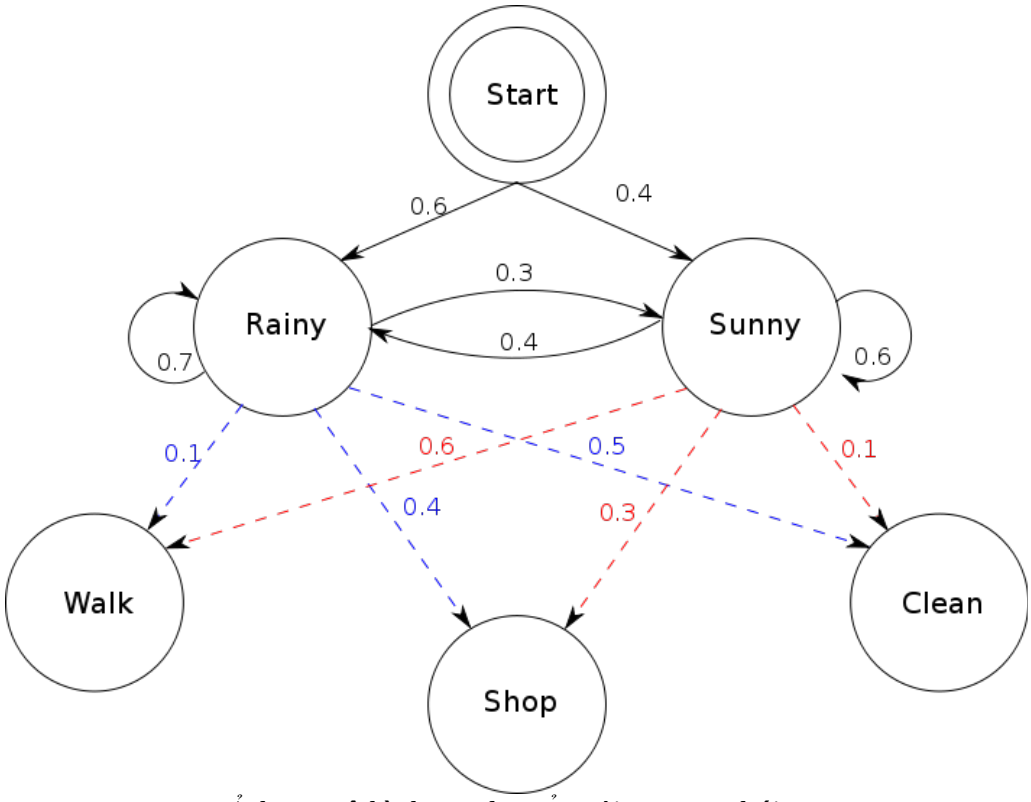
$= 0.6 * 0.3 * 0.4 = 0.072$

- Tổng hợp lại, ta có

$P(x_2 = Sunny, x_3 = Rainy) = 0.168$

2.2. Mô hình Markov ẩn

Chuỗi Markov được ứng dụng trong việc tính toán xác suất của một chuỗi các trạng thái mà ta có thể quan sát được. Tuy nhiên, thực tế tồn tại nhiều bài toán trong đó các trạng thái hoàn toàn **không thể quan sát được**, hay nói cách khác là bị ẩn đi. Ví dụ như bài toán gắn nhãn từ trong một đoạn văn bản, các từ là các sự kiện quan sát được, các nhãn (trạng thái) của mỗi từ được giấu đi và ta phải tìm các nhãn đó sao cho xác suất các nhãn tương ứng với đoạn văn bản đầu vào là lớn nhất. Ta gọi các nhãn đó là các trạng thái ẩn và đưa bài toán về biểu diễn dưới mô hình Markov ẩn.



