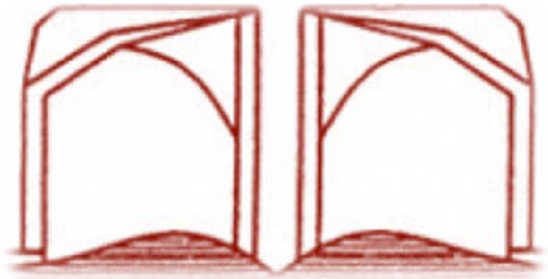


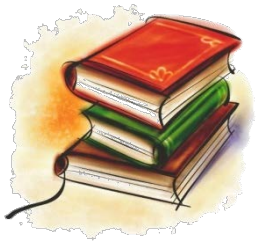
روش‌های یادگیری ماشین در پردازش زبان طبیعی

مدل مخفی مارکوف (HMM)

هادی ویسی

h.veisi@ut.ac.ir

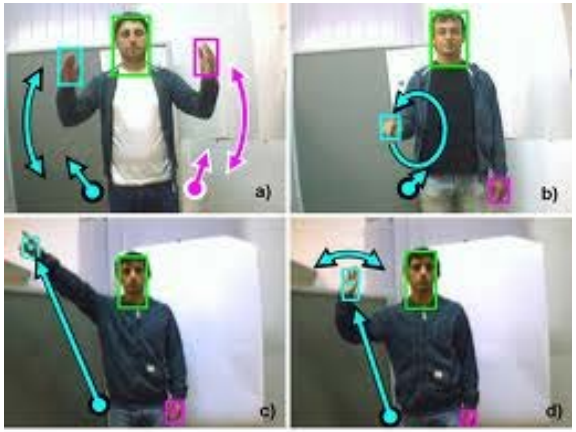
دانشگاه تهران – دانشکده علوم و فنون نوین



فهرست

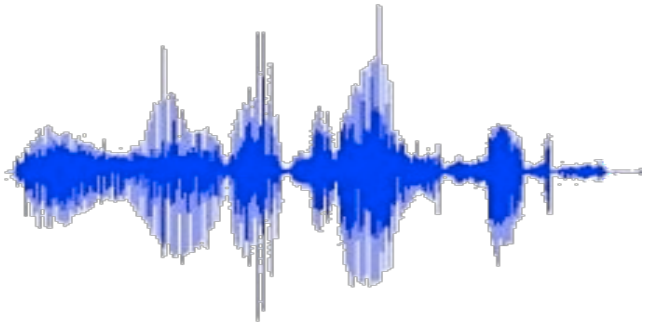
مدل مخفی مارکوف: معرفی . . .

- مدل مخفی مارکوف (HMM: Hidden Markov Model)
  - روشی برای مدل‌سازی داده‌های ترتیبی (sequential data)



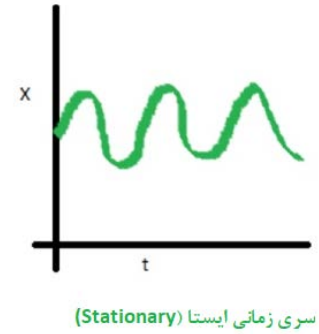
- داده‌های ترتیبی
  - نرخ ارز در روزهای مختلف
  - نمونه‌های پشت سرهم سیگنال گفتار، دست‌نوشته، امضا
  - دنباله نویسه‌ها یا واژه‌ها در یک متن
  - دنباله تصاویر پشت سر هم (ویدئو): حرکات انسان
  - میزان بارش باران در روزهای متوالی سال

مفهوم ایران بزرگ از جهات گوناگون ریشه در تاریخ چند هزار ساله آن دارد و به دوران نخستین امپراتوری ایرانی که توسط پارس‌ها بنیان گذاشته‌شد بازمی گردد. در دوران جدید، ایران بسیاری از **سرزمینهای** خود را از دست داد از جمله واگذاری بخش‌های غربی در ترکیه و عراق امروز به امپراتوری عثمانی (۱۵۳۳ میلادی)، واگذاری بخش‌های شرقی در افغانستان امروز به بریتانیا طی قرار داد پاریس در ۱۸۵۷ میلادی و ۱۹۰۵ میلادی و واگذاری **سرزمینهای** قفقاز به روسیه در قرن هجدهم و نوزدهم میلاد؛ عهدنامه ترکمانچای در سال ۱۸۲۸ و پس از نبرد روسیه و ایران، استانهای قفقازی ایران را برای همیشه به روسیه واگذار کرد و مرزهای جدید در طول رودخانه ارس شکل گرفت. بر طبق عهدنامه گلستان در سال ۱۸۱۳، مناطق ارمنستان، جمهوری آذربایجان و شرق گرجستان که پیشتر بخشی از ایران بودند، به روسیه واگذار شدند. در اثر این تجزیه تاریخی کشورها و ملت‌های جدیدی تحت نفوذ روسیه و انگلستان شکل گرفتند که اگرچه از طریق زبان یا فرهنگ با ایران پیوستگی داشتند اما جوامع خاص خود را شکل دادند. در سال ۱۹۳۵ در زمان سلطنت رضا شاه، نام ایران رسماً در مجامع بین‌المللی به‌عنوان نام بخش بجا مانده از **سرزمین** ایران بکار رفت.



Hadi Veisi (h.veisi@ut.ac.ir)

مدل مخفی مارکوف: معرفی . . .



سری زمانی ایستا (Stationary)

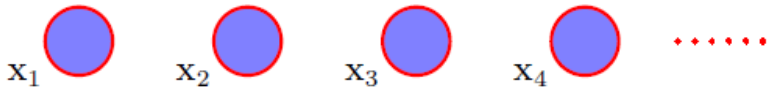
- فرایند ایستان (stationary)
  - فرآیندی که توزیع توام آن در طول زمان تغییر نکند
    - در نتیجه پارامترهای آماری مانند میانگین و واریانس هم در طول زمان ثابت می‌ماند

سری زمانی ناایستا (non-Stationary)

- فرایند ناایستان (non-stationary)
  - فرآیندی که توزیع توام آن و در نتیجه پارامترهای آماری در طول زمان تغییر کند
- فرایند ارگادیک (ergodic)
  - فرآیندی که می‌توان ویژگی‌های آماری آن مانند میانگین و واریانس را از روی یک نمونه از آن فرآیند (با تعداد نمونه‌های کافی) بدست آورد

مدل مخفی مارکوف: معرفی . . .

- برخورد با داده‌های ترتیبی
  - پیش‌بینی بارندگی: دنباله‌ای از بارانی بودن یا نبودن در N روز گذشته را در نظر می‌گیریم
  - حالت اول: عدم در نظر گرفتن رابطه بین مقادیر پشت سر هم دنباله (فرایند i.i.d)
    - در نظر نگرفتن وابستگی بین بارش در روزهای متوالی: عدم استفاده از دانش موجود در دنباله
    - تنها می‌توان درصد روزهای بارانی را محاسبه کرد



$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \prod_{n=1}^{N} p(\mathbf{x}_n)$$

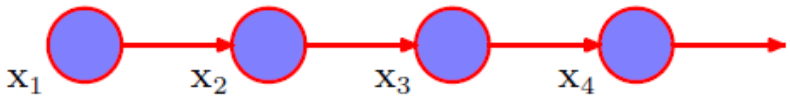
  - حالت دوم: در نظر گرفتن وابستگی بین مقادیر پشت سر هم دنباله (فرایند مارکوف)
    - استفاده از اطلاعات موجود در وابستگی مقادیر

$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \prod_{n=1}^{N} p(\mathbf{x}_n | \mathbf{x}_1, \ldots, \mathbf{x}_{n-1})$$

مدل مخفی مارکوف: معرفی

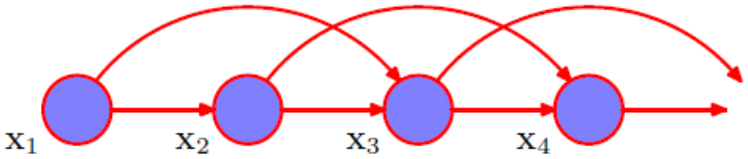
زنجیره مارکوف

- درجه اول (first-order Markov chain): هر نقطه (مشاهده) از دنباله تنها به نقطه قبل وابسته است



$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) \prod_{n=2}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}).$$

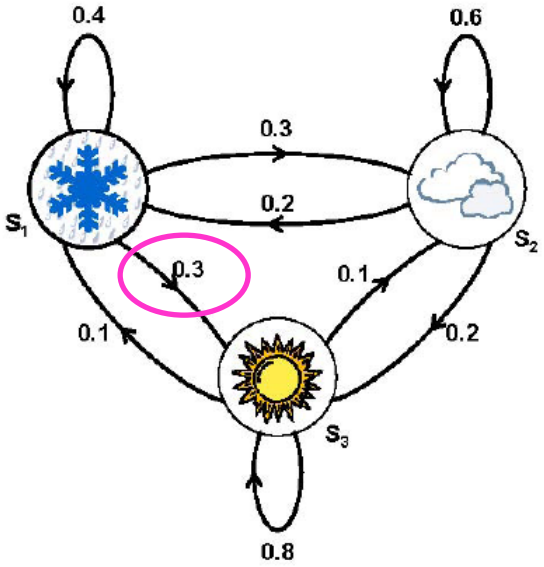
- درجه دوم (second-order Markov chain): هر نقطه (مشاهده) از دنباله به دو نقطه قبل وابسته است



$$p(\mathbf{x}_1, \ldots, \mathbf{x}_N) = p(\mathbf{x}_1) p(\mathbf{x}_2 | \mathbf{x}_1) \prod_{n=3}^{N} p(\mathbf{x}_n | \mathbf{x}_{n-1}, \mathbf{x}_{n-2})$$

- افزایش درجه زنجیره = افزایش پیچیدگی

مدل مخفی مارکوف: مثال . . .



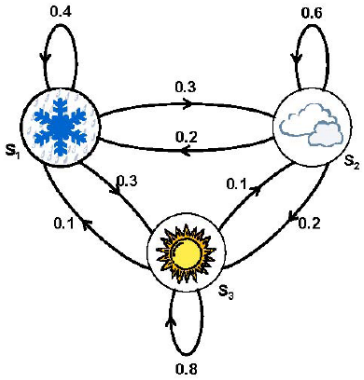
- پیش‌بینی وضعیت هوا . . .
  - در نظر گرفتن ۳ حالت (state) مختلف
    - حالت ۱ (S_1): بارندگی (برف یا باران)
    - حالت ۲ (S_2): ابری
    - حالت ۳ (S_3): آفتابی
  - در نظر گرفتن احتمال انتقال حالت‌ها (state transition probability)

$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

احتمال رفتن از حالت ۱ (بارندگی) به حالت ۳ (آفتابی)

- سوال ۱: با فرض اینکه امروز آفتابی است، احتمال اینکه هوای ۷ روز آینده به صورت زیر باشد، چقدر است؟ {آفتابی، آفتابی، باران، باران، آفتابی، ابری، آفتابی}
- سوال ۲: امروز بارانی است، احتمال اینکه d روز متوالی بارانی باشد، چقدر است؟

مدل مخفی مارکوف: مثال . . .



$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

- پیش‌بینی وضعیت هوا . . .

- **سوال ۱: امروز آفتابی است، احتمال اینکه هوای ۷ روز آینده به صورت زیر باشد، چقدر است؟ {آفتابی، آفتابی، باران، باران، آفتابی، ابری، آفتابی}**
  - استفاده از وابستگی درجه ۱ (وضعیت هر روز به روز قبل)

مشاهده

شامل حالت‌ها و ارتباط بین آنها

$$P(O|\text{Model}) = P[S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3|\text{Model}]$$

$$= P[S_3] \cdot P[S_3|S_3] \cdot P[S_3|S_3] \cdot P[S_1|S_3] \cdot P[S_1|S_1] \cdot P[S_3|S_1] \cdot P[S_2|S_3] \cdot P[S_3|S_2]$$

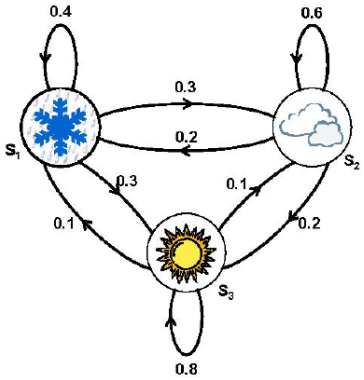
$$= \pi_3 \cdot a_{33} \cdot a_{33} \cdot a_{31} \cdot a_{11} \cdot a_{13} \cdot a_{32} \cdot a_{23}$$

$$= 1 \cdot (0.8)(0.8)(0.1)(0.4)(0.3)(0.1)(0.2)$$

$$= 1.536 \times 10^{-4}$$

احتمال حالت اولیه
(initial state probability)

مدل مخفی مارکوف: مثال . . .



$$A = \{a_{ij}\} = \begin{bmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}$$

- پیش‌بینی وضعیت هوا . . .

- **سوال ۲: امروز بارانی است، احتمال اینکه d روز متوالی بارانی باشد، چقدر است؟**
  - استفاده از وابستگی درجه ۱ (وضعیت هر روز به روز قبل)
  - سوال فوق معادل مشاهده دنباله روبرو است (حالت کلی) $O = \{S_{i_1}, S_{i_2}, S_{i_3}, \cdots, S_{i_d}, S_{i_{d+1}} \neq S_i\}$

$$P(O|\text{Model}, q_1 = S_i) = (a_{ii})^{d-1}(1 - a_{ii}) = p_i(d).$$

تابع توزیع احتمال دوره d در حالت i

  - برای حالت بارانی (S_1) داریم: $p_1(d)=(0.4)^{d-1}(0.6)$ که برای $d=5$ برابر است با 0.0061

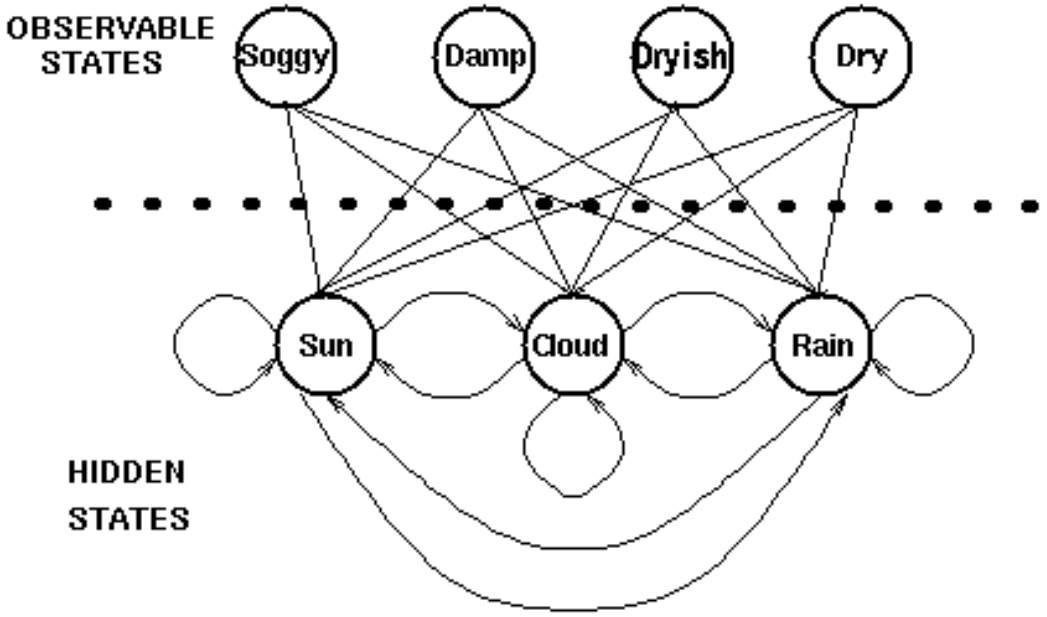
- **سوال ۳: با فرض شروع از یک حالت (مثلاً آفتابی)، متوسط تعداد روزهای پشت سرهم آینده که در همان حالت می‌مانیم، چند روز است؟**
  - برابر است با امید ریاضی $p_i(d)$
    - آفتابی = ۵ روز (۱/۰.۲)
    - بارندگی = ۱.۶۷
    - ابری = ۲.۵

$$\bar{d}_i = \sum_{d=1}^{\infty} d p_i(d)$$

$$= \sum_{d=1}^{\infty} d(a_{ii})^{d-1}(1 - a_{ii}) = \frac{1}{1 - a_{ii}}.$$

مدل مخفی مارکوف: مثال . . .

