Sequence labeling

Bài toán Gán nhãn chuỗi

Nội dung

1. Định nghĩa bài toán
2. Bài toán Gán nhãn từ loại (Part-of-speech tagging)
3. Bài toán Nhận diện thực thể có tên (Named entity recognition)
4. HMM
5. CRF
6. Độ đo đánh giá

Định nghĩa bài toán

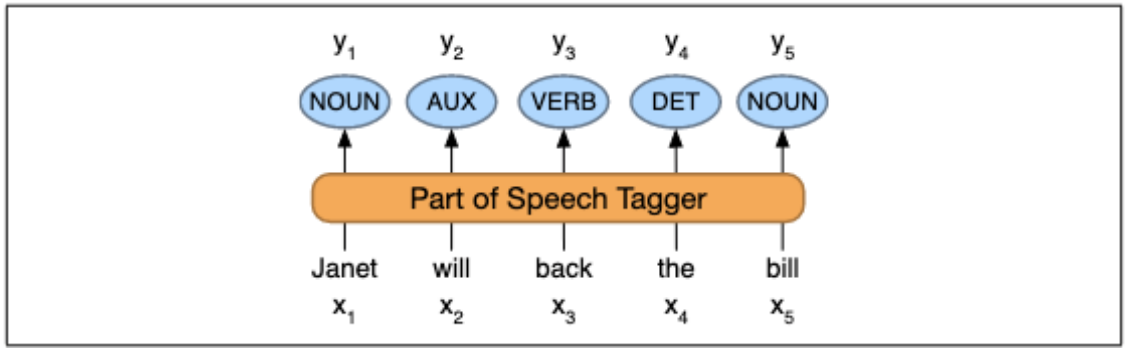
- Input: Một chuỗi X có độ dài là n từ. Ký hiệu:

$$X = <w_1\ w_2\ w_3\ ....\ w_n>$$

- Output: n nhãn ứng với từng phần tử/từ trong chuỗi X. Ký hiệu:

$$y = <y_1\ y_2\ y_3\ ...\ y_n>$$

Việc dự đoán và gán nhãn/thẻ cho từng từ trong chuỗi đầu vào được gọi là “tagging”.



Các bài toán liên quan đến Gán nhãn chuỗi

- Bài toán 1: Gán nhãn từ loại (Part of speech tagging).
- Bài toán 2: Nhân diện thực thể có tên (Named entity recognition).

Từ loại (Part of speech)

- Định nghĩa:
  - Từ loại là lớp từ có cùng bản chất ngữ pháp, được phân chia theo **ý nghĩa khái quát**, theo **khả năng kết hợp** với các từ ngữ khác trong ngữ lưu và **thực hiện những chức năng ngữ pháp** nhất định trong câu (Đinh Văn Đức. *Ngữ pháp tiếng Việt – Từ loại*).

- Các tiêu chí phân định từ loại:
  - Ý nghĩa khái quát của từ: sự vật, hành động, tính chất...
  - Khả năng kết hợp với các từ ngữ khác
  - Chức năng ngữ pháp (chức vụ ngữ pháp, chức năng thành phần câu)

Từ loại trong tiếng Anh

- Trong tiếng Anh, từ loại thường rơi vào 2 dạng sau:
  - **Lớp từ mở (Open word classes)**:
    - **Danh từ, động từ, tính từ, phó từ**. Số lượng mỗi từ loại thuộc nhóm này có thể từ một vài nghìn đến cả trăm nghìn từ.
    - Gồm những từ nội dung (**content words)**: từ mang nghĩa quan trọng hay nghĩa từ vựng (lexical meaning) như:
      - home (nhà ở, quê hương), bridge (cây cầu), slowly (chậm chạp).

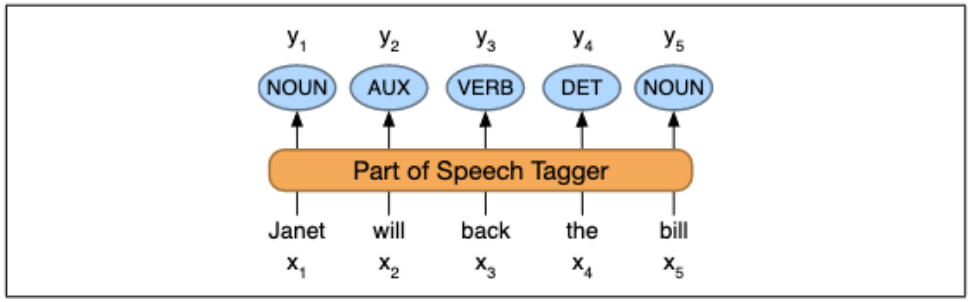
  - **Lớp từ đóng (Closed word classes):**
    - **Mạo từ, định từ, đại từ, giới từ, liên từ và thán từ**. Số lượng mỗi từ loại thuộc nhóm này chỉ từ vài từ đến vài chục từ và rất ít khi nhận thêm từ mới.
    - Gồm những từ chức năng (**function words)**: từ ít mang nghĩa nội dung nhưng lại đóng vai trò quan trọng trong quan hệ cú pháp của câu
      - on (ở trên), beside (bên cạnh), he (ông ấy), and (và).

Gán nhãn từ loại

- **Part-of speech tagging (POS-TAGGING)** là bài toán gán nhãn từ loại cho từng từ trong một văn bản.
- Tagging là một bài toán khử nhập nhằng nghĩa: một từ có thể có nhiều từ loại khác nhau. Mục tiêu bài toán là tìm ra từ loại đúng nhất cho một từ **trong một ngữ cảnh (context) cụ thể**.
- VD: từ loại cho từ "book"

book that flight ➔ book được gán nhãn từ loại là động từ

hand me that book. ➔ book được gán nhãn từ loại là danh từ



Vai trò của Gán nhãn từ loại

- Bài toán phân tích ngữ pháp gồm:
  - *Phân tích từ pháp*: xác định từ loại của các từ trong câu.
  - *Phân tích cú pháp*: Xây dựng nên cây cú pháp cho câu, hoặc tìm ra mối quan hệ giữa các thành phần trong câu.
- Xây dựng ứng dụng Xử lý ngôn ngữ

Từ loại trong tiếng Anh

	Tag	Description	Example
Open Class	**ADJ**	Adjective: noun modifiers describing properties	*red, young, awesome*
	ADV	Adverb: verb modifiers of time, place, manner	*very, slowly, home, yesterday*
	NOUN	words for persons, places, things, etc.	*algorithm, cat, mango, beauty*
	VERB	words for actions and processes	*draw, provide, go*
	PROPN	Proper noun: name of a person, organization, place, etc..	*Regina, IBM, Colorado*
	INTJ	Interjection: exclamation, greeting, yes/no response, etc.	*oh, um, yes, hello*
Closed Class Words	**ADP**	Adposition (Preposition/Postposition): marks a noun's spacial, temporal, or other relation	*in, on, by under*
	AUX	Auxiliary: helping verb marking tense, aspect, mood, etc.,	*can, may, should, are*
	CCONJ	Coordinating Conjunction: joins two phrases/clauses	*and, or, but*
	DET	Determiner: marks noun phrase properties	*a, an, the, this*
	NUM	Numeral	*one, two, first, second*
	PART	Particle: a preposition-like form used together with a verb	*up, down, on, off, in, out, at, by*
	PRON	Pronoun: a shorthand for referring to an entity or event	*she, who, I, others*
	SCONJ	Subordinating Conjunction: joins a main clause with a subordinate clause such as a sentential complement	*that, which*
Other	**PUNCT**	Punctuation	; , ()
	SYM	Symbols like $ or emoji	$, %
	X	Other	asdf, qwfg

Universal Dependencies tagset (Nivre et al., 2016)