Ảnh 2: Mô hình Markov ẩn với 2 trạng thái

- Mô hình Markov ẩn gồm các thành phần:
  - $Q=q_1,q_2,\ldots,q_m$: một tập gồm M trạng thái.
  - $A=a_{1,1},a_{1,2},\ldots,a_{m,1},\ldots,a_{m,m}$ Ma trận xác suất chuyển đổi trạng thái, trong đó $a_{i,j}$ là xác suất chuyển đổi từ trạng thái i đến j .
  - $O=o_1,o_2,\ldots,o_t$: chuỗi đầu vào gồm T phần tử.
  - $B=b_0(o_0),b_0(o_1),\ldots,b_m(o_0),\ldots b_m(o_n)$ Ma trận xác suất của mỗi sự kiện đầu vào tương ứng với từng trạng thái, trong đó $b_i(o_t)$ là xác suất sự kiện o_t được gán với trạng thái b_i .
  - $\pi_1,\pi_2,\ldots,\pi_m$: Xác suất khởi điểm của các trạng thái, trong đó π_i là xác suất chuỗi Markov sẽ bắt đầu ở trạng thái i .
- Mô hình **first-order** Markov có 2 tính chất :
  - Xác suất của một trạng thái ở một thời điểm chỉ phụ thuộc duy nhất vào trạng thái đứng trước nó.

$$P(q_i)=P(q_i|q_{i-1})$$

- Xác suất của một biến o_i chỉ phụ thuộc duy nhất vào trạng thái sinh ra biến này và không phụ thuộc vào bất kỳ trạng thái nào khác.

$$P(o_i|q_1 \dots q_m, o_1 \dots o_t) = P(o_i|q_i)$$

Với mô hình Markov ẩn như hình trên, {Rainy , Sunny} là 2 trạng thái ẩn và {Walk, Shop, Clean} là 3 sự kiện có thể quan sát được. Bài toán: Cho một chuỗi các sự kiện {Walk, Shop, Walk} với mỗi sự kiện xảy ra trong một ngày, gán nhãn trạng thái thời tiết {Rainy, Sunny} cho mỗi sự kiện trên. Để giải quyết bài toán này, ta có thể thực hiện theo cách: Sinh ra tất cả các chuỗi trạng thái có thể có ({Rainy, Rainy, Rainy}, {Rainy, Sunny, Sunny}, …) và áp dụng mô hình xác suất Markov ẩn để tính xác suất cho từng chuỗi này và chọn chuỗi có xác suất cao nhất. Tuy nhiên, phương pháp này có độ phức tạp $O(M^T)$ và không thể sử dụng trong những bài toán có số lượng trạng thái ẩn lớn. Nhiều thuật toán tối ưu đã được phát triển để giải quyết vấn đề này và một trong số đó là thuật toán **Viterbi**.

2.3. Thuật toán Viterbi

Thuật toán Viterbi được thiết kế nhằm mục đích tìm một chuỗi các trạng thái ẩn cho một chuỗi các sự kiện đầu vào một cách tối ưu. Thuật toán được xây dựng theo mô hình thuật toán tối ưu **quy hoạch động**. Thuật toán có độ phức tạp $O(M^n T)$, với n là số các trạng thái trong quá khứ mà trạng thái hiện tại phụ thuộc. Ở mô hình first-order Markov, thuật toán có độ phức tạp $O(M^2 T)$.

- Ý tưởng của thuật toán:
  - Tạo 1 ma trận $M \times N$ với M là số trạng thái ẩn, N là số sự kiện quan sát được.
  - Khởi tạo: Tính xác suất xuất hiện của các trạng thái ở sự kiện đầu tiên:

$$V_1(j) = \pi_j b_j(o_1) \forall\, j \in Q$$

  - Từ sự kiện tiếp theo đến sự kiện cuối cùng: Xét các sự kiện theo thứ tự từ trái qua phải, tại mỗi sự kiện t, tính xác suất xảy ra của từng trạng thái tại sự kiện đó bằng công thức

$$V_t(j) = \max_{i \in Q}(V_{t-1}(i) a_{i,j} b_j(o_t)) \forall\, j \in Q$$

với $V_{t-1}(i)$ là xác suất của trạng thái i đối với sự kiện trước đó.

Đường đi có xác suất cao nhất từ sự kiện đầu đến trạng thái j ở sự kiện t sẽ được lưu lại.

- Đường đi được chọn sẽ là quãng đường có xác suất cao nhất tại sự kiện cuối cùng.