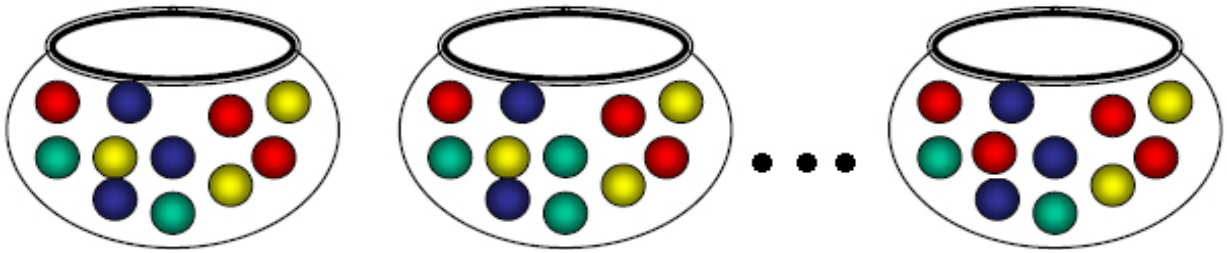
- پیش‌بینی وضعیت هوا
  - در آنچه تاکنون بیان شد: حالت‌ها با مشاهده‌ها یکسان بودند
  - در بسیاری از کاربردها، مشاهده‌ها با حالت‌های مساله یکی نیستند
  - حالت‌های اصلی مخفی هستند و باید مشاهده‌ها را با آنها متناظر کرد



مدل مخفی مارکوف
Hidden Markov Model

مدل مخفی مارکوف: مثال . . .

- گوی و گلدان . . .
  - فرض کنید تعداد N گلدان در یک اتاق داریم
  - در هر گلدان تعداد زیادی گوی رنگی، شامل M رنگ وجود دارد



URN 1

P(RED) = $b_1(1)$
P(BLUE) = $b_1(2)$
P(GREEN) = $b_1(3)$
P(YELLOW) = $b_1(4)$
⋮
P(ORANGE) = $b_1(M)$

URN 2

P(RED) = $b_2(1)$
P(BLUE) = $b_2(2)$
P(GREEN) = $b_2(3)$
P(YELLOW) = $b_2(4)$
⋮
P(ORANGE) = $b_2(M)$

URN N

P(RED) = $b_N(1)$
P(BLUE) = $b_N(2)$
P(GREEN) = $b_N(3)$
P(YELLOW) = $b_N(4)$
⋮
P(ORANGE) = $b_N(M)$

مدل مخفی مارکوف: مثال

- گوی و گلدان
  - تعداد N گلدان و M رنگ
  - فرآیند
    - یک نفر (در اتاقی که ما نمی‌بینیم)، یکی از گلدان‌ها را به صورت تصادفی انتخاب می‌کند
    - از داخل گلدان انتخاب شده، یک گوی را بیرون آورده و رنگ آن را اعلام می‌کند
    - گوی را به داخل گلدان مربوطه برمی‌گرداند
    - بر اساس مقداری تصادفی وابسته به گلدان فعلی، گلدان بعدی انتخاب می‌شود
    - مراحل فوق به صورت متوالی تکرار می‌شود

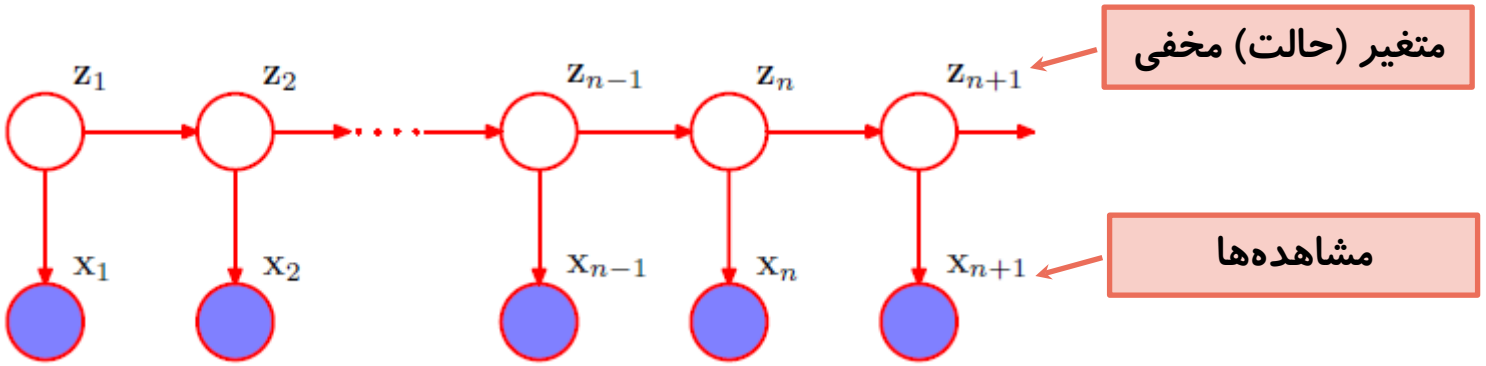
- دنباله مشاهده: دنباله گوی‌ها (رنگ‌ها)

$$O = \{GREEN, GREEN, BLUE, RED, YELLOW, RED, \ldots\ldots, BLUE\}$$

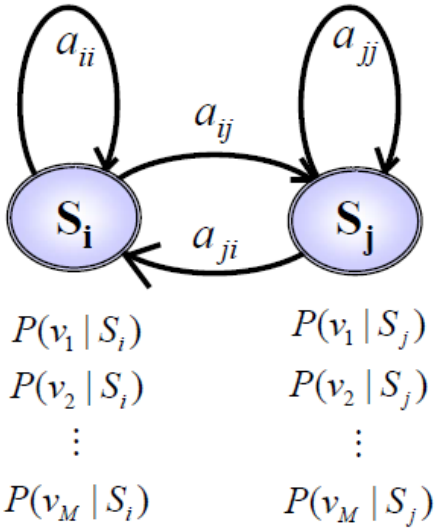
- حالت‌ها: گلدان‌ها (از دید مشاهده کننده مخفی است)
- انتقال حالت‌ها: فرآیند انتخاب گلدان‌ها

مدل مخفی مارکوف: مبانی

- مشاهده‌ها توابع احتمالاتی از حالت‌ها هستند
- دنباله حالت‌ها قابل مشاهده نیستند (مخفی هستند)
- فرض وابستگی درجه اول
- مشاهده‌ها فقط به حالت‌ها وابسته هستند و نه به همدیگر



مدل مخفی مارکوف: تعریف . . .



- **عناصر اصلی یک مدل مخفی مارکوف**

1. **مجموعه‌ای از N حالت**
   - در گوی و گلدان: گلدان‌ها

$$S = \{S_1, \cdots S_N\}$$

2. **مجموعه‌ای از M نماد مشاهده**
   - در گوی و گلدان: رنگ‌ها

$$V = \{v_1, v_2, \cdots v_M\}$$

3. **احتمال انتقال حالت‌ها**
   - احتمال جابجایی از یک حالت به حالت دیگر

$$a_{ij} = P(q_t = S_j \mid q_{t-1} = S_i)$$

4. **احتمال اولیه حالت‌ها**
   - احتمال انتخاب هر کدام از حالت‌ها در زمان t=1

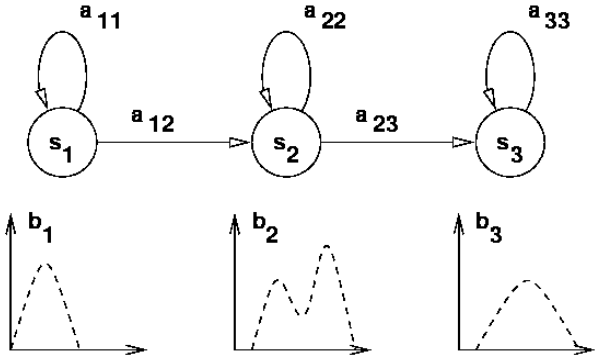
$$\pi = \{\pi_i\} = P(q_1 = i)$$

5. **تابع توزیع برای نماد kام در حالت jام**
   - تابع توزیع مشاهده‌ها (مثلاً گاوسی)- احتمال تولید مشاهده $o_t=v_k$ در حالت $q_t=j$

$$b_j(k) = P(o_t = v_k \mid q_t = j)$$

مدل مخفی مارکوف: تعریف . . .

- پس، یک مدل مخفی مارکوف شامل پارامترهای زیر است:
  - تعداد حالت‌ها (N) و تعداد نمادهای مشاهده‌ها (M)
  - مجموعه M نماد مشاهده
  - احتمال‌ها $\lambda = (A, B, \pi)$
    - A= احتمال انتقال بین حالت‌ها
    - π= احتمال اولیه حالت‌ها
    - B= تابع توزیع‌های حالت‌ها



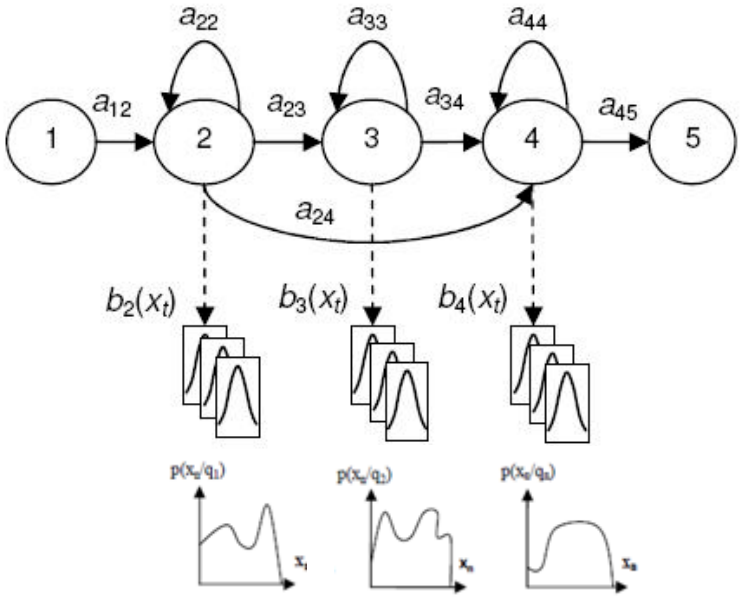
- مراحل تولید دنباله مشاهده $O = O_1 O_2 \cdots O_T$ با مدل مخفی مارکوف

1. بر اساس احتمال‌های اولیه حالت‌ها، اولین حالت را انتخاب کن، $q_1=S_i$ و قرار بده $t=1$
2. بر اساس تابع توزیع احتمال $b_i(k)$ خروجی را تعیین کن، $O_t=v_k$
3. بر اساس احتمال انتقال a_{ij} از حالت فعلی به حالت بعدی برو، $q_{t+1}=S_j$
4. اگر $t<T$، قرار بده $t=t+1$ و برو به گام ۲

مدل مخفی مارکوف: تعریف

- مدل مخفی مارکوف نوعی توسعه از مدل مخلوط گاوسی (GMM) است

- مدل مخلوط گاوسی (GMM)
  - معادل با مدل مخفی مارکوف با یک حالت (و چند مخلوط در آن حالت)
  - مدل کردن الگوهای ایستا



- مدل مخفی مارکوف (HMM)
  - مدل کردن الگوهای ترتیبی

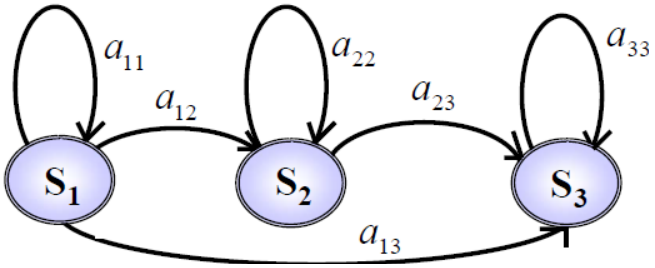
مدل مخفی مارکوف: انواع . . .

- چپ به راست (Left-to-Right)
  - جهت حرکت از یک حالت به حالت بعدی، از چپ به راست است
    - حرکت بین حالت‌ها از راست به چپ مجاز نیست

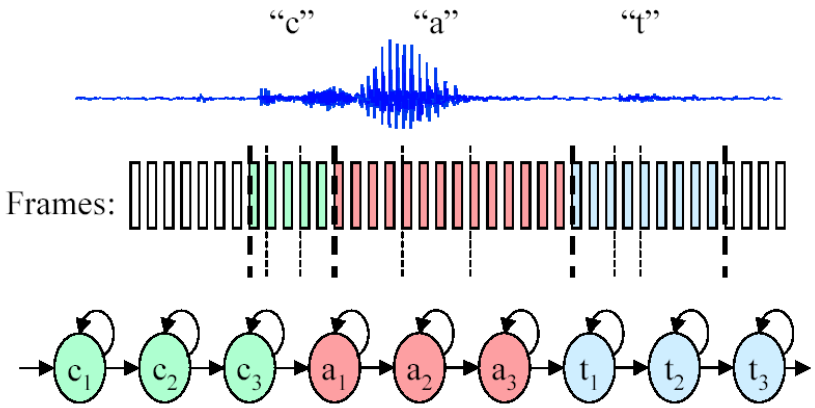
$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$a_{ij} = 0 \qquad j < i$$

$$\pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases}$$



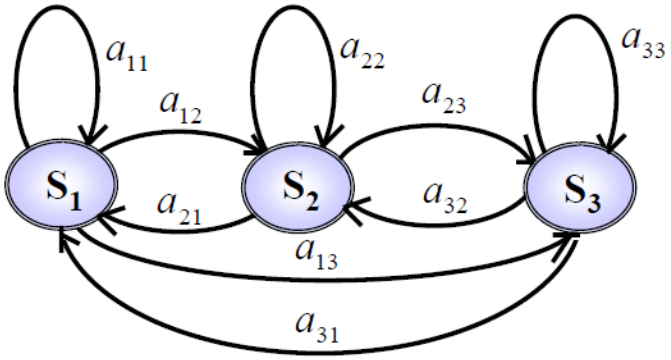
- کاربرد: تشخیص گفتار، تشخیص دست‌خط
  - جهت تولید داده‌ها فقط در یک سمت است



مدل مخفی مارکوف: انواع

- ارگادیک (Ergodic)
  - حرکت از هر حالتی به هر حالت دیگر مجاز است

$$a_{ij} > 0 \qquad \forall i, \forall j$$



  - کاربردها: بهسازی گفتار، کدینگ

- انواع دیگری هم برای ارتباط بین حالت‌ها ممکن است

مدل مخفی مارکوف: ۳ مساله مهم

- **ارزیابی: محاسبه احتمال مشاهده**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان مقدار $p(O \mid \lambda)$ را محاسبه کرد؟
- **دیکدینگ: یافتن دنباله حالت‌ها**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان بهترین دنباله حالت‌های $Q=q_1q_2\dots q_T$ را که متناسب با مشاهده است، بدست آورد؟
- **آموزش مدل**
  - پارامترهای مدل $\lambda(A,B,\pi)$ را چگونه بدست آوریم که $p(O \mid \lambda)$ بیشینه شود؟

محاسبه احتمال مشاهده ...