Penn Tree bank

Tag	Description	Example	Tag	Description	Example	Tag	Description	Example
CC	coord. conj.	*and, but, or*	NNP	proper noun, sing.	*IBM*	TO	"to"	*to*
CD	cardinal number	*one, two*	NNPS	proper noun, plu.	*Carolinas*	UH	interjection	*ah, oops*
DT	determiner	*a, the*	NNS	noun, plural	*llamas*	VB	verb base	*eat*
EX	existential 'there'	*there*	PDT	predeterminer	*all, both*	VBD	verb past tense	*ate*
FW	foreign word	*mea culpa*	POS	possessive ending	*'s*	VBG	verb gerund	*eating*
IN	preposition/ subordin-conj	*of, in, by*	PRP	personal pronoun	*I, you, he*	VBN	verb past partici-ple	*eaten*
JJ	adjective	*yellow*	PRP$	possess. pronoun	*your, one's*	VBP	verb non-3sg-pr	*eat*
JJR	comparative adj	*bigger*	RB	adverb	*quickly*	VBZ	verb 3sg pres	*eats*
JJS	superlative adj	*wildest*	RBR	comparative adv	*faster*	WDT	wh-determ.	*which, that*
LS	list item marker	*1, 2, One*	RBS	superlatv. adv	*fastest*	WP	wh-pronoun	*what, who*
MD	modal	*can, should*	RP	particle	*up, off*	WP$	wh-possess.	*whose*
NN	sing or mass noun	*llama*	SYM	symbol	*+,%, &*	WRB	wh-adverb	*how, where*

Marcus, Mitchell, Beatrice Santorini, and Mary Ann Marcinkiewicz. "Building a large annotated corpus of English: The Penn Treebank." (1993).

Từ loại trong tiếng Việt

Định từ	**Danh từ**	nhà, đất, người, ...
	Đại từ	tao, nó, đây, đó, ...
	Động từ	đi, bò, ...
	Tính từ	đẹp, xấu, ...
	Lượng từ	các, những, ...
	Chỉ từ	này, kia, nọ, ...
Phó từ	**Tiền phó từ**	đã, sẽ, ...
	Hậu phó từ	rồi, hết, ra, ...
Kết từ	**Giới từ**	của, ...
	Liên từ	và, với, ...
tình thái từ	**Trợ từ**	cũng, nhưng, ...
	Tiểu từ	à, ôi, ...

Nguyễn Công Hồng, Về vấn đề phân định từ loại trong tiếng Việt. T/c Ngôn ngữ số 2, 2003

Một số bộ dữ liệu cho Gán nhãn từ loại

- Tiếng Anh:
  - Universal Dependencies tagset (Nivre et al., 2016)
  - Penn Treebank P.O.S. Tags (Marcus et al., 1993)
- Tiếng Việt:
  - VLSP 2013 dataset for Word segmentation and POS Tagging (https://vlsp.org.vn/resources-vlsp2013)
  - Bộ dữ liệu NII-VTB (https://github.com/mynlp/niivtb)

Bài toán Nhận diện thực thể

- **Thực thể có tên (Named entity)** là các thực thể đề cập đến trong văn bản, thường là tên riêng.
  - Các danh từ riêng này có thể là: người (person - PER), địa điểm (location - LOC), tổ chức (organization - ORG), hoặc là thực thể địa chính trị (geo-political entity - GPE).

- **Bài toán xác định tự động các thực thể có tên** trong văn bản được gọi là bài toán Nhận diện thực thể (**named-entity recognition -** NER).

Một số thực thể có tên trong tiếng Anh

Type	Tag	Sample Categories	Example sentences
People	PER	people, characters	**Turing** is a giant of computer science.
Organization	ORG	companies, sports teams	The **IPCC** warned about the cyclone.
Location	LOC	regions, mountains, seas	**Mt. Sanitas** is in **Sunshine Canyon**.
Geo-Political Entity	GPE	countries, states	**Palo Alto** is raising the fees for parking.

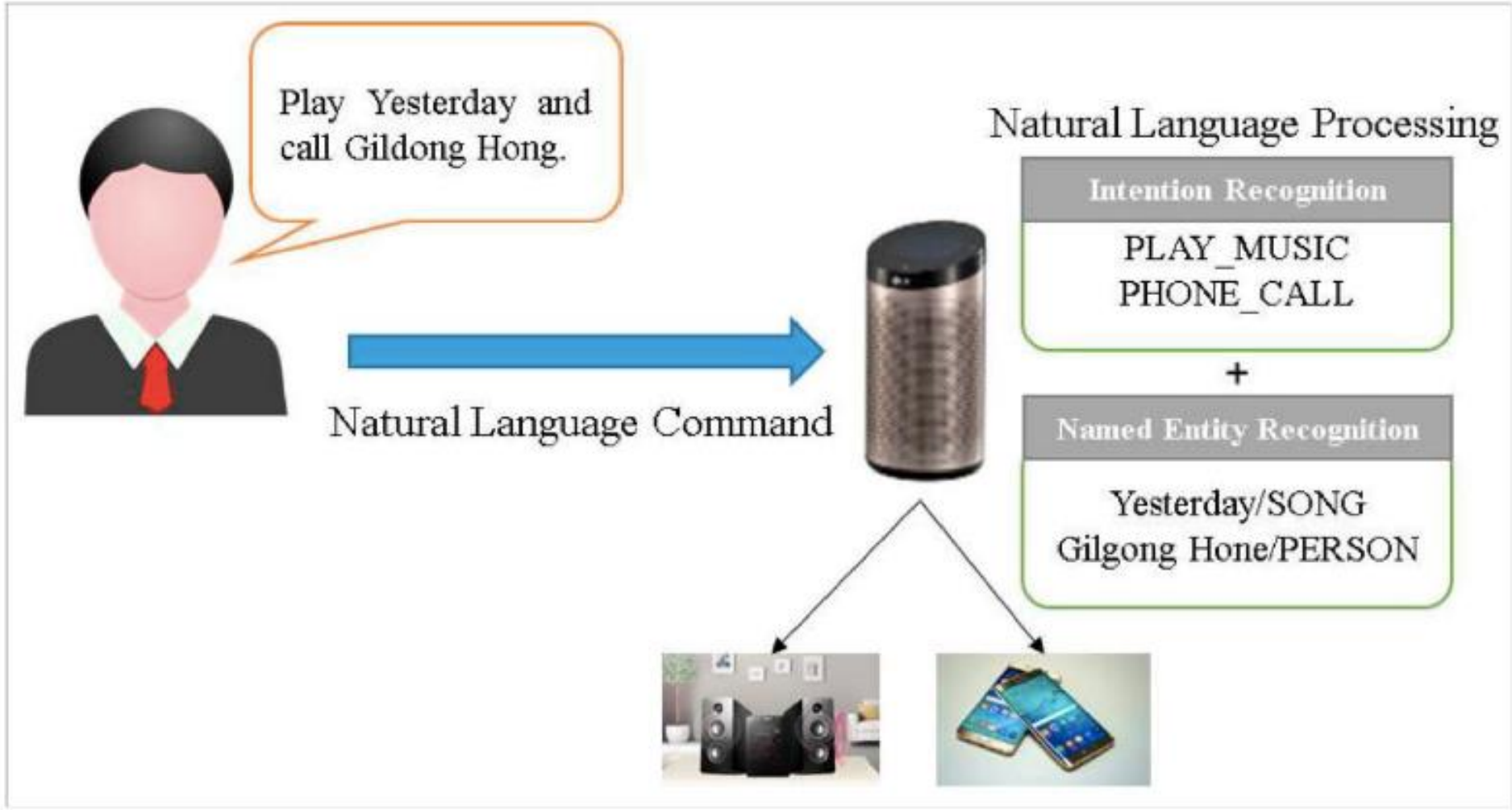
Ví dụ về nhận dạng thực thể tên trong văn bản

Citing high fuel prices, [ORG **United Airlines**] said [TIME **Friday**] it has increased fares by [MONEY **$6**] per round trip on flights to some cities also served by lower-cost carriers. [ORG **American Airlines**], a unit of [ORG **AMR Corp.**], immediately matched the move, spokesman [PER **Tim Wagner**] said. [ORG **United**], a unit of [ORG **UAL Corp.**], said the increase took effect [TIME **Thursday**] and applies to most routes where it competes against discount carriers, such as [LOC **Chicago**] to [LOC **Dallas**] and [LOC **Denver**] to [LOC **San Francisco**].