◆ Pseudocode:

function VITERBI (tập các sự kiện độ dài *T*, tập các trạng thái độ dài *M*) **return** best-path

// tạo một ma trận xác suất đường đi $V[M,T]$

for mỗi trạng thái s từ 1 tới *M*

// khởi tạo

$V[s,1] \leftarrow \pi_s b_s(o_1)$

$path[s,1] \leftarrow 0$

for mỗi sự kiện t từ 2 đến *T*

// tính xác suất đường đi

for mỗi trạng thái s từ 1 tới *M*

$V[s,t] \leftarrow \max_{i \in Q} V[i,t-1] a_{i,s} b_s(o_t)$

$path[s,t] \leftarrow \operatorname{argmax}_{i \in Q} V[i,t-1] a_{i,s} b_s(o_t)$

// quay lui để tìm đường đi tốt nhất

last-state ← $\operatorname{argmax}_{i \in Q} V[i,T]$

state-bp ← $path[last-state, T]$

best-path← last-state

for mỗi sự kiện t từ *T* - 1 đến 1

best-path ←state-bp

state-bp← $path[state-bp, t]$

return best-path

CHƯƠNG 3
THIẾT KẾ HỆ THỐNG GÁN NHÃN TỪ

3.1. Xử lý dữ liệu học

Dữ liệu học bao gồm 2500 file POS, mỗi file gồm nhiều câu Tiếng Anh đã được gán nhãn với tổng cộng 52681 câu. Đối với mỗi file POS, ta tách từng câu trong file, sau đó tách riêng từ và nhãn rồi lưu thành 2 file gồm dữ liệu đoạn văn bản đầu vào và chuỗi trạng thái tương ứng.

- Dữ liệu học sử dụng trong bảng báo cáo này được gán nhãn dựa trên tập dữ liệu nhãn **TreeBank** gồm 36 nhãn như sau :

- CC - Coordinating conjunction
- CD - Cardinal number
- **DT** - Determiner
- EX - Existential there
- FW - Foreign word
- **IN** - Preposition or subordinating conjunction
- **JJ** - Adjective
- JJR - Adjective, comparative
- JJS - Adjective, superlative
- LS - List item marker
- MD - Modal
- **NN** - Noun, singular or mass
- NNS - Noun, plural
- **NNP** - Proper noun, singular
- NNPS - Proper noun, plural
- PDT - Predeterminer
- POS - Possessive ending
- PRP - Personal pronoun
- PRP$ - Possessive pronoun
- RB - Adverb
- RBR - Adverb, comparative
- RBS - Adverb, superlative
- RP - Particle
- SYM - Symbol
- **TO** - to
- UH - Interjection
- VB - Verb, base form
- VBD - Verb, past tense
- VBG - Verb, gerund or present participle
- **VBN** - Verb, past participle
- VBP - Verb, non-3rd person singular present
- **VBZ** - Verb, 3rd person singular present
- WDT - Wh-determiner
- WP - Wh-pronoun
- WP$ - Possessive wh-pronoun
- WRB - Wh-adverb

Ảnh 3: TreeBank tag

3.2. Thiết kế thuật toán gán nhãn từ

Mô hình Markov có các tham số :

- Xác suất chuyển đổi trạng thái $A = a_{1,1}, a_{1,2}, \ldots, a_{m,1}, \ldots, a_{m,m}$
- Xác suất trạng thái khởi điểm $\pi_1, \pi_2, \ldots, \pi_m$
- Xác suất từ được gắn nhãn $B = b_0(o_0), b_0(o_1), \ldots, b_m(o_0), \ldots b_m(o_n)$

Để học các tham số này, ta có thể áp dung phương pháp học không giám sát hoặc học có giám sát.

3.2.1. Học không giám sát

Phương pháp này được áp dụng khi tập dữ liệu học không được gán nhãn hoặc chỉ được gán nhãn một phần. Khi đó, các tham số của mô hình Markov sẽ được đánh giá bằng thuật toán **Expectation – Maximization**.

Expectation – Maximization (EM) là thuật toán lặp với mục đích tối ưu hóa việc đánh giá các tham số của một mô hình xác suất bằng việc tìm cực đại của hàm **likekihood** $L(\hat{\theta}|x)=P_{\hat{\theta}}(x)$, trong đó $\hat{\theta}$ làm tham số đang được đánh giá, x là biến ngẫu nhiên có một phân phối xác suất G nào đó và phân phối này có tham số θ . Để tối ưu hóa hàm likelihood, ta cần tìm $\hat{\theta}$ sao cho $P_{\hat{\theta}}(x)$ là lớn nhất, đồng nghĩa với việc $\hat{\theta}\approx\theta$. Thuật toán này gồm 2 bước :

- **E-step** : Đánh giá hàm likelihood với các tham số $\hat{\theta}$ được tính ở vòng lặp trước.
- **M-step** : Tính lại các tham số $\hat{\theta}$ sao cho các tham số này tối ưu hóa hàm likelihood tìm được ở E-step. Các tham số này sẽ được dùng để đánh giá hàm likelihood ở bước E của vòng lặp kế tiếp.

