

Sequential Learning

WHAT IS SEQUENTIAL LEARNING?

Topics from class

- Classification learning: learn $\mathbf{x} \rightarrow y$
 - Linear (naïve Bayes, logistic regression, ...)
 - Nonlinear (neural nets, trees, ...)
- Not quite classification learning:
 - Regression (y is a number)
 - Clustering, EM, graphical models, ...
 - there is no y , so build a distributional model the instances
 - Collaborative filtering/matrix factoring
 - many linked regression problems
 - **Learning for sequences: learn $(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k) \rightarrow (y_1, \dots, y_k)$**
 - special case of “structured output prediction”

A sequence learning task: named entity recognition (NER)

person company jobTitle

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...



October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...

Name entity recognition (NER) is one part of information extraction (IE)

person company jobTitle

October 14, 2002, 4:00 a.m. PT

For years, [Microsoft Corporation](#) [CEO Bill Gates](#) railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

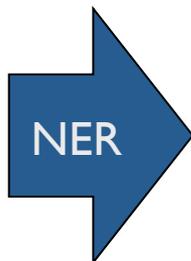
Today, [Microsoft](#) claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. [Gates](#) himself says [Microsoft](#) will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said [Bill Veghte](#), a [Microsoft VP](#). "That's a super-important shift for us in terms of code access."

[Richard Stallman](#), [founder](#) of the [Free Software Foundation](#), countered saying...



<u>NAME</u>	<u>TITLE</u>	<u>ORGANIZATION</u>
Bill Gates	CEO	Microsoft
Bill Veghte	VP	Microsoft
Richard St...	founder	Free Soft..



IE Example: Job Openings from the Web

OPUS International, Inc., an executive search firm focusing on the Food Science industry. - Microsoft Internet Explorer

File Edit View Favorites

Back Forward Stop

Address http://www.foodscience.com/jobs_midwest.html#top

Links AMEX Rewards Time DogHouse My Yahoo!

Welcome

About OPUS

Executive Staff

Job Listings

Résumé Form

Job Hunt Hints

Academic Links

Science Fair Help

Industry Assocs.

FAQs

Contact Us

Site Map

e-mail

OPUS INTERNATIONAL INC.

About | Staff | Job

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorite

Address http://www.foodscience.com/jobs_midwest.html#top

Links AMEX Rewards Time DogHouse My Yahoo!

Welcome

About OPUS

Executive Staff

Job Listings

Résumé Form

Job Hunt Hints

Academic Links

Science Fair Help

Industry Assocs.

FAQs

Contact Us

Site Map

e-mail

Test Kitchen-Consumer Food Relations

Major food manufacturer in Chicago area seeks a consumer food professional to write recipes. Will make presentations; will be a key player in a cross-functional team. Requires a BS in human ecology, nutrition, Food Science, or related field with a minimum three years' and experience.

Contact: Moira: [e-mail](mailto:email)
1-800-488-2611

Ice Cream Guru

If you dream of cold creamy chocolate or goochy boochy cookie, there's a great opportunity for you to maintain and expand this major corporation's high-end ice cream brand. Will be based in the Upper Midwest for about a year. After that, California here I come! Requires a BS in Food Science or dairy, plus ice cream formulation experience. Will consider entry level with an MS and an internship.

Contact: Susan: [e-mail](mailto:email)
1-800-488-2611

foodscience.com-Job2

JobTitle: Ice Cream Guru

Employer: foodscience.com

JobCategory: Travel/Hospitality

JobFunction: Food Services

JobLocation: Upper Midwest

Contact Phone: 800-488-2611

DateExtracted: January 8, 2001

Source: www.foodscience.com/jobs_midwest.htm

OtherCompanyJobs: foodscience.com-Job1



IE Example: A Job Search Site

job search find employment careers @ FlipDog.com free! - Microsoft Internet Explorer

Address <http://www.flipdog.com/home.html> Go File Edit View Favorites Tools Help Links



Home Find Jobs Your Account Resource Center Support Employers

Job Search at FlipDog.com: Employment & Career Management



647,514
Job Opportunities
from **53,641** Employers

Find a Job!