Vai trò của NER

- **Trong sentiment analysis**: nhận diện thực thể tên giúp cho việc phân tích ý kiến của người dùng đối với một đối tượng cụ thể (VD: sản phẩm, hàng hoá, dịch vụ, nhân viên, ...)
- **Trong question-answering**: nhận diện thực thể có tên giúp cho việc trích xuất các thông tin về các sự kiện (event) được đề cập trong câu hỏi hoặc câu trả lời và mối liên kết giữa các sự kiện đó với thông tin cần truy xuất để tìm ra câu trả lời chính xác.
- Và các ứng dụng khác.

Natural language understanding



Khó khăn đối với NER

- Vấn đề nhận diện thực thể cho một spans of text ➔ **phụ thuộc vào chất lượng tách từ/tách cụm**.
  - Spans of text: một hợp các từ trong một khoảng nhất định trong văn bản.
  - VD: [John F. Kennedy] ➔ spans gồm các từ: “John”, “F.”, “Kennedy”
- Sự **nhập nhằng về ngữ nghĩa** (type ambiguity).

  VD: từ **JFK** có thể là:
  - Tên một người (PER): John F. Kennedy (tổng thống thứ 35 của Mỹ).
  - Tên một địa điểm (LOC): sân bay JFK ở TP New York.
  - Tên một trường trung học (ORG).
  - Tên một con đường (ORG).
  - ....

Ví dụ về type ambiguity trong bài toán NER

[PER Washington] was born into slavery on the farm of James Burroughs.
[ORG Washington] went up 2 games to 1 in the four-game series.
Blair arrived in [LOC Washington] for what may well be his last state visit.
In June, [GPE Washington] passed a primary seatbelt law.

Biểu diễn BIO trong NER

- Biểu diễn đầu ra kiểu **B**IO trong NER: một thực thể tên riêng được output có 2 thành phần:
  - ***Loại thực thể*** *(named entity types).*
  - ***Phạm vi*** *(boundary).*
- Cách đánh nhãn tên thực thể theo cách tiếp cận BIO:
  - *Token bắt đầu của thực thể (span) được ký hiệu là B.*
  - *Token nằm trong thực thể được ký hiệu là I.*
  - *Token nằm ngoài các thực thể được ký hiệu là O.*
- Ứng với mỗi loại thực thể ta phải có 2 nhãn B và I.
  - Như vậy, với n loại thực thể có tên ban đầu, theo cách tiếp cận của BIO tagging, chúng ta sẽ có tổng cộng **2n+1** nhãn thực thể tên.

Các dạng biểu diễn khác

- IO: giống như BIO, nhưng bỏ đi ký hiệu B (boundary).
- BIOES: Giống như BIO tagging, nhưng thêm vào ký hiệu E để ký hiệu cho token kết thúc spans, và ký hiệu S để chỉ spans chỉ có một ký tự.

Ví dụ về gán nhãn thực thể

Text: *Jane Villanueva of United Airlines Holding discussed the Chicago route*.

Words	IO Label	BIO Label	BIOES Label
Jane	I-PER	B-PER	B-PER
Villanueva	I-PER	I-PER	E-PER
of	O	O	O
United	I-ORG	B-ORG	B-ORG
Airlines	I-ORG	I-ORG	I-ORG
Holding	I-ORG	I-ORG	E-ORG
discussed	O	O	O
the	O	O	O
Chicago	I-LOC	B-LOC	S-LOC
route	O	O	O
.	O	O	O

Một số bộ dữ liệu cho NER

- Tiếng Anh:
  - CoNLL-2002 and CoNLL-2003 (British newswire).
  - ACE.
  - MUC-6 and MUC-7 (American newswire)
- Tiếng Việt:
  - VLSP 2016 (https://vlsp.org.vn/resources-vlsp2016).

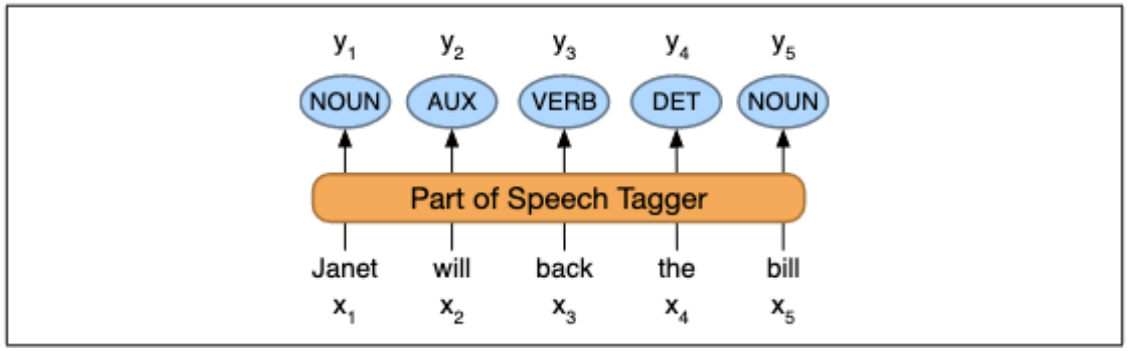
Mô hình HMM

Bài toán Gán nhãn chuỗi

Phương pháp dùng cho POS-Tagging và NER

- HMM – Hidden Markov chains.
- CRF – Conditional Random Fields.

…



- So sánh HMM và các phương pháp khác:

[Spoustov´a, D.j., Haji˘c, J., Raab, J., Spousta, M.: Semi-supervised training for the averaged perceptron POS tagger. In: Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009). (2009) 763–77] https://www.aclweb.org/anthology/E09-1087.pdf

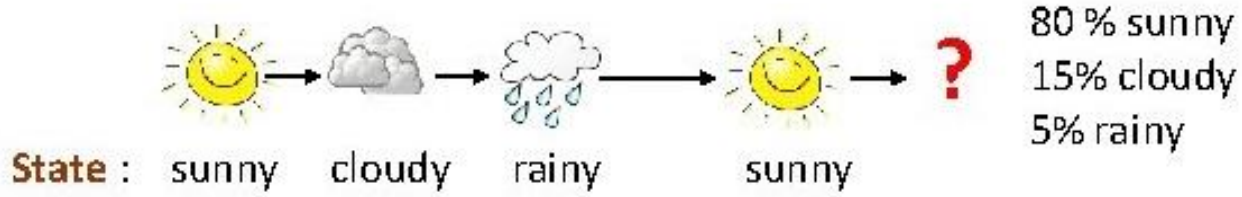
Giới thiệu HMM

- Mô hình Markov Ẩn - HMM là một **mô hình thống kê** với các tham số không biết trước, phải xác định từ các tham số đã biết.
- Mô hình HMM mở rộng từ mô hình toán học: Chuỗi Markov.

- **Ứng dụng:**
  - Gán nhãn chuỗi trong XLNNTN: NER, POS tagging
  - Nhận dạng tiếng nói
  - Nhận dạng chữ viết (OCR)
  - Tin sinh học: Dự đoán các vùng mang mã trên một trình tự gene. xác định các họ gene hoặc họ protein, mô phỏng cấu trúc không gian của protein từ trình tự amino acid.

Markov chains (Chuỗi Markov/Quá trình Markov)

- **Chuỗi Markov** (Markov chain) là một mô hình mô tả cho chúng ta biết xác suất của các chuỗi biến cố ngẫu nhiên (các trạng thái).
  - Chuỗi trạng thái **có thứ tự trước sau** theo thời gian.
  - Trạng thái có thể nhận giá trị từ một **tập hợp giá trị trạng thái** (có thể là tập hợp các từ, các nhãn, hoặc bất kỳ tập giá trị rời rạc nào).
  - **Giả định Markov** (assumption): Để dự đoán



Andrei Andreyevich Markov
(1856 - 1922)

Giả định Markov

- Khi dự đoán trạng thái tương lai chỉ dựa vào trạng thái hiện tại, không dựa vào quá khứ.