Đối với bài toán gán nhãn từ loại, thuật toán EM có thể được sử dụng để tính các xác suất cho mô hình Markov ẩn như sau : Với một chuỗi các sự kiện đầu vào O và một tập hợp các trạng thái ẩn Q , tính các xác suất

- Xác suất hồi quy (**backward probability**) : Xác suất của một chuỗi các sự kiện từ sự kiện o_{t+1} đến sự kiện cuối khi sự kiện o_t được gán nhãn i

$$\beta_t(i)=P(o_{t+1},o_{t+2},\dots,o_T|q_t=i)$$

$$= \sum_{j=1}^{M} a_{i,j}\,b_j(o_{t+1})\beta_{t+1}(j), 1\le t<T$$

với $\beta_T(i)=1 \forall i\in Q$

- Xác suất chuyển tiếp (**forward probability**) : Xác suất sự kiện o_t được gán nhãn j sau khi đi qua t sự kiện đầu tiên

$$\alpha_t(j) = P(o_1, o_2, \ldots, o_t, q_t = j)$$

$$= \sum_{i=1}^{M} a_{i,j} b_j(o_t) \alpha_{t-1}(i), 1 < t \leq T$$

với $\alpha_1(j) = \pi_j b_j(o_1) \forall\, j \in Q$

- Với hai xác suất trên, ta có thể tính được xác suất trạng thái o_t được gán nhãn i và trạng thái o_{t+1} được gán nhãn j

$$\xi_t(i,j) = \frac{(\alpha_t(i) a_{i,j} b_j(o_t + 1) \beta_{t+1}(j))}{(\sum_{k \in Q} \alpha_T(k))}$$

Đây chính là hàm likekihood cho tham số $A = a_{1,1}, a_{1,2}, \ldots, a_{m,m}$ của mô hình.

- Bằng xác suất trên, ta có thể tính được xác suất chuyển đổi trạng thái từ trạng thái i sang trạng thái j

$$\hat{a_{i,j}} = \frac{(\sum_{t}^{T-1} \xi_t(i,j))}{(\sum_{t}^{T-1} \sum_{k=1}^{Q} \xi_t(i,k))}$$

- Để tính tham số xác suất $B = b_0(o_0), b_0(o_1), \ldots b_m(o_n)$, ta cần tính xác suất nhãn j xuất hiện ở sự kiện thứ t

$$\chi_t(j) = P(q_t = j | O)$$

$$= \frac{(\alpha_t(j) \beta_t(j))}{(\sum_{k \in Q} \alpha_T(k))}$$

Đây chính là hàm likelihood cho tham số B của mô hình.

- Bằng xác suất trên, ta tính được xác suất một từ w được gán nhãn j

$$\hat{b}_j(w) = \frac{\left(\sum_{t=1 s.t. O_t = w}^{T} \chi_t(j)\right)}{\left(\sum_{t=1}^{T} \chi_t(j)\right)}$$

với $\sum_{t=1 s.t. o_t = w}^{T} \chi_t(j)$ có nghĩa là tính tổng $\chi_t(j)$ từ $t=1$ đến T nếu $o_t = w$

- Thuật toán EM :

function EM(tập các sự kiện độ dài *T,* từ điển *V,* tập các trạng thái *Q*) **return** *HMM=(A,B)*

khởi tạo *A, B*

lặp đến khi hội tụ

E-step

$$\chi_t(j) = \frac{(\alpha_t(j)\beta_t(j))}{\left(\sum_{k \in Q} \alpha_T(k)\right)} \forall t \in T, j \in Q$$

$$\xi_t(i,j) = \frac{(\alpha_t(i) a_{i,j} b_j(o_t+1)\beta_{t+1}(j))}{\left(\sum_{k \in Q} \alpha_T(k)\right)} \forall t \in T, i \text{ và } j \in Q$$