- محاسبه $p(O|\lambda)$ (دنباله مشاهده $O=O_1O_2...O_T$ و مدل λ)
  - فرض کنید دنباله حالت‌های معادل دنباله مشاهده فوق، $Q=q_1q_2...q_T$ باشد
  - داریم

$$P(\mathbf{O}|\lambda) = \sum_{\text{all q}} P(\mathbf{O}|q,\lambda)P(q|\lambda)$$

احتمال تولید دنباله مشاهدات توسط حالت‌های در نظر گرفته شده

احتمال انتخاب دنباله حالت‌های در نظر گرفته شده

  - که (برای یکی از qها)

$$P(\mathbf{O}|q,\lambda) = \prod_{t=1}^{T} p(\mathbf{o}_t|q_t,\lambda) = b_{q_1}(\mathbf{o}_1)\cdot b_{q_2}(\mathbf{o}_2)\cdots b_{q_T}(\mathbf{o}_T)$$

$$P(q|\lambda) = \pi_{q_1}(a_{q_1q_2})\cdot(a_{q_2q_3})\cdots(a_{q_{T-1}q_T})$$

محاسبه احتمال مشاهده ...

- بنابراین

$$P(\mathbf{O}\mid\lambda)=\sum_{\text{all q}}P(\mathbf{O}\mid q,\lambda)P(q\mid\lambda)=\sum_{\text{all q}}\pi_{q_1}b_{q_1}(o_1)a_{q_1q_2}b_{q_2}(o_2)\cdots a_{q_{T-1}q_T}b_{q_T}(o_T)$$

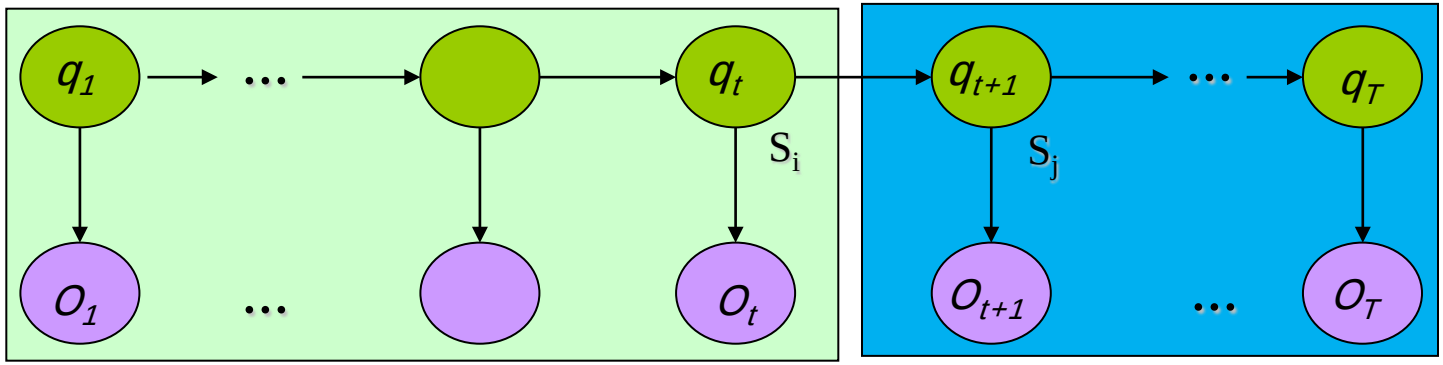
- اما ...

طول دنباله

تعداد حالت‌ها

- محاسبات با این روش بسیار بالاست $O(2T*N^T)$
  - اگر یک دنباله ۳۰ تایی داشته باشیم و تعداد حالت‌ها ۱۰ باشد، محاسبات=$۲۰*۱۰^{۳۰}$

- راه‌حل: الگوریتم جلورو (Forward) یا عقب‌رو (Backward)
  - مشاهده $O=O_1O_2\dots O_T$ را به صورت $O=O_1O_2\dots O_t\,O_{t+1}\dots O_T$ می‌نویسیم



محاسبه احتمال مشاهده: الگوریتم جلورو . . .

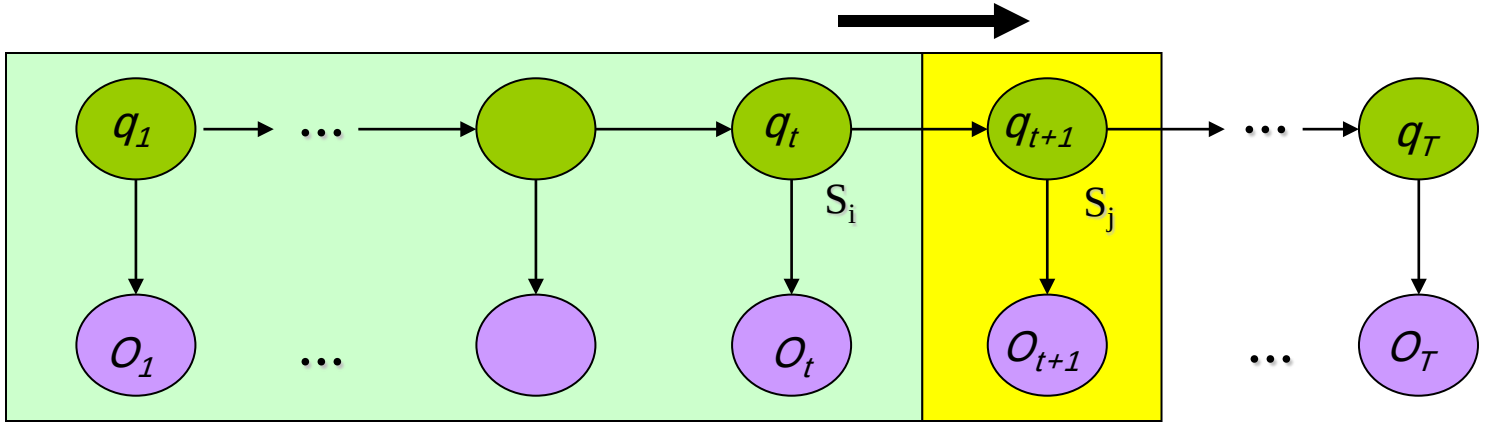
- الگوریتم جلورو (Forward) . . .
  - تعریف $\alpha_t(i)$: احتمال بودن در حالت i در زمان t (حضور در حالت i بعد از دیدن t مشاهده اول)

$$\alpha_t(i) = P(\mathbf{o}_1 \mathbf{o}_2 \ldots \mathbf{o}_t, q_t = i \mid \lambda)$$

    - شروع از اولین مشاهده و رسیده به مشاهده tام

  - با توجه به $\alpha_t(i)$، می‌توان $\alpha_{t+1}(j)$ را حساب کرد:

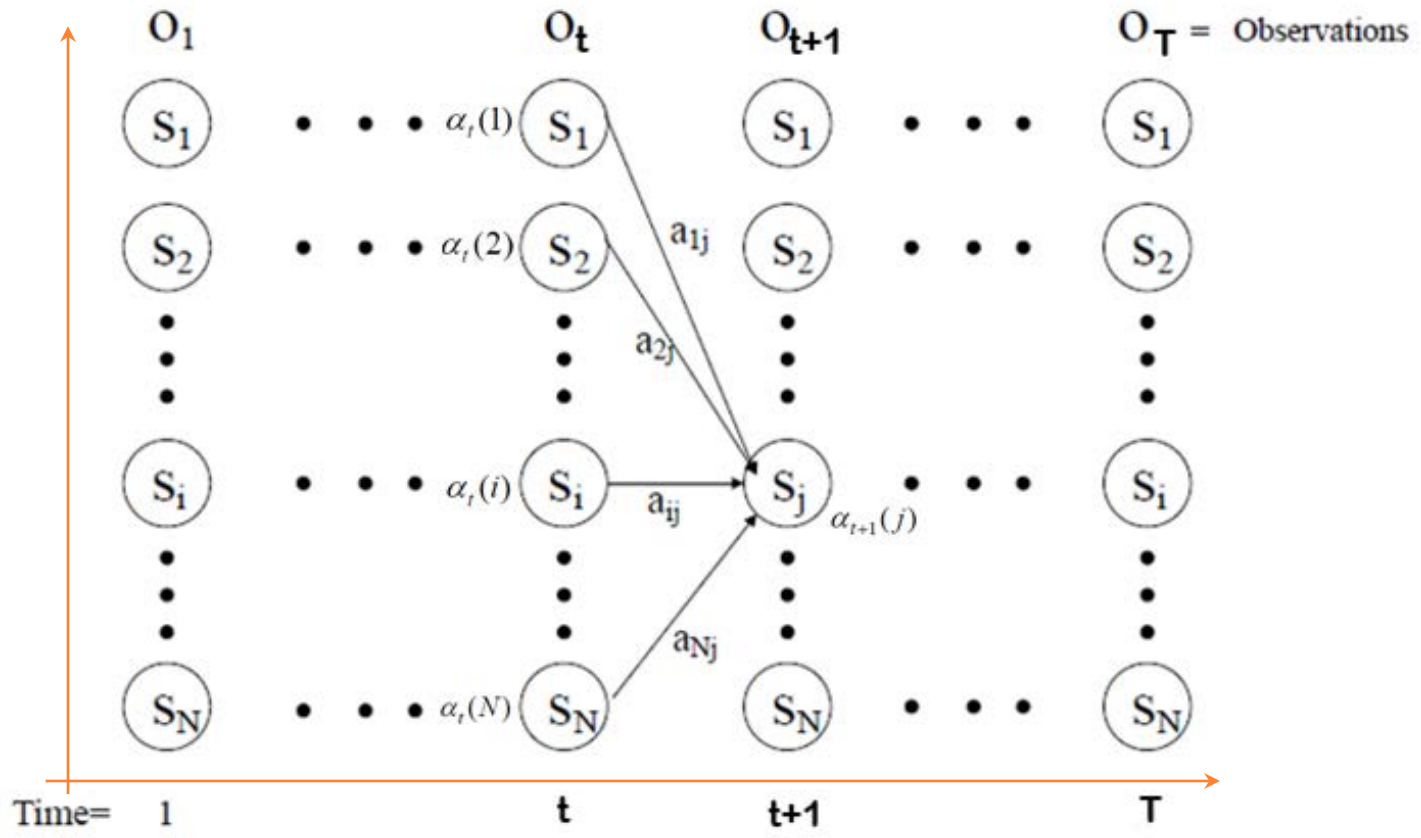
$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(\mathbf{o}_{t+1})$$



محاسبه احتمال مشاهده: الگوریتم جلورو . . .

- الگوریتم جلورو (Forward) . . .
  - رابطه $\alpha_t(i)$ و $\alpha_{t+1}(j)$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(\mathbf{o}_{t+1})$$



محاسبه احتمال مشاهده: الگوریتم جلورو . . .

- **الگوریتم جلورو (Forward): مراحل**
  - **گام اول: مقداردهی اولیه در زمان t=0**
    - برای تمامی حالت‌ها (از i=1 تا N)

$$\alpha_0(i) = \pi_i$$

  - **گام دوم: محاسبه مقدار $\alpha_{t+1}(j)$ از روی مقادیر $\alpha_t(i)$ برای تمامی حالت‌ها و تمام مشاهده‌ها**
    - برای تمامی tها (از t=0 تا T)
    - و برای تمامی حالت‌ها (از j=1 تا N)

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(\mathbf{o}_{t+1})$$

  - **گام سوم: محاسبه مقدار $p(O \mid \lambda)$**

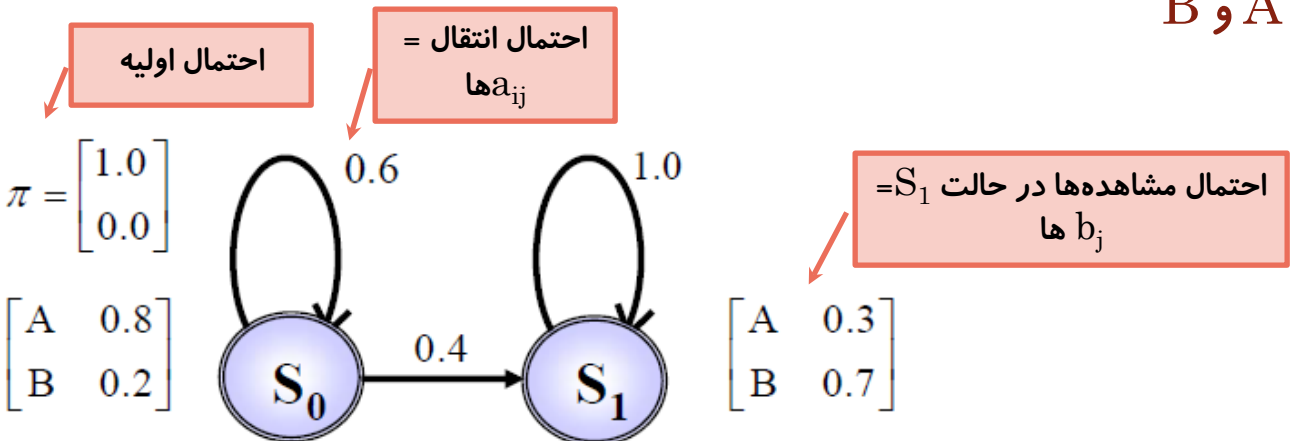
$$P(\mathbf{O} \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

- **بار محاسباتی الگوریتم جلورو (Forward)**

$$O(T * N^2)$$

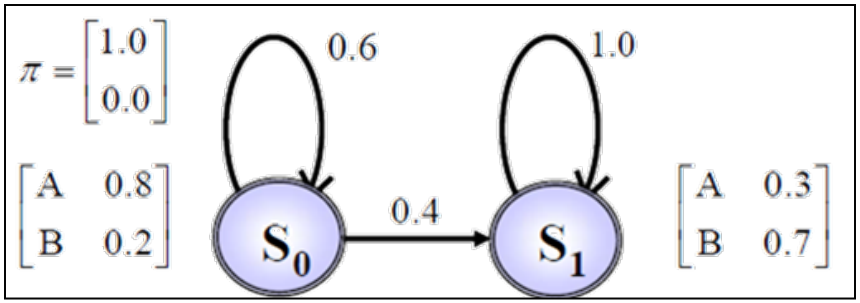
محاسبه احتمال مشاهده: الگوریتم جلورو . . .