VD: dự báo thời tiết của ngày mai thì dựa vào trạng thái thời tiết của ngày hiện tại, không dựa vào thời tiết của các ngày trước đó

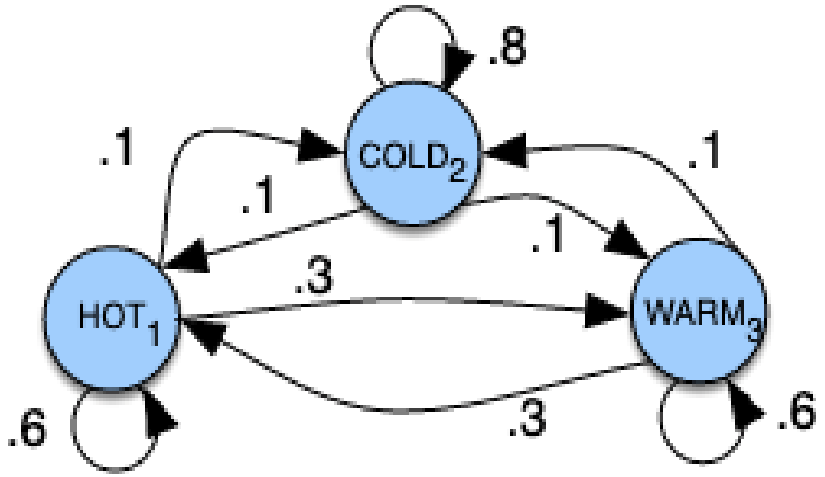
Markov Assumption: $P(q_i = a|q_1 ... q_{i-1}) = P(q_i = a|q_{i-1})$

Các thành phần của mô hình Markov

- Q: Tập các trạng thái (states)
- A: ma trận xác suất chuyển đổi. Mỗi phần tử trong ma trận a_{ij} cho biết xác suất chuyển đổi từ trạng thái i sang trạng thái j.
- π: phân phối trạng thái ban đầu: π_i là xác suất chuỗi bắt đầu bởi trạng thái thứ i.

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} a_{12} \ldots a_{N1} \ldots a_{NN}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
$\pi = \pi_1, \pi_2, \ldots, \pi_N$	an **initial probability distribution** over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

Ví dụ 1



(a)

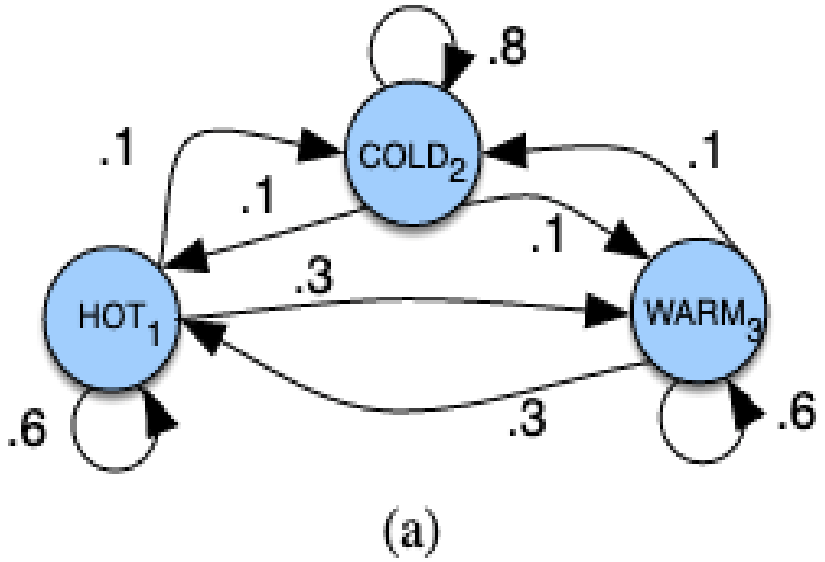
Chuỗi Markov ở hình (a):

- Tập các trạng thái **Q** = {HOT, COLD, WARM} (**N**=3)
- Ma trận xác suất chuyển đổi trạng thái **A**:

	π	HOT	COLD	WARM
HOT	0.7	0.6	0.1	0.3
COLD	0.1	0.3	0.8	0.1
WARM	0.2	0.3	0.1	0.6

- Phân phối trạng thái ban đầu $\boldsymbol{\pi}$ = [0.7, 0.1, 0.2]

Ví dụ 1



(a)

• Tính xác suất các chuỗi sau:

(1) hot hot hot hot

(2) cold hot cold hot

Chuỗi Markov ở hình (a):

- Tập các trạng thái **Q** = {HOT, COLD, WARM} (**N**=3)
- Ma trận xác suất chuyển đổi trạng thái **A**:

	π	HOT	COLD	WARM
HOT	0.7	0.6	0.1	0.3
COLD	0.1	0.3	0.8	0.1
WARM	0.2	0.3	0.1	0.6

- Phân phối trạng thái ban đầu $\boldsymbol{\pi}$ = [0.7, 0.1, 0.2]

Ví dụ 2

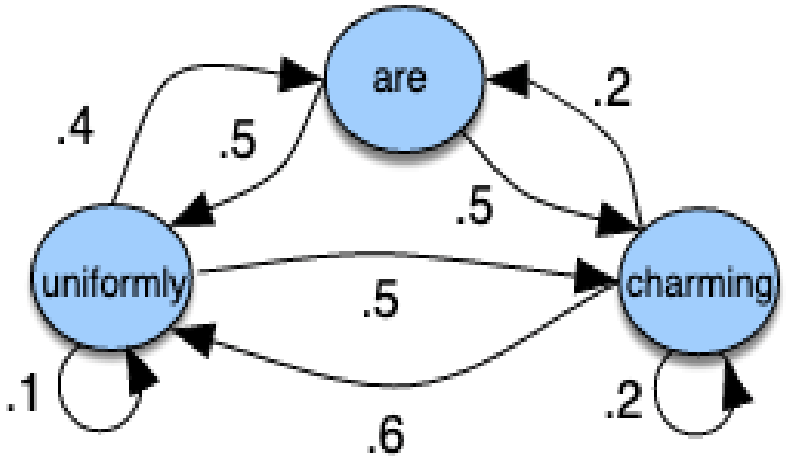
Chuỗi Markov ở hình (b):

Q = {uniformly, are, charming}

π = [0.7, 0.1, 0.2]

Hãy:

1. Xây dựng ma trận xác suất chuyển đổi A.



(b)

2. Tính:

P(are|uniformly) = ?

P(are|charming) = ?

Ví dụ 2

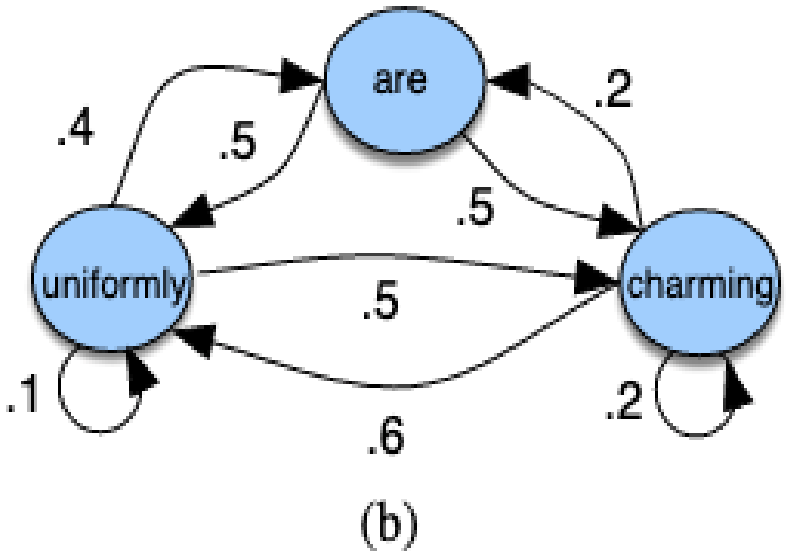
Chuỗi Markov ở hình (b):

Q = {uniformly, are, charming}

π = [0.7, 0.1, 0.2]