M-step

$$\hat{a}_{i,j} = \frac{\left(\sum_{t}^{T-1} \xi_t(i,j)\right)}{\left(\sum_{t}^{T-1} \sum_{k=1}^{Q} \xi_t(i,k)\right)} \forall i, j \in Q$$

$$\hat{b}_j(w) = \frac{\left(\sum_{t=1 s.t. O_t = w}^{T} \chi_t(j)\right)}{\left(\sum_{t=1}^{T} \chi_t(j)\right)} \forall w \in V, i \text{ và } j \in Q$$

return *A, B*

3.2.2. Học có giám sát

Phương pháp này chỉ được áp dụng khi tập dữ liệu học đã được gắn sẵn nhãn. Do đánh giá xác suất dựa trên các kết quả chính xác nên phương pháp này có độ chính xác cao hơn phương pháp học không giám sát. Các tham số của mô hình Markov ẩn được tính như sau :

- Xác suất chuyển đổi trạng thái $P(q_i, q_{i-1})$ được tính bằng cách đếm số lần q_i đứng trước q_{i-1} và chia cho số lần q_{i-1} xuất hiện trong tập dữ liệu học :

$$P(q_i|q_{i-1}) = \frac{C(q_{i-1}, q_i)}{C(q_{i-1})}$$

- Xác suất gán nhãn – từ $P(w|q_i)$ được tính bằng cách tính số lần từ w được gán nhãn q_i và chia cho số lần nhãn q_i xuất hiện :

$$P(w|q_i) = \frac{C(w, q_i)}{C(q_i)}$$

- Xác suất khởi tạo trạng thái π_i được tính bằng cách đếm số lần trạng thái i nằm ở đầu chuỗi và chia cho tổng số chuỗi trong tập dữ liệu học :

$$\pi_i = \frac{(C(q_i|\emptyset))}{N}$$

3.3. Thuật toán Viterbi với mô hình tri-gram

Đối với mô hình **tri-gram**, xác suất của một trạng thái tại một thời điểm phụ thuộc vào 2 trạng thái đứng trước nó

$$P(q_i|q_{i-1} \dots q_1) = P(q_i|q_{i-1}, q_{i-2})$$

do xác suất này không có nghĩa với q_2 và q_1 nên ta sẽ thêm 2 trạng thái $q_1 = q_0 = '*'$ ‘vào đầu câu và thêm trạng thái $q_{T+1} = 'STOP'$ vào cuối câu để đánh dấu điểm kết thúc câu. Ví dụ với chuỗi đầu vào *the cat jumps* có nhãn *DT NN VBZ,* ta có xác suất :

$$P(the\ cat\ jumps|DT\ NN\ VBZ) = P(DT|'*','*')\ x\ P(the|DT)\ x\ P(NN|DT,'*')$$
$$x\ P(cat|NN)\ x\ P(VBZ|NN,DT)\ x\ P(jumps|VBZ)\ x\ P(STOP|V,N)$$

Xác suất của mỗi phần tử tri-gram được bằng cách đếm số lần q_i đứng trước bi-gram (q_{i-1}, q_{i-2}) chia cho số lần bi-gram (q_{i-1}, q_{i-2}) xuất hiện

$$P(q_i|q_{i-1},q_{i-2})=\frac{(C(q_i,q_{i-1},q_{i-2}))}{(C(q_{i-1},q_{i-2}))}$$

Tuy nhiên, trong thực tế có nhiều những phần tử tri-gram được tính theo công thức trên có giá trị bằng 0 do sự đa dạng trong cách sử dụng từ dẫn tới sự thưa thớt dữ liệu các nhãn đi cùng nhau. Vì vậy, thay vì sử dụng công thức trên, ta sẽ tính $P(q_i|q_{i-1},q_{i-2})$ dựa trên sự kết hợp của uni-gram, bi-gram và tri-gram :

$$P(q_i|q_{i-1},q_{i-2})=\lambda_3\hat{P}(q_i|q_{i-1},q_{i-2})+\lambda_2\hat{P}(q_i|q_{i-1})+\lambda_1\hat{P}(q_i)$$ với

- $\lambda_3+\lambda_2+\lambda_1=1$ là các trong số tương ứng của tri-gram, bi-gram và uni-gram. Các trọng số này giúp cho việc phân loại nhãn linh hoạt hơn, không bị phụ thuộc vào tri-gram, từ đó tối ưu việc xử lý với các chuỗi nhãn chưa có trong dữ liệu học và tránh việc overfit dữ liệu học. Giá trị các trọng số này được xác định thông qua thuật toán **Deleted-Interpolation**.