Post Your Resume

Employers
click here for
Products & Services



Pigskin Places

- Health Care in NY [2,770](#)
- Health Care in MD [1,262](#)
- Sales in NY [3,751](#)
- Sales in MD [958](#)
- Computing in NY [8,050](#)
- Computing in MD [4,114](#)

Jobs for Sports Fans

- [Head Football Coach](#)
- [Football Coach](#)
- [Asst. Football Coach](#)
- [High School Football Coach](#)
- [Univ. Asst. Football Coach](#)

Job Seeker Newsletter

Enter your e-mail address:

[Sign Me Up!](#)

Showcase Jobs



Management Recruiters
of Charlotte North

We provide total staffing solutions in the areas of Human Resources, Compensation, Web-based HR self-service, and Customer Management Systems.

[Learn More](#)



Looking for a Vice President of Academic Affairs to oversee planning, operation and evaluation of the college's academic programs.

[Learn More](#)

Job Seekers: Find your dream job!

- Check our 'Best Places to Find a Job' [January report](#).
- Open your [FREE account](#) and put your [resume online](#).
- Search 24x7 with our FREE automatic [JobHunters™](#).
- Research our database of over [50,000 employers](#).
- Get [expert advice](#) at our new [Resource Center](#).
- Access [salary surveys/calculators](#), [relocation tools](#), [networking opportunities](#), & [training/testing](#) tools.
- Use FlipDog.com to search jobs right from your desktop! Download [Snippets](#) today!



"Top 100 Web Sites"
PC Magazine, Nov. 2000



"Top 10 Career Web Site"
Media Metrix, Sept. 2000



"Top 10 Job Site"

powered by **WhizBang!**

Start  Internet

Microsoft PowerPoint - [sta... job search find employem...

12:12 AM

How can we do NER?

person company jobTitle

October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...



October 14, 2002, 4:00 a.m. PT

For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

Today, Microsoft claims to "love" the open-source concept, by which software code is made public to encourage improvement and development by outside programmers. Gates himself says Microsoft will gladly disclose its crown jewels--the coveted code behind the Windows operating system--to select customers.

"We can be open source. We love the concept of shared source," said Bill Veghte, a Microsoft VP. "That's a super-important shift for us in terms of code access."

Richard Stallman, founder of the Free Software Foundation, countered saying...

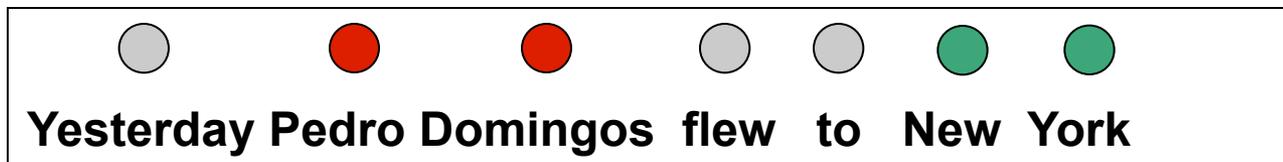
Most common approach: NER by classifying tokens

Given a sentence:

Yesterday Pedro Domingos flew to New York.

1) Break the sentence into *tokens*, and **classify** each token with a label indicating *what sort of entity* it is part of:

	person name
	location name
	background



2) Identify names based on the entity labels

Person name: **Pedro Domingos**
Location name: **New York**

3) To learn an NER system, use YFCL and whatever features you want....

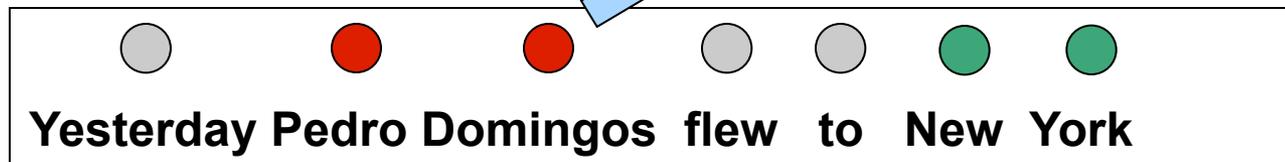
Most common approach: NER by classifying tokens

Feature	Value
isCapitalized	yes
numLetters	8
suffix2	-os
word-1-to-right	flew
word-2-to-right	to
...	