Hãy:

1. Xây dựng ma trận xác suất chuyển đổi A.



(b)

2. Tính:

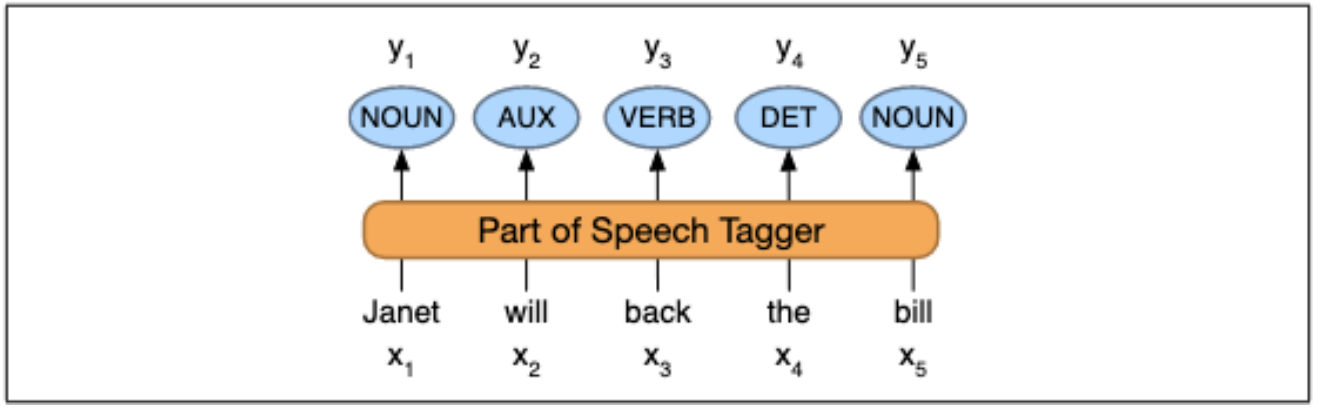
P(are|uniformly) = 0.4

P(are|charming) = 0.2

	π	uniformly	are	charming
uniformly	0.7	0.1	0.4	0.5
are	0.1	0.5	0.1	0.5
charming	0.2	0.6	0.2	0.2

Mô hình Markov ẩn

- Hidden Markov Model (HMM) là một mô hình dựa trên chuỗi Markov cho phép biểu diễn cho các sự kiện không thể quan sát trực tiếp (ẩn).
- Các sự kiện không thể quan sát trực tiếp:
  - Từ loại
  - Loại thực thể



Mô hình Markov ẩn bậc 1 (first order HMM)

- Q: tập trạng thái ẩn.
- A: ma trận xác suất chuyển đổi (trạng thái ẩn).
- O: chuỗi các trạng thái quan sát được (observation).
- B: chuỗi xác suất thể hiện khả năng một trạng thái quan sát được o được sinh ra từ trạng thái ẩn q (**emission probability**), còn gọi là xác suất phát xạ.
- π: phân phối trạng thái ẩn ban đầu. π_i là xác suất chuỗi bắt đầu bởi trạng thái ẩn thứ i.

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} \ldots a_{ij} \ldots a_{NN}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{N} a_{ij} = 1 \quad \forall i$
$O = o_1 o_2 \ldots o_T$	a sequence of T **observations**, each one drawn from a vocabulary $V = v_1, v_2, \ldots, v_V$
$B = b_i(o_t)$	a sequence of **observation likelihoods**, also called **emission probabilities**, each expressing the probability of an observation o_t being generated from a state q_i
$\pi = \pi_1, \pi_2, \ldots, \pi_N$	an **initial probability distribution** over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

Hai giả định của mô hình Markov ẩn bậc 1

1. Xác suất của 1 trạng thái ẩn chỉ phụ thuộc trạng thái ẩn liền trước.
2. Xác suất của 1 trạng thái quan sát được chỉ phụ thuộc vào 1 trạng thái ẩn sinh ra nó, mà không phụ thuộc các trạng thái khác (cả ẩn và quan sát được).

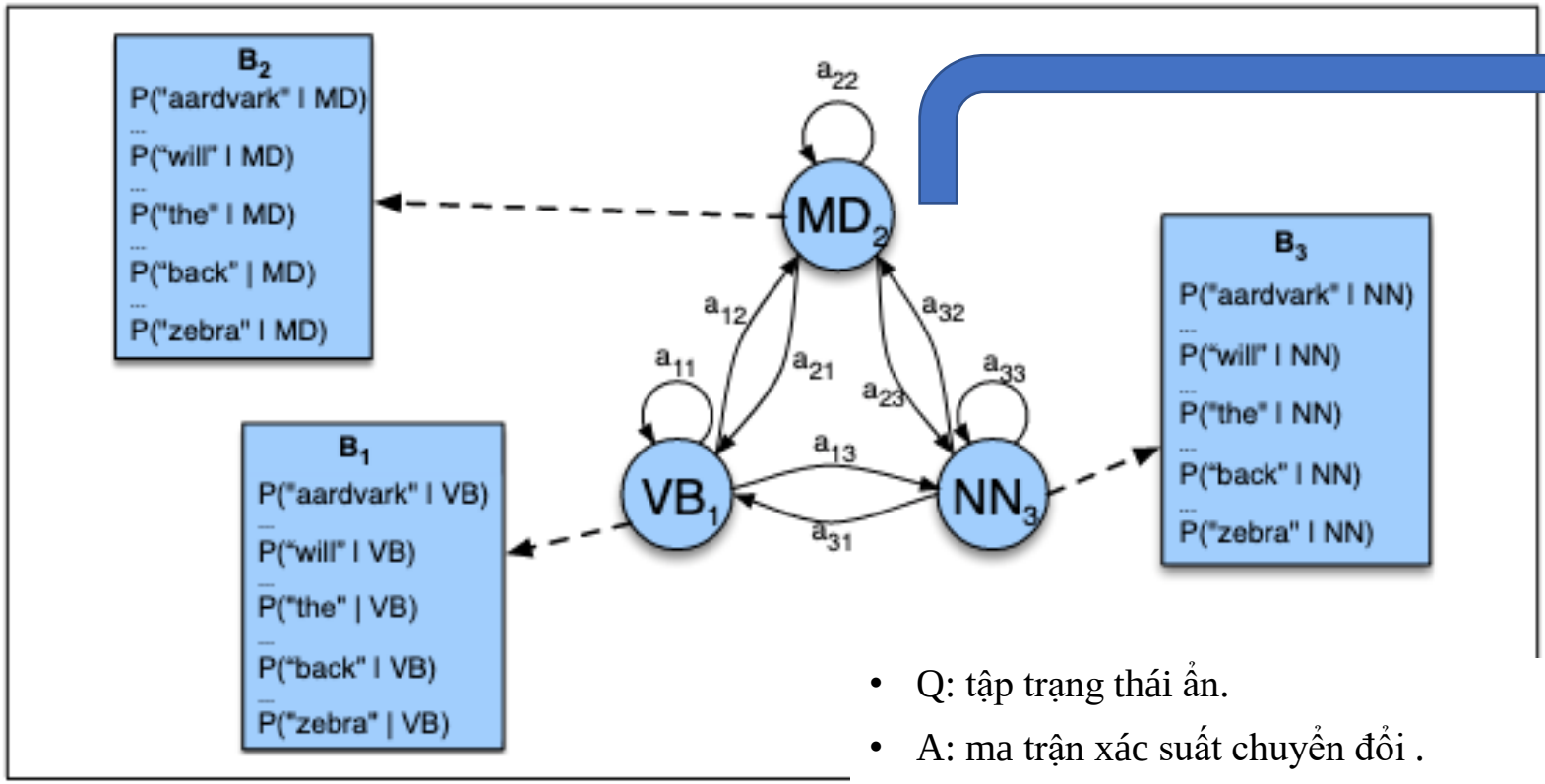
HMM Tagger

HMM Tagger

- Q: tập trạng thái ẩn.
- A: ma trận xác suất chuyển đổi .
- O: chuỗi các trạng thái quan sát được (observation).
- B: chuỗi xác suất thể hiện khả năng một trạng thái quan sát được o được sinh ra từ trạng thái ẩn q (**emission probability**)
- π: phân phối trạng thái ban đầu. π_i là xác suất chuỗi bắt đầu bởi trạng thái thứ i.