- الگوریتم جلورو (Forward): مثال . . .
  - دو مشاهده A و B
  - مدل λ



- مطلوب است محاسبه احتمال مشاهده دنباله O=AAB، یعنی $p(O=\{A,A,B\} \mid \lambda)$

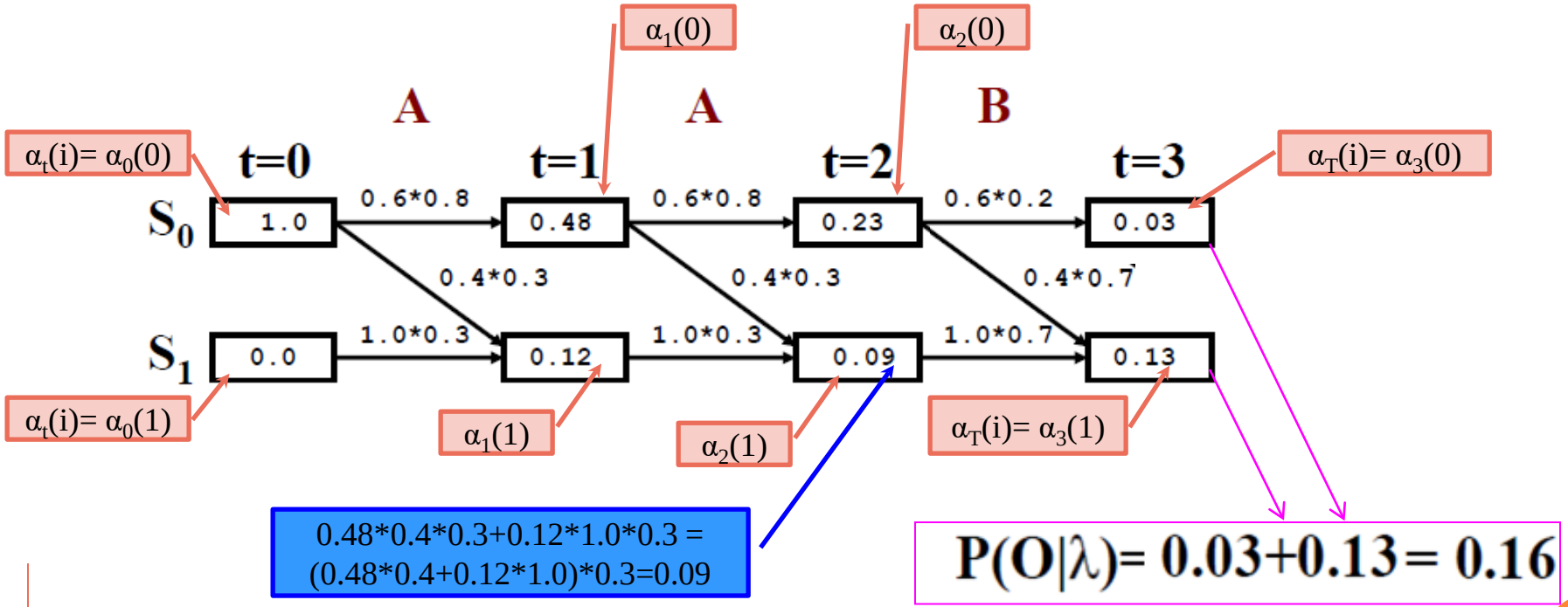
محاسبه احتمال مشاهده: الگوریتم جلورو . . .



- الگوریتم جلورو (Forward): مثال

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(\mathbf{o}_{t+1})$$

$$P(\mathbf{O} \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i)$$



محاسبه احتمال مشاهده: الگوریتم عقب‌رو ...

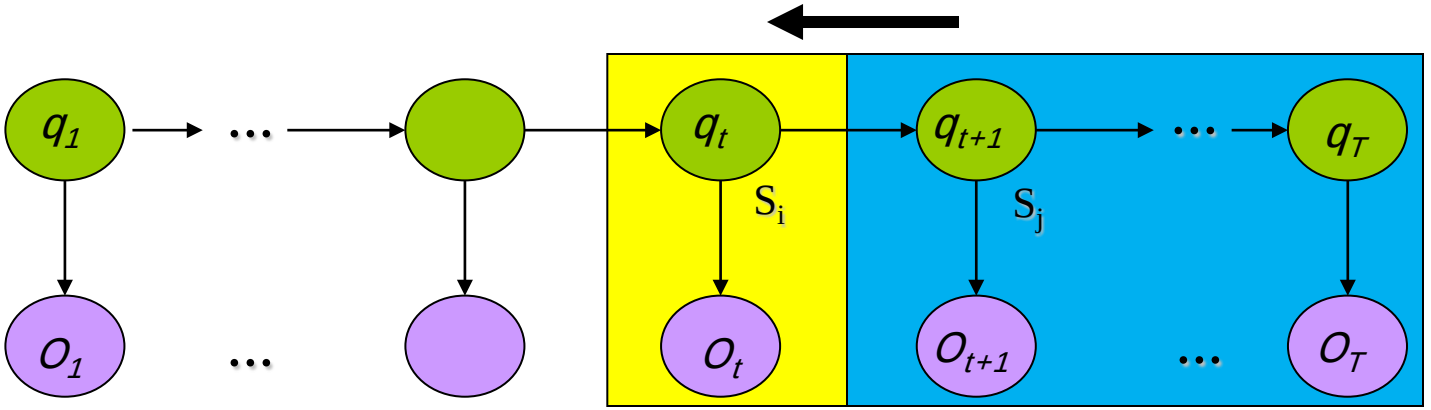
- **الگوریتم عقب‌رو (Backward) ...**
  - **تعریف $\beta_t(i)$: احتمال دنباله مشاهده o_{t+1} تا o_T با فرض بودن در حالت i در زمان t (شروع از حالت i و دیدن مشاهده‌های پایانی)**
    - شروع از حالت iام و دیدن مشاهده $t+1$ام تا رسیدن به انتهای دنباله مشاهده‌ها

$$\beta_t(i) = P(\mathbf{o}_{t+1}\mathbf{o}_{t+2}\ldots\mathbf{o}_T, q_t = i \mid \lambda)$$

- **با توجه به $\beta_{t+1}(j)$، می‌توان $\beta_t(i)$ را حساب کرد:**

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)$$

$$t = T-1, T-2, \ldots, 1$$
$$1 \le i \le N$$

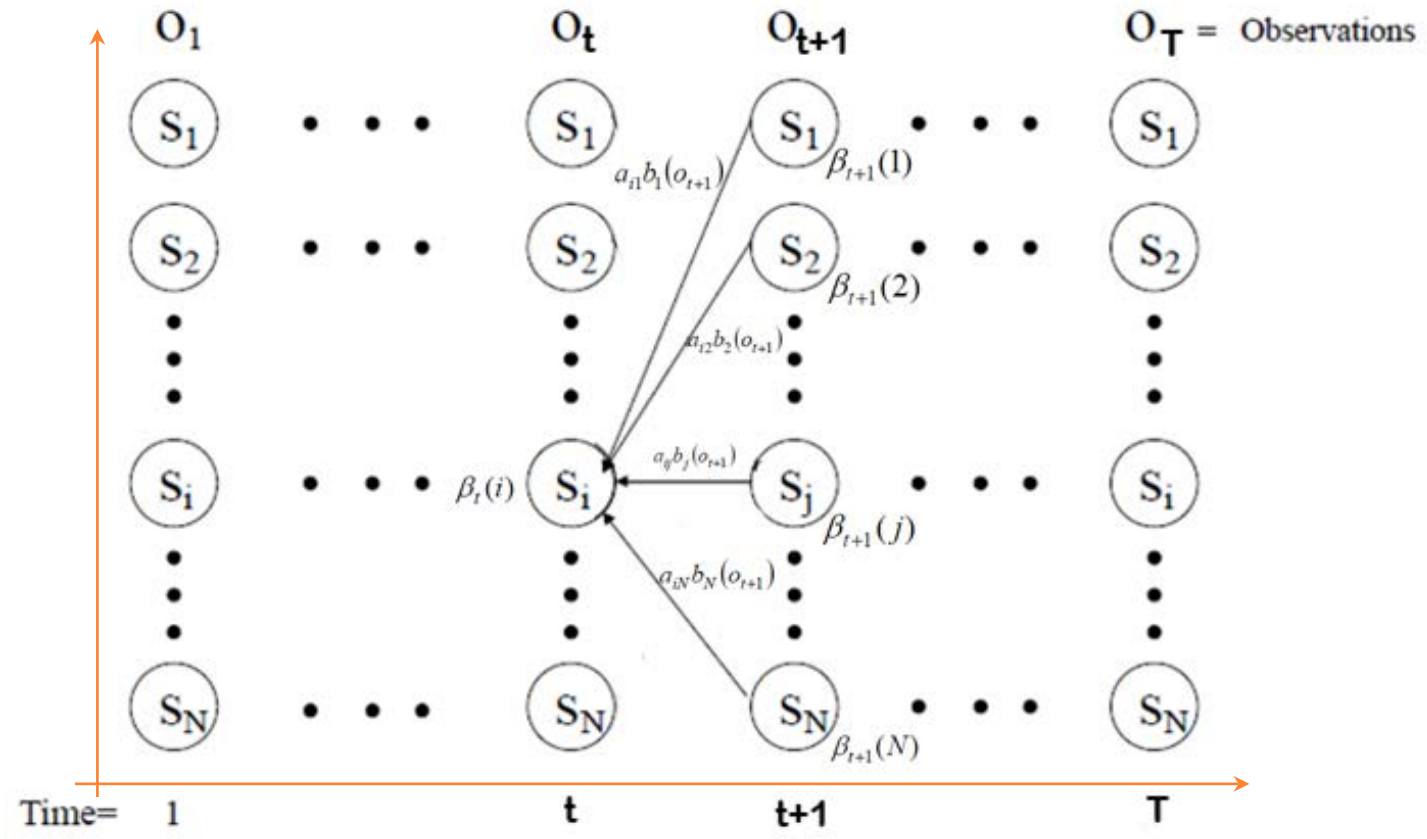


محاسبه احتمال مشاهده: الگوریتم عقب‌رو . . .

- الگوریتم عقب‌رو (Backward) . . .
  - محاسبه $\beta_t(i)$ از روی مقدارهای $\beta_{t+1}(j)$

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)$$

$$t = T-1, T-2, \ldots, 1$$
$$1 \le i \le N$$



محاسبه احتمال مشاهده: الگوریتم عقب‌رو . . .

- **الگوریتم عقب‌رو (Backward): مراحل**
  - **گام اول: مقداردهی اولیه β در زمان t=T**
    - برای تمامی حالت‌ها (از i=1 تا N)

$$\beta_T(i) = 1$$

  - **گام دوم: محاسبه مقدار $\beta_t(i)$ از روی مقادیر $\beta_{t+1}(j)$ برای تمامی حالت‌ها و تمام مشاهده‌ها**
    - برای تمامی tها (از t=T-1 تا 0)
    - و برای تمامی حالت‌ها (از i=1 تا N)

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)$$

  - **گام سوم: محاسبه مقدار $p(O \mid \lambda)$**

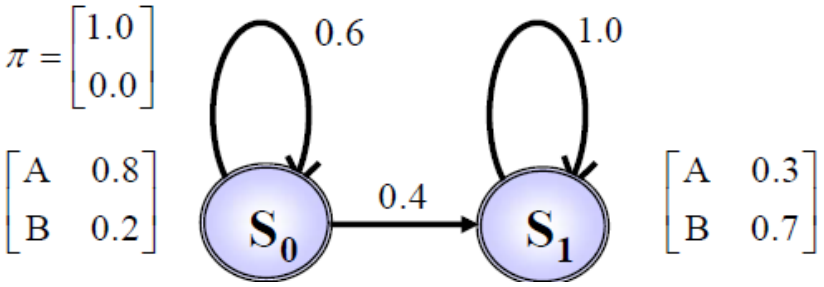
$$P(\mathbf{O} \mid \lambda) = \sum_{i=1}^{N} \pi_i \beta_0(i)$$

- **بار محاسباتی الگوریتم عقب‌رو (Backward)**
  - **مشابه الگوریتم جلورو (Forward)**

$$O(T * N^2)$$

محاسبه احتمال مشاهده: الگوریتم عقب‌رو . . .

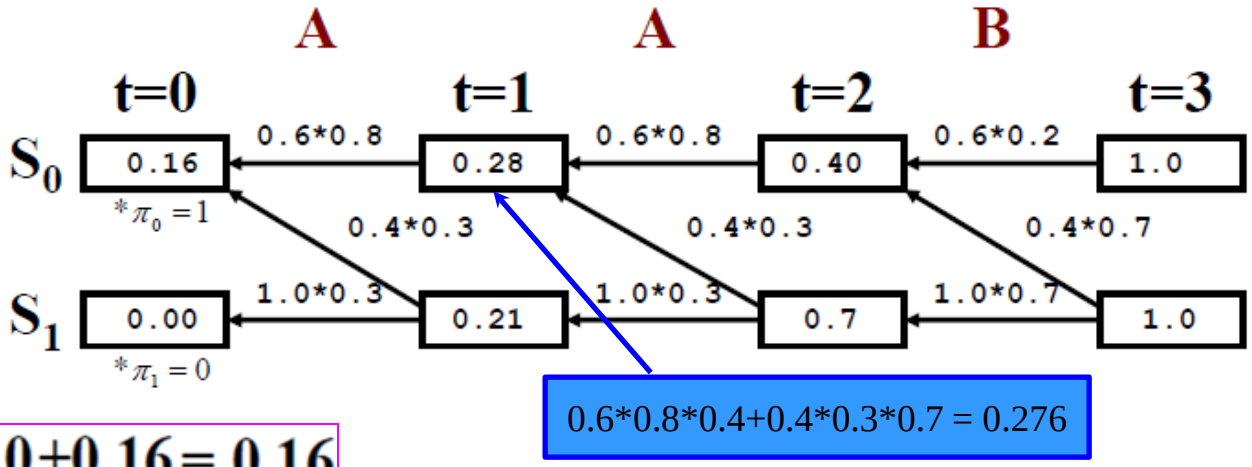
- الگوریتم عقب‌رو (Backward): مثال (قبلی)
  - دو مشاهده A و B و مدل λ



$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)$$

$$P(\mathbf{O} \mid \lambda) = \sum_{i=1}^{N} \pi_i \beta_0(i)$$

- مطلوب است محاسبه احتمال مشاهده دنباله O=AAB، یعنی $p(O=\{A,A,B\} \mid \lambda)$



$P(O|\lambda) = 0.0+0.16 = 0.16$

محاسبه احتمال مشاهده: الگوریتم جلورو یا عقب‌رو

- برای محاسبه $p(O|\lambda)$
  - فقط یکی از دو الگوریتم جلورو (Forward) یا عقب‌رو (Backward) کافی است
  - برای حل مساله ۳ به هردو الگوریتم نیاز است

مدل مخفی مارکوف: ۳ مساله مهم

- **ارزیابی: محاسبه احتمال مشاهده**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان مقدار $p(O \mid \lambda)$ را محاسبه کرد؟

الگوریتم جلورو یا عقب‌رو

- **دیکدینگ: یافتن دنباله حالت‌ها**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان بهترین دنباله حالت‌های $Q=q_1q_2\dots q_T$ که متناسب با مشاهده است، را بدست آورد؟

- **آموزش مدل**
  - پارامترهای مدل $\lambda(A,B,\pi)$ را چگونه بدست آوریم که $p(O \mid \lambda)$ بیشینه شود؟

دیکدینگ (یافتن دنباله حالت‌ها) . . .

- یافتن بهترین دنباله حالت $Q=q_1q_2\ldots q_T$ متناسب با دنباله مشاهده $O=O_1O_2\ldots O_T$ در مدل λ
  - راه‌حل کامل: بررسی تمام دنباله حالت‌های ممکن و انتخاب بهترین آنها
    - بسیار زمان‌بر، از مرتبه $O(TN^T)$ که N تعداد حالت‌ها و T طول دنباله مشاهده‌هاست
  - راه‌حل ۱: برای هر مشاهده، یک حالت که به آن مشاهده شبیه‌تر است، انتخاب کنیم
  - راه‌حل ۲: در هر مرحله، حالتی را انتخاب کن که احتمال $\alpha_t(i)$ آن بیشتر است
  - مشکلات دو راه‌حل
    - تولید دنباله بهینه محلی
    - ممکن است یک دنباله غیرممکن را ایجاد کند! (دنباله‌ای که در آن انتقال از یک حالت به حالت دیگر ممکن نیست)
  - راه‌حل ۳: الگوریتم ویتربی (Viterbi): بیشینه کردن $p(O, Q \mid \lambda)$

دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .

- تعریف $\delta_t(i)$
  - محتمل‌ترین دنباله حالتی که از مشاهده اول تا مشاهده tام را شامل شده و در حالت iام پایان یافته است

$$\delta_t(i) = \max_{q_1 q_2 \cdots q_{t-1}} P(q_1 q_2 \cdots q_{t-1}, q_t = i, \mathbf{o}_1 \mathbf{o}_2 \ldots \mathbf{o}_t \mid \lambda)$$

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(o_{t+1})$$

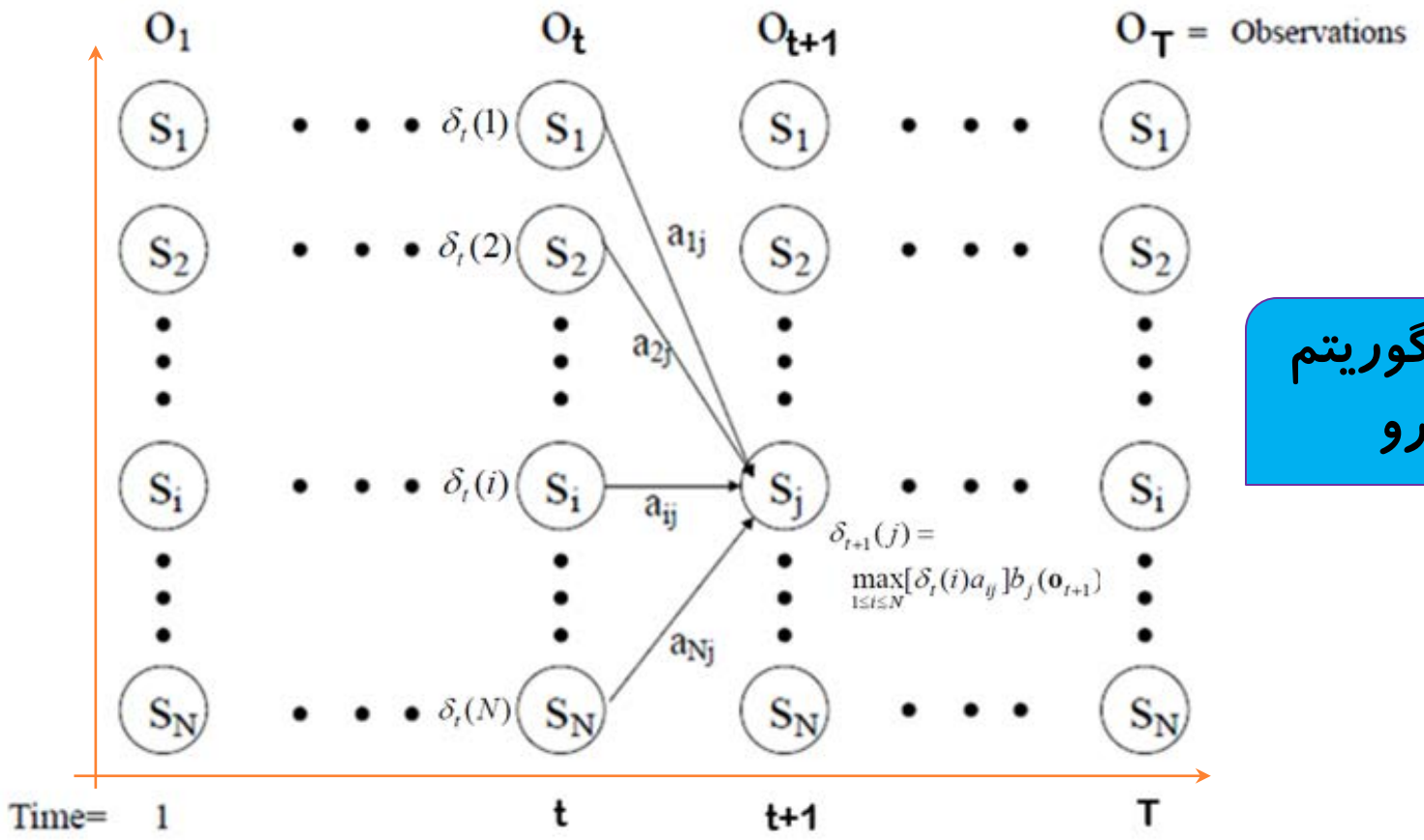
- محاسبه $\delta_{t+1}(j)$ از روی $\delta_t(i)$
  - بیشینه کردن به ازای تمام حالت‌های $i=1$ تا N

- سوال: تفاوت $\alpha_t(i)$ با $\delta_t(i)$؟
  - در محاسبه $\alpha_t(i)$ مجموع کلیه احتمال‌ها در حالت iام محاسبه می‌شود
  - در محاسبه $\delta_t(i)$ بیشینه کلیه احتمال‌ها در حالت iام محاسبه می‌شود

دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .

- محاسبه $\delta_{t+1}(j)$ از روی $\delta_t(i)$

$$\delta_{t+1}(j) = [\max_i \delta_t(i) a_{ij}] \cdot b_j(o_{t+1})$$



مشابه الگوریتم جلورو

دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .

مراحل الگوریتم ویتربی

- گام اول: مقداردهی اولیه δها
  - برای تمام حالت‌ها

$$\delta_1(i) = \pi_i b_i(\mathbf{o}_1) \qquad \psi_1(i) = 0$$

- گام دوم: مرحله بازگشتی

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(\mathbf{o}_t)$$

$$\psi_t(j) = \arg\max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}]$$

حاوی اندیس بهترین حالت‌ها

- گام سوم: پایان

$$P^* = \max_{1 \le i \le N} [\delta_T(i)] \qquad q_T^* = \arg\max_{1 \le i \le N} [\delta_T(i)]$$

- گام سوم: عقب‌گرد برای انتخاب دنباله مسیر بهینه

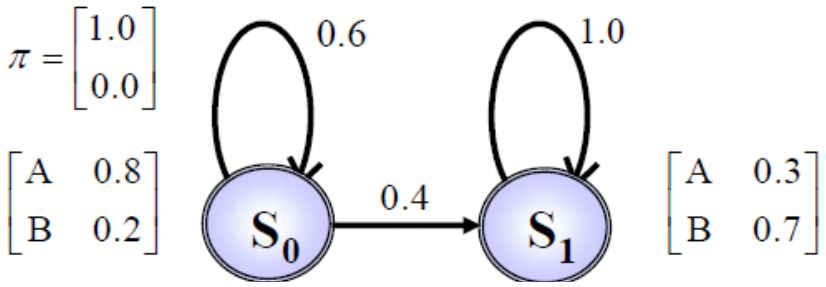
$$q_t^* = \psi_{t+1}(q_{t+1}^*)$$

مشابه الگوریتم جلورو: بیشینه به جای جمع در محاسبه احتمال‌ها + نگهداری بهترین مسیر

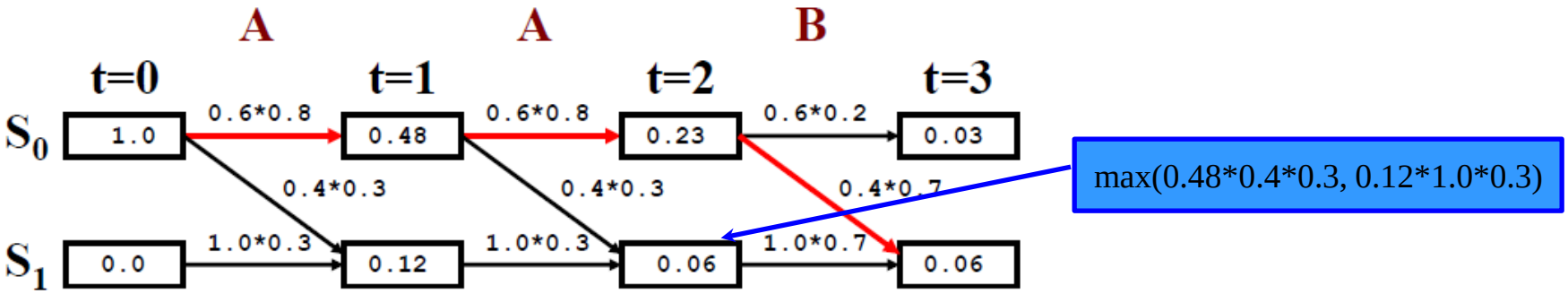
دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .

- **مثال: همان مثال قبلی**
  - دو مشاهده A و B و مدل λ

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(\mathbf{o}_t)$$

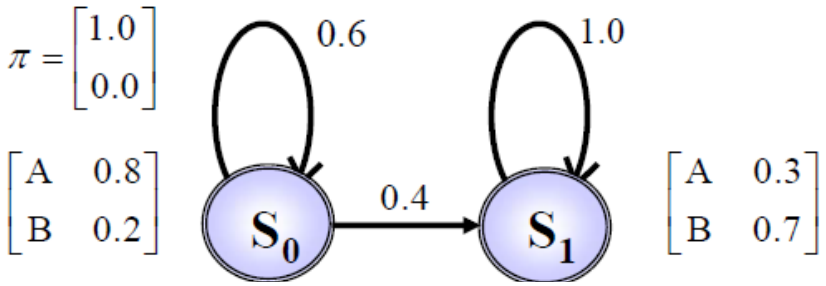


- مطلوب است یافتن دنباله حالات متناسب با دنباله مشاهده O=AAB



- بنابراین، داریم: $q^* = S_0 S_0 S_1$

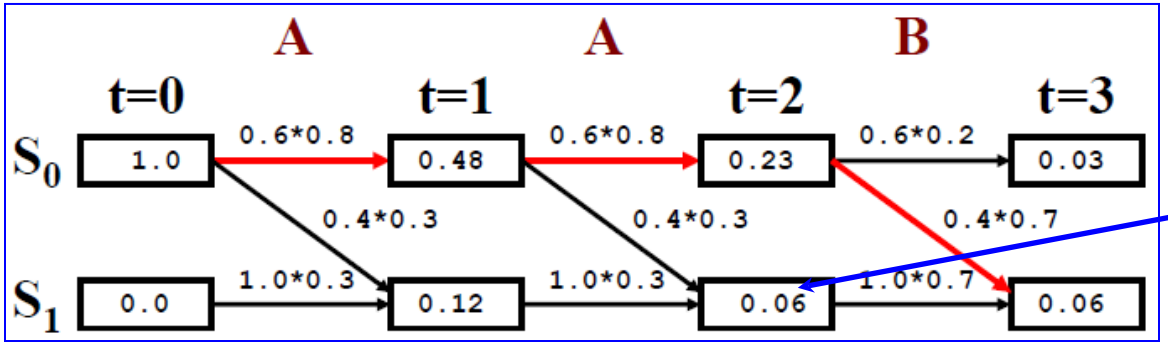
دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .



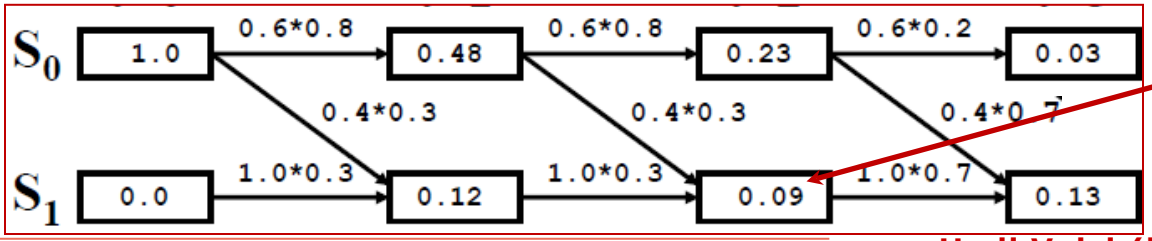
- همان مثال قبلی
  - مقایسه $\delta_t(i)$ با $\alpha_t(i)$

$$\delta_t(j) = \max_{1 \le i \le N} [\delta_{t-1}(i) a_{ij}] b_j(\mathbf{o}_t)$$

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(\mathbf{o}_{t+1})$$



max(0.48*0.4*0.3, 0.12*1.0*0.3)



0.48*0.4*0.3+0.12*1.0*0.3 =
(0.48*0.4+0.12*1.0)*0.3=0.09

دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی . . .

- استفاده از لگاریتم
  - به دلیل ضرب مقادیر احتمال‌ها (که کمتر از ۱ هستند)، با افزایش طول دنباله مشاهده، مقادیر δ به صفر می‌رسند (underflow)
  - استفاده از لگاریتم احتمال، محاسبات را ساده‌تر می‌کند: تبدیل ضرب به جمع

$$\tilde{\pi}_i = \log(\pi_i)$$
$$\tilde{b}_j(o_t) = \log(b_j(o_t))$$
$$\tilde{a}_{ij} = \log(a_{ij})$$

  - تغییر الگوریتم ویتربی: تبدیل ضرب احتمال‌ها به جمع مقدار لگاریتم آنها

دیکدینگ (یافتن دنباله حالت‌ها): الگوریتم ویتربی

مراحل الگوریتم ویتربی در دامنه لگاریتم

- گام اول: مقداردهی اولیه δها

$$\tilde{\delta}_1(i) = \tilde{\pi}_i + \tilde{b}_i(\mathbf{o}_1) \quad \psi_1(i) = 0$$

- گام دوم: مرحله بازگشتی

$$\tilde{\delta}_t(j) = \max_{1 \le i \le N}[\tilde{\delta}_{t-1}(i) + \tilde{a}_{ij}] + \tilde{b}_j(\mathbf{o}_t)$$

$$\psi_t(j) = \arg\max_{1 \le i \le N}[\tilde{\delta}_{t-1}(i) + \tilde{a}_{ij}]$$

- گام سوم: پایان

$$\tilde{P}^* = \max_{1 \le i \le N}\left[\tilde{\delta}_T(i)\right] \quad q_T^* = \arg\max_{1 \le i \le N}\left[\tilde{\delta}_T(i)\right]$$

- گام سوم: عقب‌گرد برای انتخاب دنباله مسیر بهینه

$$q_t^* = \psi_{t+1}(q_{t+1}^*)$$

مدل مخفی مارکوف: ۳ مساله مهم

- **ارزیابی: محاسبه احتمال مشاهده**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان مقدار $p(O \mid \lambda)$ را محاسبه کرد؟

الگوریتم جلورو یا عقب‌رو

- **دیکدینگ: یافتن دنباله حالت‌ها**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان بهترین دنباله حالت‌های $Q=q_1q_2\dots q_T$ که متناسب با مشاهده است، را بدست آورد؟

الگوریتم ویتربی

- **آموزش مدل**
  - پارامترهای مدل $\lambda(A,B,\pi)$ را چگونه بدست آوریم که $p(O \mid \lambda)$ بیشینه شود؟

آموزش مدل . . .