◆ Thuật toán Deleted-Interpolation

function DELETED -INTERPOLATION (dữ liệu học có *N* phần tử) **return** $\lambda_1,\lambda_2,\lambda_3$

$\lambda_1 \leftarrow 0$
$\lambda_2 \leftarrow 0$
$\lambda_3 \leftarrow 0$

foreach tri-gram t_1,t_2,t_3 with $C(t_1,t_2,t_3) > 0$

tìm **giá trị lớn nhất** trong 3 công thức

case $\frac{(C(t_1,t_2,t_3)-1)}{(C(t_1,t_2)-1)}$: $\lambda_3 \leftarrow \lambda_3 + C(t_1,t_2,t_3)$

case $\frac{(C(t_2,t_3)-1)}{(C(t_2)-1)}$: $\lambda_2 \leftarrow \lambda_2 + C(t_1,t_2,t_3)$

case $\frac{(C(t_3)-1)}{(N-1)}$: $\lambda_1 \leftarrow \lambda_1 + C(t_1,t_2,t_3)$

normalize $\lambda_1,\lambda_2,\lambda_3$

return $\lambda_1,\lambda_2,\lambda_3$

3.4. Xử lý từ chưa được học

Tiếng Anh là một ngôn ngữ có vốn từ rất đa dạng với những từ có khả năng kết hợp với nhau để tạo thành một từ mới. Thêm vào đó, việc những từ ngữ mới được hình thành một cách không giới hạn khiến việc học tất cả các từ trong tiếng Anh trở thành một bài toán khó. Để giải quyết vấn đề thiếu hụt dữ liệu từ, người ta đã thiết kế một phương pháp xử lý những từ chưa được học dựa theo đặc điểm của ngôn ngữ tiếng Anh. Trong tiếng Anh, những từ có cùng phần kết thúc (**suffix**) thường có nhãn giống nhau. Ví dụ như : những từ có phần kết thúc là **-ly, -able** thường là tính từ; **-ed** là động từ thì quá khứ,… Dựa theo đặc điểm này, ta có thể tính được phân phối xác suất giữa phần kết thúc của các từ và nhãn tương ứng của chúng thông qua các từ trong tập dữ liệu học. Dựa theo phân phối này, ta sẽ tính được xác suất nhãn của các từ chưa được học bằng cách xét phần kết thúc của từ đó.

Ở phương pháp này, các suffix có độ dài từ 1-4 thường được xét với độ dài tối đa của một suffix của một từ bằng độ dài của từ đó trừ 2. Xác suất suffix/nhãn được tính bằng cách đếm số lần xuất hiện của suffix đó rồi chia cho số lần nhãn đó xuất hiện. Suffix của những từ có độ dài từ 4 trở xuống thường không mang lại thông tin nên những từ này được bỏ qua. Những từ chưa được học được giả định là những từ có nhãn **open-class** (Danh từ, động từ, tính từ, trạng từ), nên những từ không gắn nhãn này trong tập dữ liệu học sẽ được bỏ qua. Những từ chưa được học sẽ được gắn nhãn theo suffix có độ dài lớn nhất của nó.

3.5. Kết quả thực nghiệm

- Accuracy score : Cross validate test với 10 tập test, mỗi tập gồm 500 câu với tổng cộng 12000 từ.

Accuracy	Bigram	Trigram
Gắn nhãn từ đã được học	96,12%	96,33%
Gắn nhãn cho từ chưa được học	88,41%	89,54%
Tổng hợp	95,83%	96,01%

TÀI LIỆU THAM KHẢO

[1]. Daniel Ramage (2007), *Hidden Markov Models Fundamentals, S*tanford University.

[2]. Daniel Jurafsky & James H. Martin (2007), *Speech and Language Processing: An introduction to natural language processing,*

computational linguistics, and speech recognition (2nd Edition).

[3]. Scott Thede & Mary Harper (2002), *A Second-Order Hidde Markov Model for Part-of-Speech Tagging,* School of Electrical and Computer Engineering, Purdue University.

[4]. Michael Collins (2013), *Tagging Problems, and Hidden Markov Models*, Columbia University.