- Là một mô hình xử lý trong XLNNTN dựa trên HMM, dùng để gán nhãn cho các phần tử trong chuỗi.
- HMM Tagger gồm 2 thành phần chính:
  - **A: xác suất xuất hiện của một nhãn dựa trên nhãn xuất hiện trước nó**.
  - **B: xác suất xuất hiện của một từ** theo một nhãn.

Ví dụ



A

	VB	MD	NN
VB	a_{11}	a_{12}	a_{13}
MD	a_{21}	a_{22}	a_{23}
NN	a_{31}	a_{32}	a_{33}

- Q: tập trạng thái ẩn.
- A: ma trận xác suất chuyển đổi .
- O: chuỗi các trạng thái quan sát được (observation).
- B: chuỗi xác suất thể hiện khả năng một trạng thái quan sát được o được sinh ra từ trạng thái ẩn q (**emission probability**)
- π: phân phối trạng thái ban đầu. π_i là xác suất chuỗi bắt đầu bởi trạng thái thứ i.

Xác suất xuất hiện của một nhãn (A)

$$P(t_i|t_{i-1}) = \frac{count(t_{i-1}, t_i)}{count(t_{i-1})}$$

Ví dụ: Trong bộ dữ liệu WSJ:

- Nhãn MD xuất hiện 13124 lần
- Nhãn MD xuất hiện cùng với nhãn VB là 10471 lần
- xác suất chuyển đổi từ nhãn MD sang nhãn VB là:

$$P(VB|MD) = \frac{count(MD, VB)}{count(MD)} = \frac{10471}{13124} = 0.8$$

Xác suất xuất hiện của một từ theo một nhãn (B)

$$P(w_i|t_i) = \frac{count(t_i, w_i)}{count(t_i)}$$

Ví dụ: Trong bộ dữ liệu WSJ:

từ “will” xuất hiện với vai trò modal verb (WD) là 4046 lần.

Nhãn MD xuất hiện cùng với nhãn VB là 10471 lần

$$P(will|MD) = \frac{count(MD, will)}{count(MD)} = \frac{4046}{10471} = 0.31$$

HMM Decoding (Decoding = Predict)

- Giải mã (Decode): Là tác vụ **xác định giá trị chuỗi trạng thái ẩn** dựa trên dữ liệu trạng thái quan sát được.
- Trong HMM Tagger: giải mã là xác định chuỗi từ loại của một câu.

- Input:

  HMM: $\lambda = (A, B)$

  $O = o_1, o_2, ..., o_n$ hay trong XLNNTN: $S = w_1, w_2, ..., w_n$
- Output:

  Chuỗi nhãn có khả năng xuất hiện cao nhất

  $Q = q_1, q_2, ..., q_n$ hay trong XLNNTN: $Q = t_1, t_2, ..., t_n$

HMM Decoding method

1. Từ giả định 1 của HMM - Xác suất xuất hiện của một từ chỉ phụ thuộc vào từ loại sinh ra nó, và độc lập với những từ lân cận và nhãn của các từ lân cận.

$$P(w_1 \dots w_n | t_1 \dots t_n) \approx \prod_{i=1}^{n} P(w_i | t_i)$$

2. Từ giả định 2 của HMM - Xác suất xuất hiện của một nhãn chỉ phụ thuộc vào một nhãn trước nó, và độc lập với các nhãn còn lại, ta có:

$$P(t_1 \dots t_n) \approx \prod_{i=1}^{n} P(t_i | t_{i-1})$$

3. Kết hợp 1 và 2, ta tìm ra chuỗi nhãn có khả năng cao nhất:

$$\hat{t}_{1:n} = \underset{t_1 \dots t_n}{\operatorname{argmax}} P(t_1 \dots t_n | w_1 \dots w_n) \approx \underset{t_1 \dots t_n}{\operatorname{argmax}} \prod_{i=1}^{n} \overbrace{P(w_i | t_i)}^{\text{emission}} \overbrace{P(t_i | t_{i-1})}^{\text{transition}}$$

Cài đặt HMM Decoding? Thuật toán Viterbi

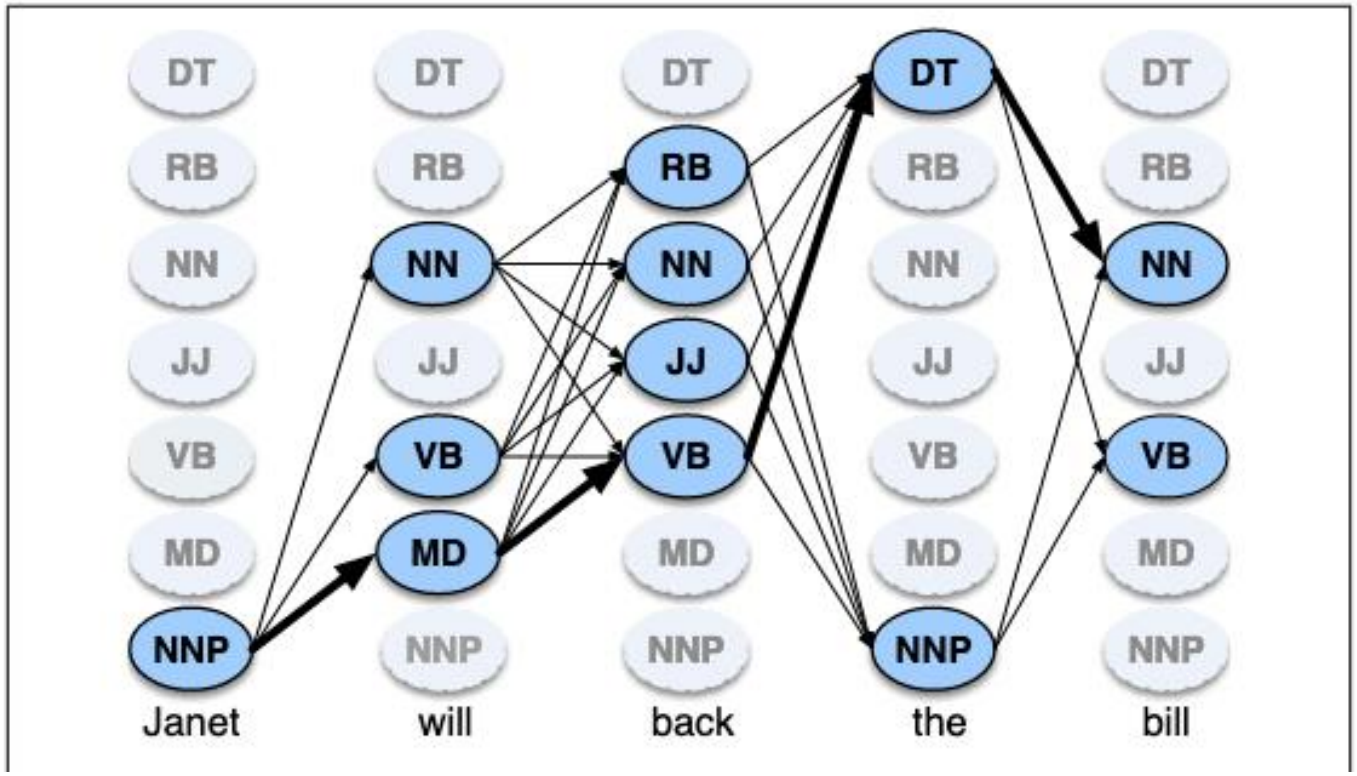
- Mục tiêu: ước tính xác suất **tối đa** của chuỗi trạng thái ẩn có khả năng xảy ra nhất, được gọi là đường dẫn Viterbi, dẫn đến một chuỗi các sự kiện được quan sát.
- Công cụ sử dụng: **minimum edit distance**.