- تعیین مقدار پارامترهای مدل $\lambda(A,B,\pi)$
  - هدف: بیشینه کردن $p(O \mid \lambda)$ برای داده‌های آموزش
  - پارامترهایی که باید تخمین زده شود
    - A= احتمال انتقال بین حالت‌ها
    - π= احتمال اولیه حالت‌ها
    - B= پارامترهای تابع توزیع‌های حالت‌ها (میانگین و واریانس در صورت نرمال بودن توزیع)
  - روش: استفاده از تخمین بیشینه شباهت (ML)
    - اگر دنباله حالت‌ها را بدانیم، تخمین ساده می‌شود
    - مشکل عدم وجود دنباله حالت‌ها = متغیر (حالت) پنهان = استفاده از الگوریتم امید بیشینه (EM)
  - تخمین بهینه نیست: بهینه محلی

آموزش مدل . . .

- تخمین بیشینه شباهت (ML) و الگوریتم امید بیشینه (EM)
  - با استفاده از مقادیر فعلی پارامترها، مقادیر احتمال‌ها را محاسبه کنید (گام E)
  - تخمین مقادیر جدید برای پارامترها بر اساس مقادیر احتمال‌های محاسبه شده (گام M)

- تعریف
  - احتمال بودن در حالت i در زمان t و حالت j در زمان t+1 (با داشتن مدل جاری و دنباله مشاهده‌ها)

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j \mid \mathbf{O}, \lambda) = \frac{P(q_t = i, q_{t+1} = j, \mathbf{O} \mid \lambda)}{P(\mathbf{O} \mid \lambda)}$$

  - احتمال بودن در حالت i در زمان t (با داشتن مدل جاری و دنباله مشاهده‌ها)

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)$$

آموزش مدل . . .

- رابطه $\gamma_t(i)$ با $\alpha_t(i)$ و $\beta_t(i)$

$$\gamma_t(i) = P(q_t = i \mid \mathbf{O}, \lambda)$$

کل دنباله مشاهده

$$= \frac{P(\mathbf{O}, q_t = i \mid \lambda)}{P(\mathbf{O} \mid \lambda)}$$

نرمال کننده: جمع همه $\gamma_t(i)$ها برابر ۱ می‌شود

$$= \frac{P(\mathbf{O}, q_t = i \mid \lambda)}{\sum_{j=1}^{N} P(\mathbf{O}, q_t = j \mid \lambda)}$$

مربوط به بخشی از دنباله مشاهده $O_1 O_2 \ldots O_t$

بخشی دیگر از دنباله مشاهده $O_{t+1} \ldots O_T$

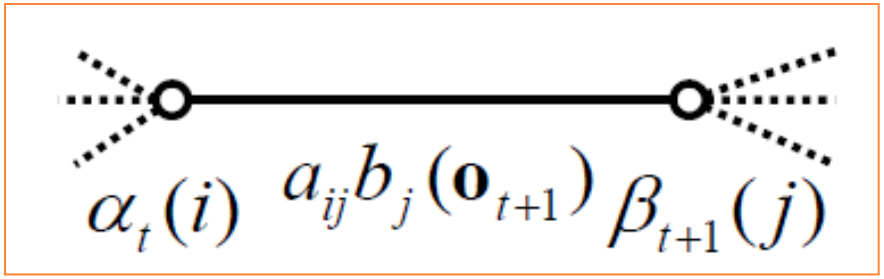
$$= \frac{\alpha_t(i)\beta_t(i)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}$$

آموزش مدل . . .

- محاسبه $\xi_t(i,j)$

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j \mid \mathbf{O}, \lambda) = \frac{\alpha_t(i) a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{P(\mathbf{O} \mid \lambda)}$$

$$= \frac{\alpha_t(i) a_{ij} b_j(\mathbf{o}_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i) \beta_t(i)}$$



محاسبه بر اساس پارامترهای الگوریتم‌های جلورو و عقب‌رو

آموزش مدل . . .

- با توجه به مقدار احتمال‌های $\xi_t(i,j)$ و $\gamma_t(i)$ داریم
  - $\xi_t(i,j)$ = احتمال بودن در حالت i در زمان t و حالت j در زمان t+1
  - $\gamma_t(i)$ = احتمال بودن در حالت i در زمان t

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)$$

- تخمین تعداد انتقال‌ها از حالت i در دنباله مشاهده T تایی = $\sum_{t=1}^{T-1} \gamma_t(i)$

- تخمین تعداد انتقال‌ها از حالت i به حالت j در دنباله مشاهده T تایی = $\sum_{t=1}^{T-1} \xi_t(i,j)$

- محاسبه پارامترها بر اساس این احتمال‌ها

آموزش مدل . . .

- احتمال اولیه حالت i
  - تعداد بارهایی که در زمان t=1 در حالت i هستیم

$$\bar{\pi}_i = \gamma_1(i)$$

تعداد حالت‌ها

$$\sum_{i=1}^{N} \bar{\pi}_i = 1$$

- احتمال انتقال از حالت i به حالت j

$$\bar{a}_{ij} = \frac{\text{expected number of transitions from state } i \text{ to state } j}{\text{expected number of transitions from state } i}$$

تعداد نمونه‌های آموزشی

$$= \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1}\sum_{j=1}^{N} \xi_t(i,j)} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$\sum_{j=1}^{N} \bar{a}_{ij} = 1 \qquad 1 \le i \le N$$

آموزش مدل . . .

تابع توزیع . . .

- تخمین برای حالت گسسته

$$\bar{b}_j(k) = \frac{\text{expected number of times in state } j \text{ and observing kth symbol}}{\text{expected number times in state } j}$$

$$= \frac{\sum_{t=1}^{T}\sum_{j=1}^{N}\xi_t(j,j)}{\sum_{t=1}^{T}\sum_{j=1}^{N}\xi_t(j,j)} = \frac{\sum_{t=1}^{T}\gamma_t(j)}{\sum_{t=1}^{T}\gamma_t(j)} = \frac{\sum_{t=1}^{T}\alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T}\alpha_t(j)\beta_t(j)}$$

$$\sum_{k=1}^{M}\bar{b}_j(k) = 1 \qquad 1 \le j \le N$$

- حالت پیوسته: هر حالت ترکیبی از M تابع توزیع است (یک GMM)
  - هر توزیعی می‌توان استفاده کرد از جمله توزیع نرمال

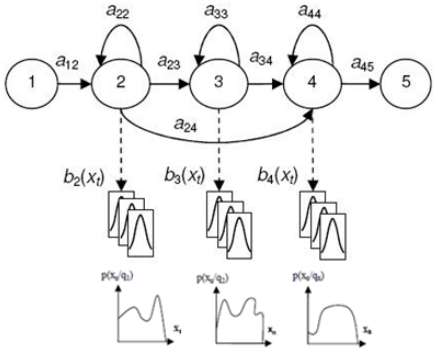
$$b_j(\mathbf{o_t}) = \sum_{k=1}^{M} c_{jk}\,\mathrm{N}(\mathbf{o_t}, \mu_{jk}, \Sigma_{jk})$$

وزن تلفیق = mixture coefficient

$$\sum_{k=1}^{M} c_{jk} = 1$$

$$c_{jk} \ge 0, \qquad 1 \le k \le M$$

تعداد توزیع‌های هر حالت



آموزش مدل . . .

- تابع توزیع (پیوسته) . . .
  - پارامترها (برای هر حالت)
    - وزن تلفیق‌ها
    - میانگین هر توزیع
    - کواریانس هر توزیع

$$b_j(\mathbf{o_t}) = \sum_{k=1}^{M} c_{jk} \mathrm{N}(\mathbf{o_t}, \mu_{jk}, \Sigma_{jk})$$

- $\gamma_t(i,k)$ = احتمال بودن در تلفیق k از حالت i در زمان t

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)} \right] \left[\frac{c_{jk} \mathrm{N}(\mathbf{o}_t, \mu_{jk}, \Sigma_{jk})}{\sum_{m=1}^{M} c_{jm} \mathrm{N}(\mathbf{o}_t, \mu_{jm}, \Sigma_{jm})} \right]$$

آموزش مدل . . .

- تابع توزیع (پیوسته)
  - توزیع نرمال

$$b_j(\mathbf{o_t}) = \sum_{k=1}^{M} c_{jk} \mathrm{N}(\mathbf{o_t}, \mu_{jk}, \Sigma_{jk})$$

$$\overline{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)}$$

$$\overline{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot \mathbf{o}_t}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)}$$

$$\overline{\Sigma}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \cdot (\mathbf{o}_t - \overline{\mu}_{jk})(\mathbf{o}_t - \overline{\mu}_{jk})'}{\sum_{t=1}^{T}\sum_{k=1}^{M} \gamma_t(j,k)}$$

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N} \alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}\mathrm{N}(\mathbf{o}_t, \mu_{jk}, \Sigma_{jk})}{\sum_{m=1}^{M} c_{jm}\mathrm{N}(\mathbf{o}_t, \mu_{jm}, \Sigma_{jm})}\right]$$

آموزش مدل

- الگوریتم آموزش: مبتنی بر امید بیشینه (EM)
- الگوریتم باوم-ولش (Baum-Welch) = الگوریتم جلورو-عقب‌رو
  - گام اول: مقداردهی اولیه پارامترها $\lambda(A,B,\pi)$ (احتمال اولیه، ماتریس انتقال حالت‌ها، پارامترهای توزیع مخلوط‌ها)
  - گام دوم: محاسبه مقادیر احتمال‌های α، β و ξ بر اساس مقدار موجود برای پارامترها $\lambda(A,B,\pi)$ (گام E)
  - گام سوم: محاسبه مقدار جدید پارامترها $\bar{\lambda} = (\bar{A},\bar{B},\bar{\pi})$ (گام M)
  - جایگزینی مقدار پارامترها با مقادیر جدید و تکرار الگوریتم
    - توقف: احتمال شباهت یا تغییرات آن به مقدار معینی رسیده باشد؛ یا حداکثر تعداد تکرار

- می‌توان نشان داد که $P(O|\bar{\lambda}) > P(O|\lambda)$

مدل مخفی مارکوف: ۳ مساله مهم

- **ارزیابی: محاسبه احتمال مشاهده**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان مقدار $p(O \mid \lambda)$ را محاسبه کرد؟

الگوریتم جلورو یا عقب‌رو

- **دیکدینگ: یافتن دنباله حالت‌ها**
  - با داشتن دنباله مشاهده $O=O_1O_2\dots O_T$ و مدل λ، چگونه می‌توان بهترین دنباله حالت‌های $Q=q_1q_2\dots q_T$ که متناسب با مشاهده است، را بدست آورد؟

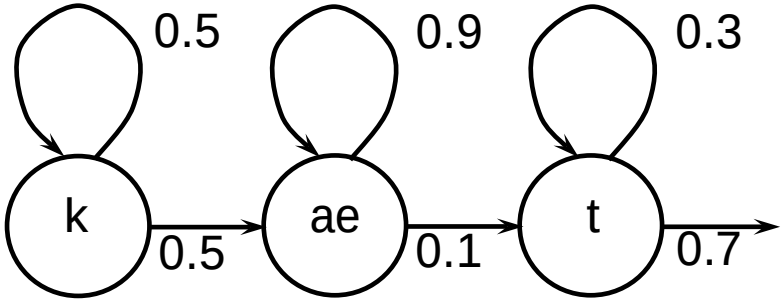
الگوریتم ویتربی

- **آموزش مدل**
  - پارامترهای مدل $\lambda(A,B,\pi)$ را چگونه بدست آوریم که $p(O \mid \lambda)$ بیشینه شود؟

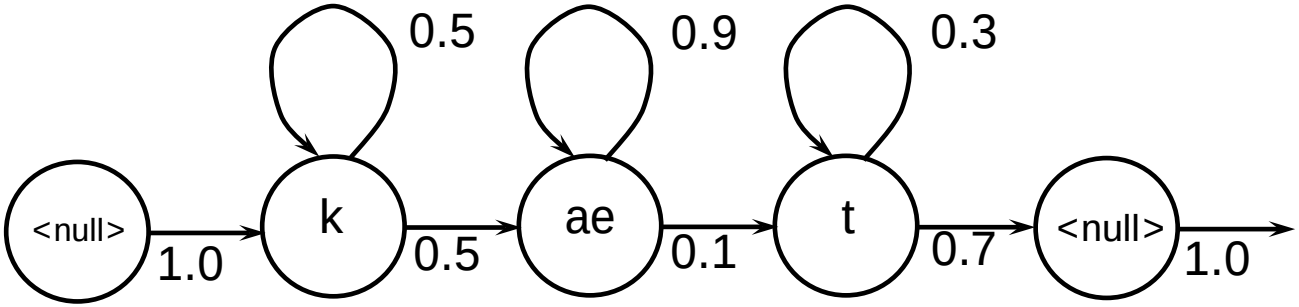
الگوریتم باوم-ولش (جلورو-عقب‌رو)

مثال: تشخیص گفتار . . .

- برای کلمه CAT
  - مشاهده‌ها: ویژگی‌های گفتار
  - مدل ۳ حالته
    - هر حالت معادل یک واج

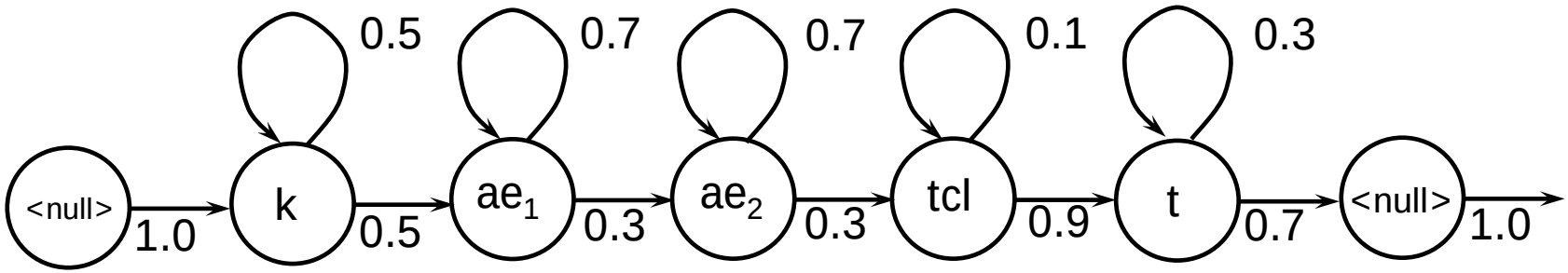


- مدل ۵ حالته برای کلمه CAT
  - حالت Null مشاهده تولید نمی‌کنند. از نظر تئوری لازم نیست اما پیاده‌سازی را راحت‌تر می‌کند



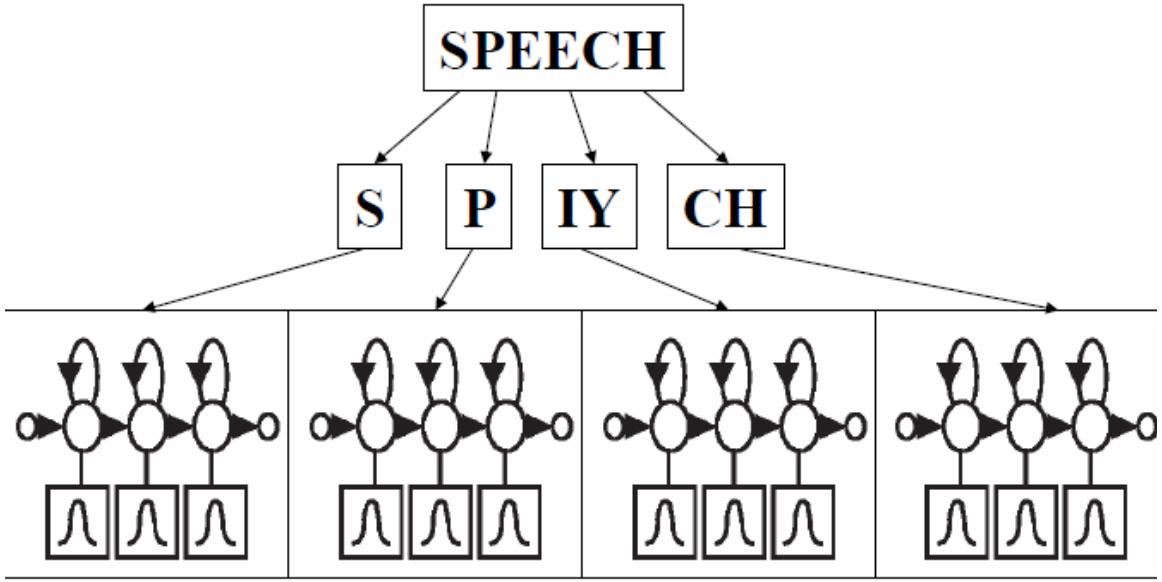
مثال: تشخیص گفتار . . .