Ví dụ

- Văn bản: *Janet will back the bill.*
- Xác suất đầu ra mong muốn: NNP – MD – VB – DT – NN

Biểu diễn cho thuật toán Viterbi

- Sử dụng một “lattice” để biểu diễn:
  - Mỗi cột biểu diễn cho một từ trong câu - trạng thái quan sát được o.
  - Mỗi dòng biểu diễn cho một trạng thái s.



Thuật toán Viterbi (mã giả)

function VITERBI(*observations* of len *T*,*state-graph* of len *N*) **returns** *best-path*, *path-prob*

create a path probability matrix *viterbi[N,T]*
for each state *s* **from** 1 **to** *N* **do** ; initialization step
    $viterbi[s,1] \leftarrow \pi_s * b_s(o_1)$
    backpointer[s,1] $\leftarrow 0$
for each time step *t* **from** 2 **to** *T* **do** ; recursion step
  for each state *s* **from** 1 **to** *N* **do**
    $viterbi[s,t] \leftarrow \max_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$
    $backpointer[s,t] \leftarrow \operatorname{argmax}_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$
$bestpathprob \leftarrow \max_{s=1}^{N} viterbi[s,T]$; termination step
$bestpathpointer \leftarrow \operatorname{argmax}_{s=1}^{N} viterbi[s,T]$; termination step
bestpath $\leftarrow$ the path starting at state *bestpathpointer*, that follows backpointer[] to states back in time
return *bestpath*, *bestpathprob*

Ví dụ

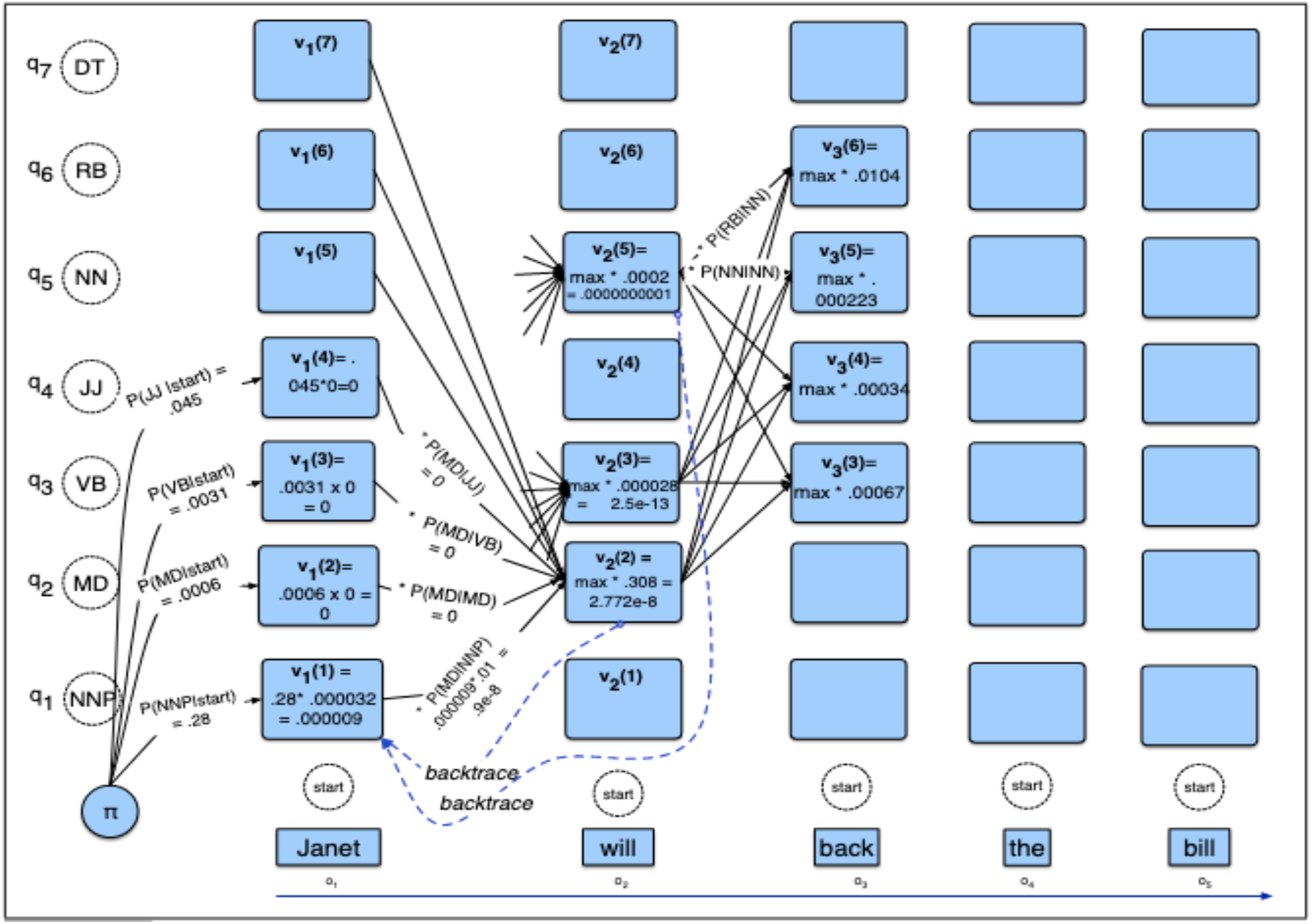
	NNP	MD	VB	JJ	NN	RB	DT
<*s*>	0.2767	0.0006	0.0031	0.0453	0.0449	0.0510	0.2026
NNP	0.3777	0.0110	0.0009	0.0084	0.0584	0.0090	0.0025
MD	0.0008	0.0002	0.7968	0.0005	0.0008	0.1698	0.0041
VB	0.0322	0.0005	0.0050	0.0837	0.0615	0.0514	0.2231
JJ	0.0366	0.0004	0.0001	0.0733	0.4509	0.0036	0.0036
NN	0.0096	0.0176	0.0014	0.0086	0.1216	0.0177	0.0068
RB	0.0068	0.0102	0.1011	0.1012	0.0120	0.0728	0.0479
DT	0.1147	0.0021	0.0002	0.2157	0.4744	0.0102	0.0017

xác suất chuyển đổi ứng với từng nhãn (tag)

	Janet	will	back	the	bill
NNP	0.000032	0	0	0.000048	0
MD	0	0.308431	0	0	0
VB	0	0.000028	0.000672	0	0.000028
JJ	0	0	0.000340	0	0
NN	0	0.000200	0.000223	0	0.002337
RB	0	0	0.010446	0	0
DT	0	0	0	0.506099	0

Xác suất từ w xuất hiện khi biết nhãn t

Các xác suất trên được tính dựa trên bộ dữ liệu WSJ.


q7 DT
q6 RB
q5 NN
q4 JJ
q3 VB
q2 MD
q1 NNP
π
v1(7)
v1(6)
v1(5)
v1(4)= . 045*0=0
v1(3)= .0031 x 0 = 0
v1(2)= .0006 x 0 = 0
v1(1) = .28* .000032 = .000009
v2(7)
v2(6)
v2(5)= max * .0002 = .0000000001
v2(4)
v2(3)= max * .000028 = 2.5e-13
v2(2) = max * .308 = 2.772e-8
v2(1)
v3(6)= max * .0104
v3(5)= max * . 000223
v3(4)= max * .00034
v3(3)= max * .00067
P(JJ |start) = .045
P(VB|start) = .0031
P(MD|start) = .0006
P(NNP|start) = .28
* P(MD|JJ) = 0
* P(MD|VB) = 0
* P(MD|MD) = 0
* P(MD|NNP) .000009* .01 = .9e-8
* P(RB|NN)
* P(NN|NN)
backtrace
backtrace
start
start
start
start
start
Janet
will
back
the
bill
o1
o2
o3
o4
o5

Mô hình CRF

Bài toán Gán nhãn chuỗi

Conditional Random Fields (CRF)

- Input: chuỗi văn bản đầu vào $X = x_1 .... x_n$ có độ dài là n.
- Output: chuỗi đầu ra $Y = y_1 ... y_n$.