- برای کلمه CAT
  - مدل ۷ حالته
  - حالت‌ها به صورت مستقیم معادل واج‌ها نیستند

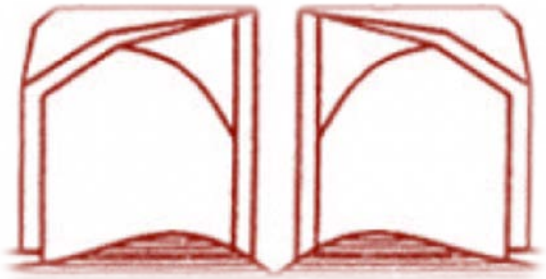


مثال: تشخیص گفتار

- مدل‌سازی گفتار (بازشناسی گفتار)
  - هر واج (phoneme) یک HMM دارد
  - تشکیل کلمه با پشت‌سرهم قرار دادن مدل واج‌ها



- قابلیت کشیدن یا فشرده‌کردن داده‌ها: مدل‌سازی داده‌های غیرهم‌طول



روش‌های یادگیری ماشین در پردازش زبان طبیعی

برچسب‌زنی اجزای کلام با مدل مخفی مارکوف

هادی ویسی

h.veisi@ut.ac.ir

دانشگاه تهران - دانشکده علوم و فنون نوین

اجزای کلام: Part-of-Speech (POS) . . .

- بیانگر مقوله نحوی که هر کلمه به آن تعلق دارد

- مثال
  - {من، تو، او، ...} یک {کتاب، گوسفند، درخت، ...} را {دیدم، خریدم، فروختم، ...}

- برچسپ‌زنی (Tagging) = POS Tagging
  - فرایند انتساب مقوله نحوی به هر کلمه در متن

  - ورودی: دنباله کلمات
    - Input: Plays well with others

  - خروجی: دنباله برچسپ‌ها
    - Output: Plays/VBZ well/RB with/IN others/NNS

برچسب‌زنی اجزای کلام با روش آماری . . .

- ایده
  - در نظر گرفتن احتمال وقوع برچسب‌ها (برچسب‌های محتمل) برای کلمات
  - با فرض داشتن دنباله کلمات $W=w_1 \ldots w_n$، دنباله برچسب‌های $T=t_1 \ldots t_n$ را طوری پیدا کنید که $P(T \mid W)$ بیشینه شود

$$\hat{T} = \arg\max_T P(T \mid W)$$

قانون بیز

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$

حذف $P(W)$ تغییری در بیشینه کردن ایجاد نمی‌کند

$$\hat{T} = \arg\max_T P(T \mid W) = \arg\max_T \frac{P(W \mid T)P(T)}{P(W)}$$

$$= \arg\max_T P(W \mid T)P(T) = \arg\max_T P(w_1 w_2 \cdots w_n \mid t_1 t_2 \cdots t_n)P(t_1 t_2 \cdots t_n)$$

Likelihood Prior

  - نحوه محاسبه؟

برچسب‌زنی اجزای کلام با روش آماری . . .

- برای محاسبه نیاز به فرض‌های ساده کننده است

$$\hat{T} = \arg\max_{T} P(w_1 w_2 \cdots w_n \mid t_1 t_2 \cdots t_n) P(t_1 t_2 \cdots t_n)$$

- فرض اول: احتمال وقوع یک کلمه فقط به برچسپ آن کلمه وابسته است و مستقل از سایر کلمات و برچسپ‌های اطراف آن است

$$P(w_1 w_2 \cdots w_n \mid t_1 t_2 \cdots t_n) \approx \prod_{i=1}^{n} P(w_i \mid t_i)$$

- فرض دوم: دیدن یک برچسپ فقط به برچسپ قبلی آن وابسته است (Bi-Gram)

$$P(t_1 t_2 \cdots t_n) \approx \prod_{i=1}^{n} P(t_i \mid t_{i-1})$$

- بنابر این داریم:

$$P(w_1 w_2 \cdots w_n \mid t_1 t_2 \cdots t_n) P(t_1 t_2 \cdots t_n) \approx \prod_{i=1}^{n} P(w_i \mid t_i) \cdot \prod_{i=1}^{n} P(t_i \mid t_{i-1})$$

برچسب‌زنی اجزای کلام با روش آماری . . .

- مفهوم احتمال $P(t_i \mid t_{i-1})$ (Tag Transition Probability)
  - احتمال آمدن یک برچسب (t_i) بعد از برچسپ دیگر (t_{i-1})
  - احتمال آمدن «اسم» (NN) یا «صفت» (JJ) بعد از «حرف تعریف» (DT) بالاست
    - The beautiful story
  - در پیکره Brown داریم: $P(NN \mid DT)=0.49$

- مفهوم احتمال $P(w_i \mid t_i)$ (Word Likelihood)
  - اگر دنبال کلمه‌ای با برچسپ t_i هستیم، احتمال اینکه آن کلمه w_i باشد
  - $P(is \mid VBZ)$= احتمال اینکه کلمه با برچسپ «VBZ» (فعل حال سوم شخص مفرد)، کلمه «is» باشد
    - در پیکره Brown داریم: $P(is \mid VBZ)=0.47$

برچسب‌زنی اجزای کلام با روش آماری . . .

محاسبه احتمال‌ها

- نیاز به یک پیکره متنی داریم که در آن کلمات دارای برچسپ باشند
- محاسبه احتمال $P(t_i \mid t_{i-1})$ (Tag Transition Probability)

$$P(t_i \mid t_{i-1}) = \frac{C(t_{i-1}t_i)}{C(t_{i-1})}$$

- مثال: برای محاسبه $P(NN \mid DT)$ (در پیکره Brown)- برچسپ DT به تعداد 116,454 آمده است که بعد از 56,509 مورد از آنها NN آمده است. پس

$$P(NN \mid DT) = \frac{56509}{116454} = 0.49$$

- محاسبه احتمال $P(w_i \mid t_i)$ (Word Likelihood)

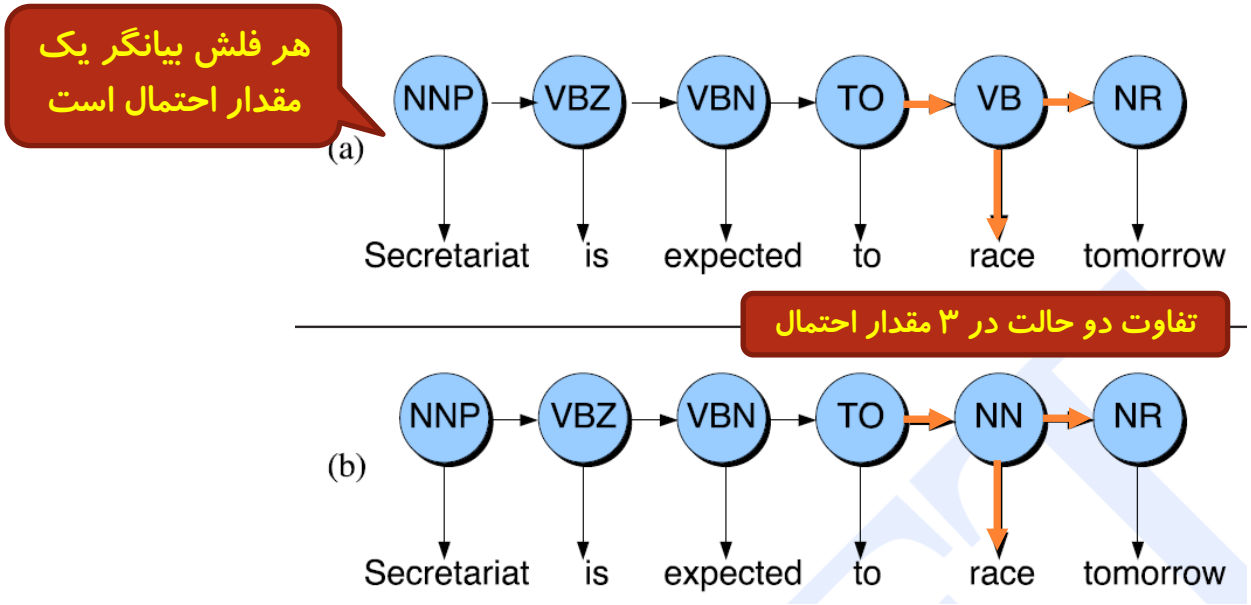
$$P(w_i \mid t_i) = \frac{C(w_i, t_i)}{C(t_i)}$$

- مثال: محاسبه $P(is \mid VBZ)$ (در پیکره Brown)- برچسپ «VBZ» به تعداد 21,627 بار آمده که از میان آنها، تعداد 10,073 کلمه «is» است. پس

$$P(is \mid VBZ) = \frac{10073}{21627} = 0.47$$

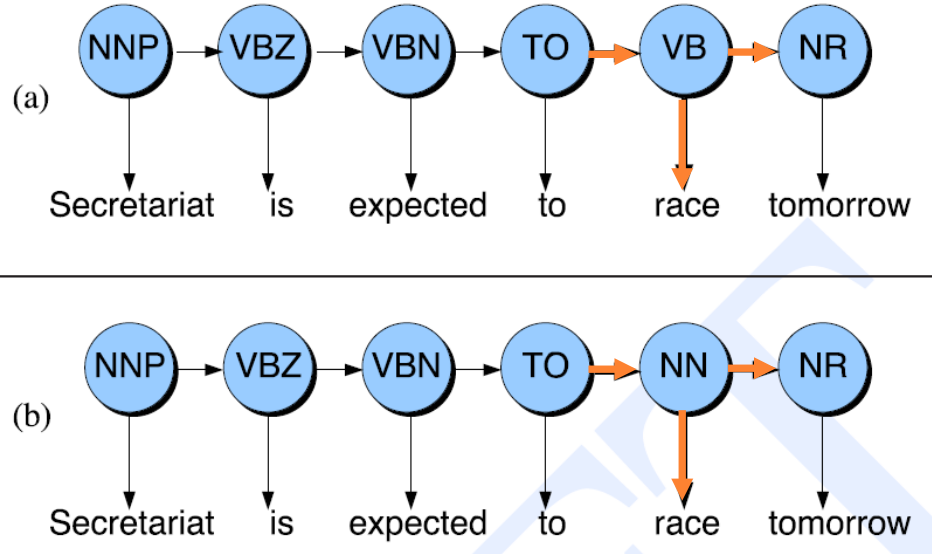
برچسب‌زنی اجزای کلام با روش آماری . . .

- مثال: محاسبه احتمال دنباله برای تعیین برچسپ درست
  - کلمه race هم می‌تواند فعل (VB) باشد و هم اسم (NN)
    - Secretariat/NNP is/BEZ expected/VBN to/TO race/VB tomorrow/NR
  - در نظر گرفتن دو حالت از دنباله حالات برای تعیین برچسپ race در جمله اول



برچسب‌زنی اجزای کلام با روش آماری . . .

- مثال: محاسبه احتمال دنباله برای تعیین برچسپ درست
  - Secretariat/NNP is/BEZ expected/VBN to/TO race/VB tomorrow/NR



$$P(\text{NN}|\text{TO}) = .00047$$
$$P(\text{VB}|\text{TO}) = .83$$

$$P(\text{NR}|\text{VB}) = .0027$$
$$P(\text{NR}|\text{NN}) = .0012$$

$$P(\text{race}|\text{NN}) = .00057$$
$$P(\text{race}|\text{VB}) = .00012$$

مقدار احتمال برای برچسپ VB بیشتر است

$$P(\text{VB}|\text{TO})P(\text{NR}|\text{VB})P(\text{race}|\text{VB}) = .00000027$$
$$P(\text{NN}|\text{TO})P(\text{NR}|\text{NN})P(\text{race}|\text{NN}) = .00000000032$$

- موارد ابهام دیگر
  - excepted می‌تواند صفت (JJ)، فعل گذشته (VBD) یا اسم مفعول (VBN) باشد

برچسب‌زنی اجزای کلام با روش آماری . . .

- محاسبه محتمل‌ترین دنباله از برچسپ‌ها
  - ساده‌ترین روش: درنظر گرفتن تمام دنباله‌های محتمل و محاسبه احتمال هر یک به روش بیان شده (Brute Force Search)
  - با فرض داشتن N برچسب و T کلمه، حداکثر N^T دنباله از برچسب‌ها تولید می‌شود.
    - محاسبات بسیار زیاد
  - روش‌های رایج
    - مدل مخفی مارکوف (HMM: Hidden Markov Model)
    - میدان تصادفی شرطی (CRF: Conditional Random Field)

برچسب‌زنی اجزای کلام با HMM . . .

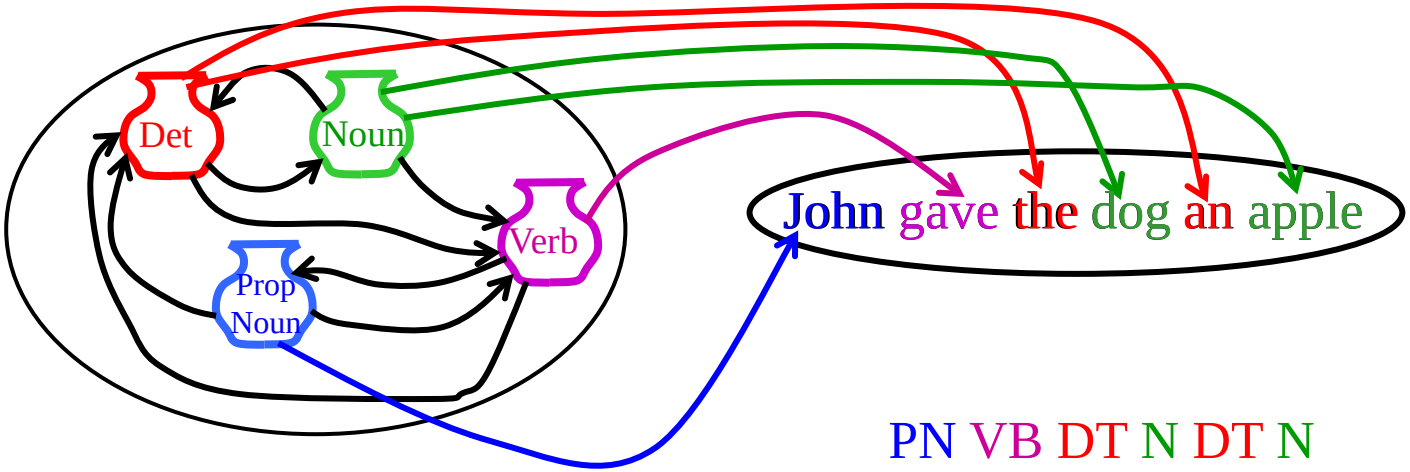
- مدل مخفی مارکوف در برچسپ‌زنی اجزای کلام
  1. مجموعه‌ای از N حالت = هر حالت بیانگر یک برچسپ
     - در گوی و گلدان: گلدان‌ها
  2. مجموعه‌ای از M نماد مشاهده = هر مشاهده بیانگر یک کلمه
     - در گوی و گلدان: رنگ‌ها
  3. احتمال انتقال حالت‌ها = احتمال وقوع یک برچسپ بعد از دیگری
     - در گوی و گلدان: جابجایی از یک گلدان به گلدان دیگر
  4. احتمال اولیه حالت‌ها = احتمال اینکه اولین کلمه چه برچسپی داشته باشد
     - در گوی و گلدان: احتمال انتخاب هر کدام از گلدان‌ها در اولین مشاهده (زمان $t=1$)
  5. تابع توزیع برای مشاهده kام در حالت jام = احتمال اینکه برچسپ jام کلمه kام باشد
     - در گوی و گلدان: احتمال انتخاب گوی kام از گلدان jام
     - تابع توزیع مشاهده‌ها (مثلاً گاوسی)- احتمال تولید مشاهده $o_t=v_k$ در حالت $q_t=j$

برچسب‌زنی اجزای کلام با HMM . . .

- **برچسپ‌زنی اجزای کلام با روش HMM**
  - **فاز آموزش**
    - در نظر گرفتن یک واژگان با M کلمه و مجموعه برچسپ‌های ممکن (N برچسپ)
    - در نظر گرفتن یک HMM با تعداد حالت‌های برابر با تعداد برچسپ‌ها (N حالت)
    - محاسبه احتمال‌های مدل با استفاده از یک پیکره متنی دارای برچسپ اجزای کلام برای کلمات
      - احتمال اولیه حالت‌ها (N مقدار: هر حالت یک عدد)
      - احتمال انتقال از یک حالت (برچسپ) به حالت دیگر (یک ماتریس N*N)
      - احتمال داشتن هر برچسپ برای هر کلمه (یک ماتریس M*N)
  - **فاز آزمون (استفاده)**
    - دریافت یک دنباله از کلمات
    - یافتن بهترین برچسپ‌های مرتبط= بهترین دنباله حالت در مدل HMM

برچسب‌زنی اجزای کلام با HMM . . .

- برچسپ‌زنی اجزای کلام با روش HMM



برچسب‌زنی اجزای کلام با HMM . . .

- یافتن دنباله حالت‌های (برچسپ‌های) بهینه برای یک دنباله از کلمات
  - با داشتن دنباله مشاهده (کلمات) $O=O_1O_2...O_T$ و مدل مخفی کارکوف λ، چگونه می‌توان بهترین دنباله حالت‌های (برچسپ‌ها) $Q=q_1q_2...q_T$ که متناسب با مشاهده است، را بدست آورد؟

  - مساله دیکدینگ در HMM (مساله دوم)
  - راه‌حل کامل: بررسی تمام دنباله حالت‌های ممکن و انتخاب بهترین آنها
    - بسیار زمان‌بر، از مرتبه $O(TN^T)$ که N تعداد حالت‌ها (برچسپ‌ها) و T طول دنباله مشاهده‌ها (کلمات) است

  - راه حل بهینه: الگوریتم ویتربی (Viterbi)
    - یک روش برنامه نویسی پویا (Dynamic Programming)
      - مشابه الگوریتم Minimum Edit Distance

برچسب‌زنی اجزای کلام با HMM . . .

الگوریتم ویتربی

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

create a path probability matrix *viterbi*$[N+2,T]$

for each state s **from** 1 **to** N **do** ;initialization step

$viterbi[s,1] \leftarrow a_{0,s} * b_s(o_1)$

$backpointer[s,1] \leftarrow 0$

for each time step t **from** 2 **to** T **do** ;recursion step

for each state s **from** 1 **to** N **do**

$$viterbi[s,t] \leftarrow \max_{s'=1}^{N} viterbi[s',t-1] * a_{s',s} * b_s(o_t)$$

$$backpointer[s,t] \leftarrow \operatorname{argmax}_{s'=1}^{N} viterbi[s',t-1] * a_{s',s}$$

$$viterbi[q_F,T] \leftarrow \max_{s=1}^{N} viterbi[s,T] * a_{s,q_F} \quad \text{; termination step}$$

$$backpointer[q_F,T] \leftarrow \operatorname{argmax}_{s=1}^{N} viterbi[s,T] * a_{s,q_F} \quad \text{; termination step}$$

return the backtrace path by following backpointers to states back in time from $backpointer[q_F,T]$

ماتریس احتمال‌ها

ماتریس دنباله حالت‌ها

احتمال اولیه حالت s

احتمال اینکه کلمه o_t دارای حالت s باشد

برچسب‌زنی اجزای کلام با HMM . . .

- مثال . . .
  - جمله: I want to race
  - ۴ برچسپ (حالت)
    - VB، TO، NN و PPSS
  - آموزش: محاسبه احتمال‌ها

Transition probabilities: $P(t_i|t_{i-1})$

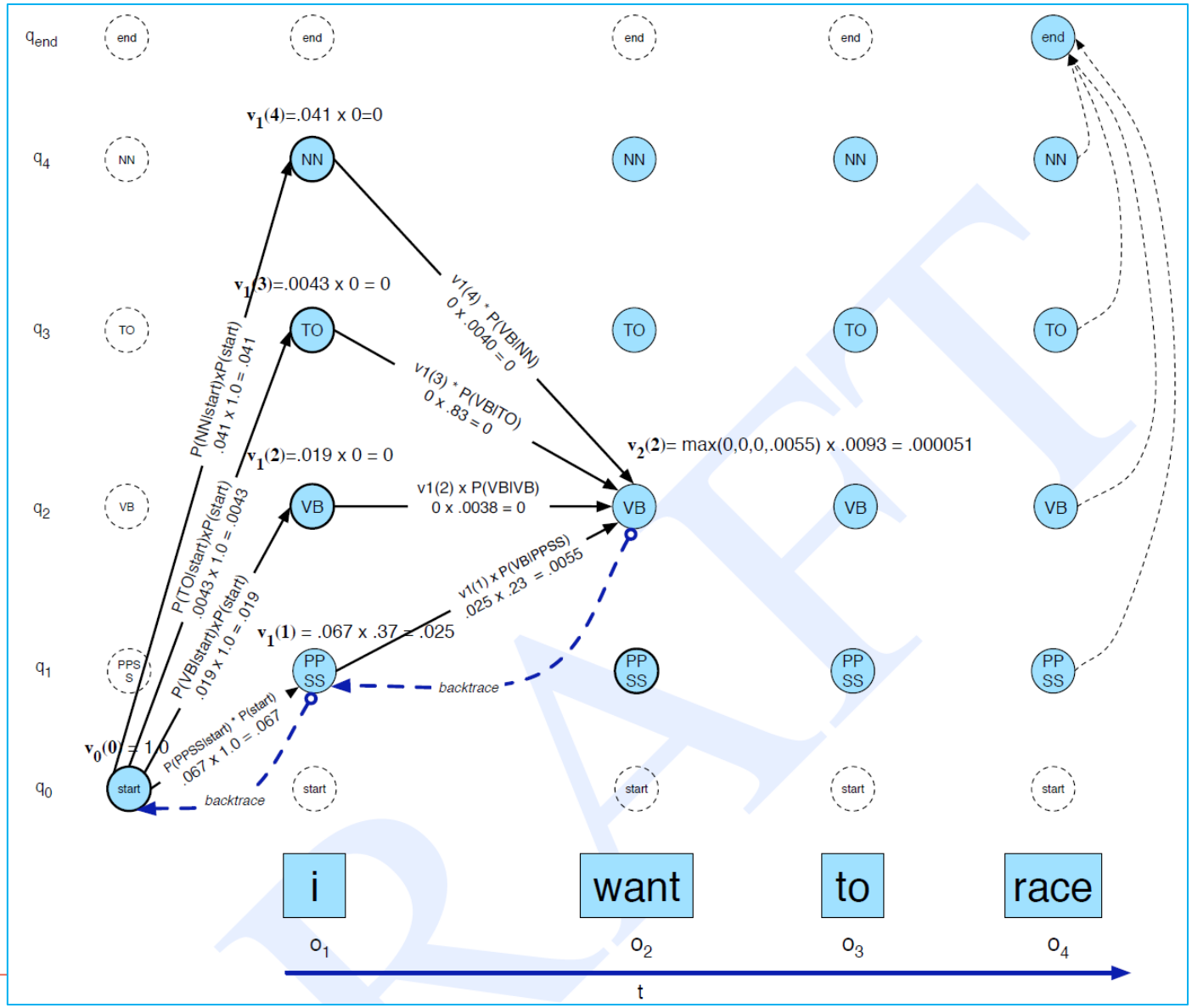
	VB	TO	NN	PPSS
start	0.019	0.0043	0.041	0.067
VB	0.0038	0.0345	0.047	0.070
TO	0.83	0	0.00047	0
NN	0.0040	0.016	0.087	0.0045
PPSS	0.23	0.00079	0.0012	0.00014

Observation likelihoods: $P(w_i|t_i)$

	I	want	to	race
VB	0	0.0093	0	0.00012
TO	0	0	0.99	0
NN	0	0.000054	0	0.00057
PPSS	0.37	0	0	0

برچسب‌زنی اجزای کلام با HMM

- مثال



برچسب‌زنی (با HMM): سایر کاربردها . . .

- بازشناسی موجودیت اسمی (NER)
  - یافتن اسم‌های خاص (اشخاص، مکان، سازمان، ...) در متن
    - حسن روحانی
    - خیابان ولیعصر
    - شرکت مخابرات ایران

INPUT:

Profits soared at Boeing Co., easily topping forecasts on Wall Street, as their CEO Alan Mulally announced first quarter results.

OUTPUT:

Profits/NA soared/NA at/NA Boeing/SC Co./CC ,/NA easily/NA topping/NA forecasts/NA on/NA Wall/SL Street/CL ,/NA as/NA their/NA CEO/NA Alan/SP Mulally/CP announced/NA first/NA quarter/NA results/NA ./NA

NA = No entity
SC = Start Company
CC = Continue Company
SL = Start Location
CL = Continue Location
. . .

برچسب‌زنی (با HMM): سایر کاربردها . . .

- بخش‌بندی عبارات اسمی (BaseNP Chunking)
  - یافتن عبارات اسمی مرتبط با یک مفهوم (استفاده در تجزیه و بازایابی اطلاعات)
    - انتساب برچسب‌های (B: شروع عبارت، I: کلمات میانی عبارت و O: سایر کلمات عبارت)
    - [the student] said [the exam question] is hard
    - the/B student/I said/O the/B exam/I question/I is/O hard/O

INPUT:
Profits soared at Boeing Co., easily topping forecasts on Wall Street, as their CEO Alan Mulally announced first quarter results.

OUTPUT:

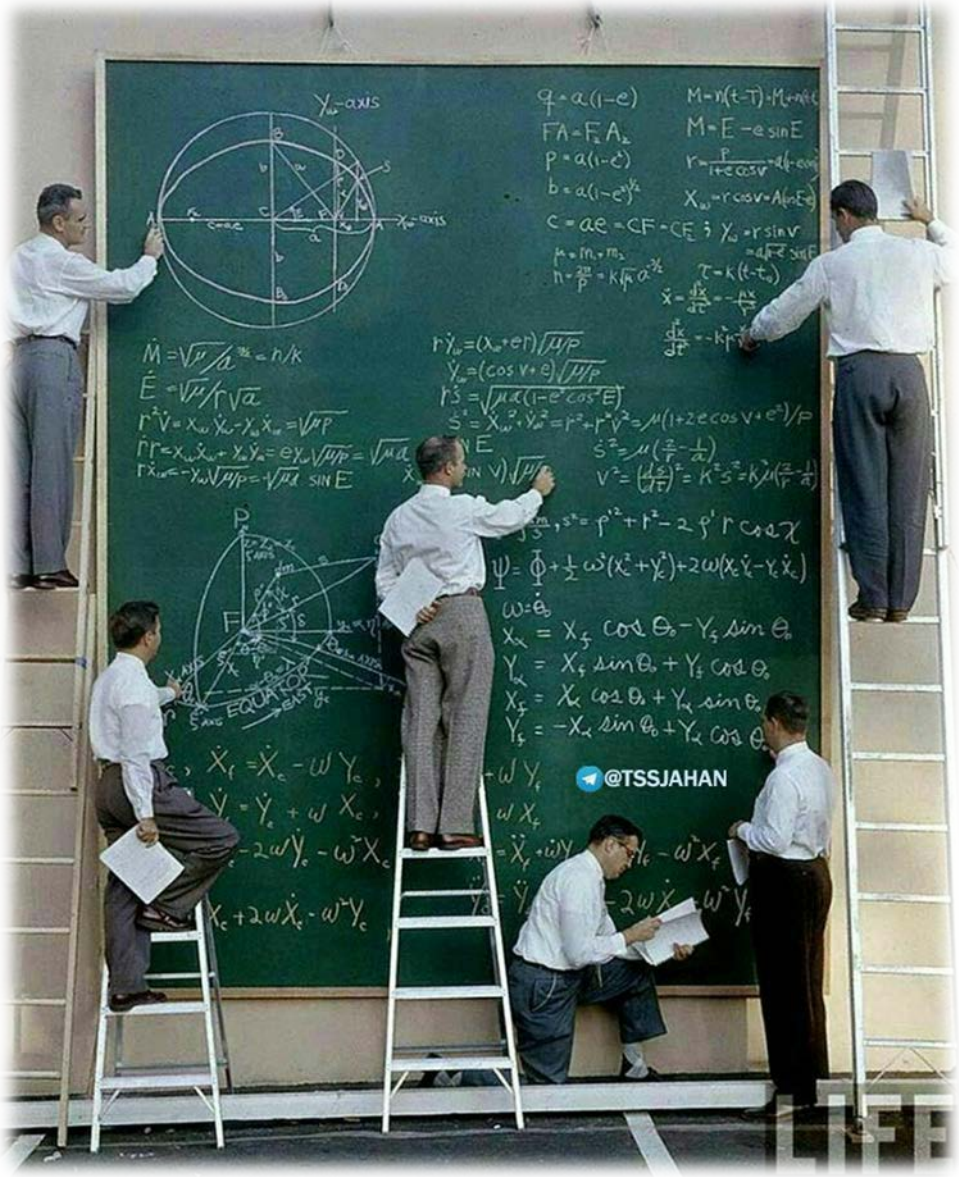
Profits/S soared/N at/N Boeing/S Co./C ,/N easily/N topping/N forecasts/S on/N Wall/S Street/C ,/N as/N their/S CEO/C Alan/C Mulally/C announced/N first/S quarter/C results/C ./N

N = Not part of noun-phrase
S = Start noun-phrase
C = Continue noun-phrase

[NP Profits] soared at [NP Boeing Co.], easily topping [NP forecasts] on [NP Wall Street], as [NP their CEO Alan Mulally] announced [NP first quarter results].

برچسب‌زنی (با HMM): سایر کاربردها

- بازسازی حالت (Case Restoration)
  - در متن (انگلیسی) فقط با حروف کوچک، نیاز به تبدیل حرف اول برخی کلمات به حرف بزرگ (مانند اسامی خاص)
  - یکی از ویژگی‌ها در تشخیص موجودیت اسمی (NER)


@TSSJAHAN