- Xác suất của đầu ra Y khi biết X là:

$$\begin{aligned}\hat{Y} &= \underset{Y}{\operatorname{argmax}}\, p(Y|X) \\ &= \underset{Y}{\operatorname{argmax}}\, p(X|Y)p(Y) \\ &= \underset{Y}{\operatorname{argmax}} \prod_i p(x_i|y_i) \prod_i p(y_i|y_{i-1})\end{aligned}$$

Trong HMM, ta tính P(Y|X) dựa vào công thức likelihood

Tính P(Y|X)

- CRF tính xác suất của tất cả các chuỗi nhãn Y khả dĩ khi biết X.
- Trong CRF, hàm F ánh xạ cả chuỗi các từ X (input) và chuỗi nhãn Y (output) thành 1 vector đặc trưng (feature vector).
  - Giả sử chúng ta có K đặc trưng thì có K trọng số w_k của mỗi đặc trưng F_k.

$$p(Y|X) = \frac{\exp\left(\sum_{k=1}^{K} w_k F_k(X,Y)\right)}{\sum_{Y' \in \mathcal{Y}} \exp\left(\sum_{k=1}^{K} w_k F_k(X,Y')\right)}$$

Chuẩn hóa

$$p(Y|X) = \frac{1}{Z(X)} \exp\left(\sum_{k=1}^{K} w_k F_k(X,Y)\right)$$

$$Z(X) = \sum_{Y' \in \mathcal{Y}} \exp\left(\sum_{k=1}^{K} w_k F_k(X,Y')\right)$$

Tính F_k

- F_k được gọi là đặc trưng toàn cục (**Global features**), là tổng của các đặc trưng cục bộ (local features) ứng với mỗi vị trí I trong Y.

$$F_k(X,Y) = \sum_{i=1}^{n} f_k(y_{i-1}, y_i, X, i)$$

- Mỗi đặc trưng cục bộ f_k khai thác thông tin của: nhãn output hiện tại $\mathbf{y_i}$, nhãn output của từ đứng liền trước $\mathbf{y_{i-1}}$, toàn bộ chuỗi (hoặc một phần chuỗi) và vị trí hiện tại i.

Xây dựng các **đặc trưng** cho bài toán POS Tagging

- Templates cho đặc trưng: $\langle y_i, x_i \rangle, \langle y_i, y_{i-1} \rangle, \langle y_i, x_{i-1}, x_{i+2} \rangle$

VD: Janet/NNP will/MD back/VB the/DT bill/NN, xi = “back”

<u>Các đặc trưng:</u>

f_{3743}: y_i = VB and x_i = back
f_{156}: y_i = VB and y_{i-1} = MD
f_{99732}: y_i = VB and x_{i-1} = will and x_{i+2} = bill

Xây dựng features cho bài toán POS Tagging (tt)

- Dạng từ (Word shapes): Chuyển từ những "từ" (word) cụ thể sang dạng biểu diễn đơn giản hóa chứa đặc trưng về: chiều dài, viết hoa thường, chữ số, ký tự Greek, các dấu đặc biệt.

Varicella-zoster	Xx-xxx
mRNA	xXXX
CPA1	XXXd

Xây dựng features cho bài toán NER

- Sử dụng Tự điển địa lý (**gazetteer)** và Danh sách tên có sẵn.

Words	POS	Short shape	Gazetteer	BIO Label
Jane	NNP	Xx	0	B-PER
Villanueva	NNP	Xx	1	I-PER
of	IN	x	0	O
United	NNP	Xx	0	B-ORG
Airlines	NNP	Xx	0	I-ORG
Holding	NNP	Xx	0	I-ORG
discussed	VBD	x	0	O
the	DT	x	0	O
Chicago	NNP	Xx	1	B-LOC
route	NN	x	0	O
.	.	.	0	O

Độ đo đánh giá

- Đối với bài toán POS Tagging: độ đo chuẩn là *Accuracy*.
- Đối với bài toán NER recognition: độ đo *recall*, *precision và F1-score*.

Các tình huống của output NER

A. exact match

1) Phạm vi và loại thực thể đều được xác định đúng.
2) Thực thể không tồn tại trong ground truth (không được gán nhãn).
3) Hệ thống sót thực thể.

B. partial match (overlapping)

1) Loại thực thể sai dù phạm vi có thể đúng (correct entity boundary, type disagree)
2) Phạm vi sai (boundary overlap)
3) Phạm vi sai và loại thực thể cũng sai

Tính toán precision và recall cho NER

- Trong CoNLL 2003 shared task:
  - *Precision is the percentage of named entities found by the learning system that are correct.*
  - *Recall is the percentage of named entities present in the corpus that are found by the system.*
  - *A named entity is correct only if it is an **exact match** of the corresponding entity in the data file.*
- Trong cuộc thi SemEval 2013 Task 9: Sử dụng độ đo MUC.

CoNLL 2003 Shared task: https://www.aclweb.org/anthology/W03-0419/
SemEval 2013 Shared task: https://www.aclweb.org/anthology/S13-2056.pdf

Độ đo MUC-5

Các loại dự đoán:

- **Correct (COR)**: match
- **Incorrect (INC):** not match
- **Partial (PAR):** predicted entity boundary overlap with golden annotation, but they are not the same.
- **Missing (MIS):** golden annotation boundary is not identified (predictions do not have, but golden label do)
- **Spurius (SPU):** predicted entity boundary does not exist in golden annotation (predictions have, but golden label do not)

https://www.aclweb.org/anthology/M93-1007.pdf

Gold-standard and predicted annotations

- Gold-standard (POS):

$$\text{POSSIBLE}(POS) = COR + INC + PAR + MIS = TP + FN$$

- Predicted by NER system

$$\text{ACTUAL}(ACT) = COR + INC + PAR + SPU = TP + FP$$

Tính toán precision/recall/f1 score

Exact match (*strict* and *exact*)

$$\text{Precision} = \frac{COR}{ACT} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{COR}{POS} = \frac{TP}{TP+FN}$$

Partial match (*partial* and *type*)

$$\text{Precision} = \frac{COR + 0.5 \times PAR}{ACT} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{COR + 0.5 \times PAR}{POS} = \frac{COR}{ACT} = \frac{TP}{TP+FP}$$

$$\text{F1 Score} = \frac{2 \text{ x (Precision x Recall)}}{\text{Precision + Recall}}$$

Ví dụ

Scenario	Golden Standard		System Prediction		Evaluation Schema			
	Entity Type	Surface String	Entity Type	Surface String	Type	Partial	Exact	Strict
III	brand	TIKOSYN			MIS	MIS	MIS	MIS
II			brand	healthy	SPU	SPU	SPU	SPU
V	drug	warfarin	drug	of warfarin	COR	PAR	INC	INC
IV	drug	propranolol	brand	propranolol	INC	COR	COR	INC
I	drug	phenytoin	drug	phenytoin	COR	COR	COR	COR
I	Drug	theophylline	drug	theophylline	COR	COR	COR	COR
VI	group	contraceptives	drug	oral contraceptives	INC	PAR	INC	INC

Tính precision/recall và F1 score

Measure	Type	Partial	Exact	Strict
Correct	3	3	3	2
Incorrect	2	0	2	3
Partial	0	2	0	0
Missed	1	1	1	1
Spurius	1	1	1	1
Precision	0.5	0.66	0.5	0.33
Recall	0.5	0.66	0.5	0.33
F1	0.5	0.66	0.5	0.33

Tổng kết

- Định nghĩa bài toán Sequence labeling: cho một chuỗi (sequence) X, tìm các nhãn (tags) ứng với từng phần tử trong chuỗi X.
- Hai bài toán cụ thể:
  1. Gán nhãn từ loại (Part of speech tagging – POS tagging)
  2. Nhận diện thực thể có tên (Named enity recognition - NER)
- Các phương pháp tiếp cận cho bài toán:
  1. HMM
  2. CRF
- Các độ đo đánh giá:
  1. POS Tagging: độ đo Accuracy
  2. NER: precision, recall và F1 score.