Given a sentence:

Yesterday Pedro Domingos flew to New

1) Break the sentence into *tokens*, and **classify** each token with a label indicating *what sort of entity* it is part of:



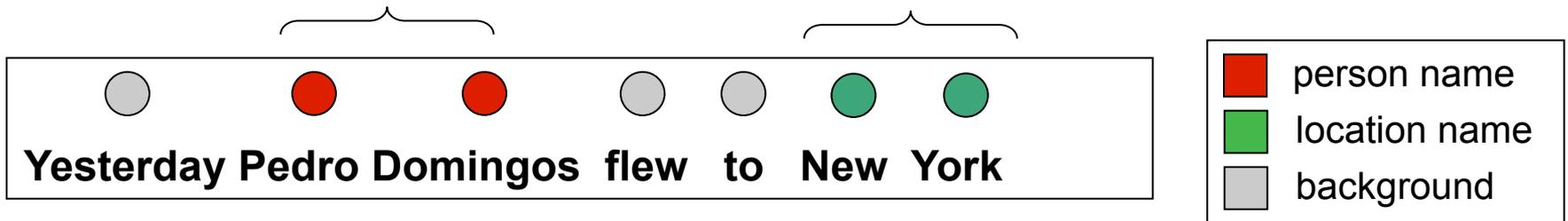
2) Identify names based on the entity labels

Person name: **Pedro Domingos**
Location name: **New York**

3) To learn an NER system, use YFCL and whatever features you want....

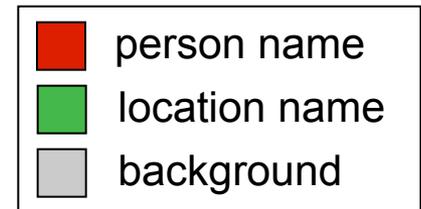
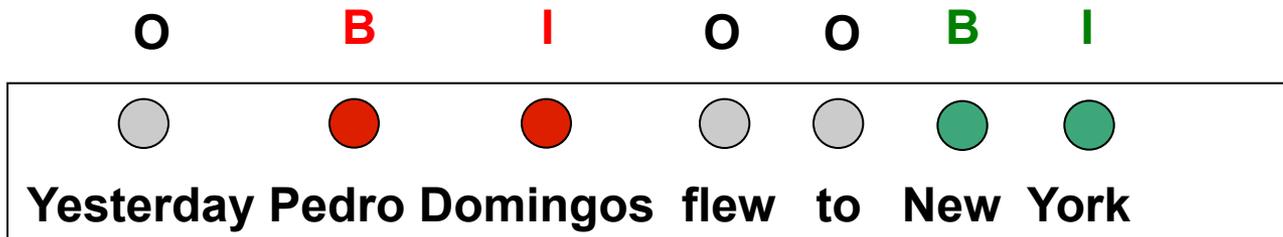
NER by classifying tokens

A Problem/Opportunity: YFCL assumes examples are iid.
But similar labels tend to *cluster together* in text



How can you model these dependencies?

NER by classifying tokens



Another common labeling scheme is **BIO** (begin, inside, outside; e.g. beginPerson, insidePerson, beginLocation, insideLocation, outside)

- **Begin** tokens are different from other name tokens
- “Tell **William Travis** is handling it”

BIO also leads to *strong dependencies between nearby labels* (eg **inside** follows **begin**).

How can you model these dependencies?

A hidden Markov model (HMM): the “naïve Bayes” of sequences

Other nice problems for HMMS

Parsing addresses

House number	Building	Road	City	State	Zip
4089	Whispering Pines	Nobel Drive	San Diego	CA	92122

Parsing citations

Author

Year

P.P.Wangikar, T.P. Graycar, D.A. Estell, D.S. Clark, J.S. Dordick (1993)
Protein and Solvent Engineering of Subtilising BPN' in Nearly
Anhydrous Organic Media J.Amer. Chem. Soc. 115, 12231-12237.

Title

Journal

Volume

Other nice problems for HMMS

Sentence segmentation: Finding words (to index) in Asian languages

第二阶段的奥运会体育比赛门票与残奥会开闭幕式门票的预订工作已经结束,现在进入门票分配阶段。在此期间,我们不再接受新的门票预订申请。

Morphology: Finding components of a single word

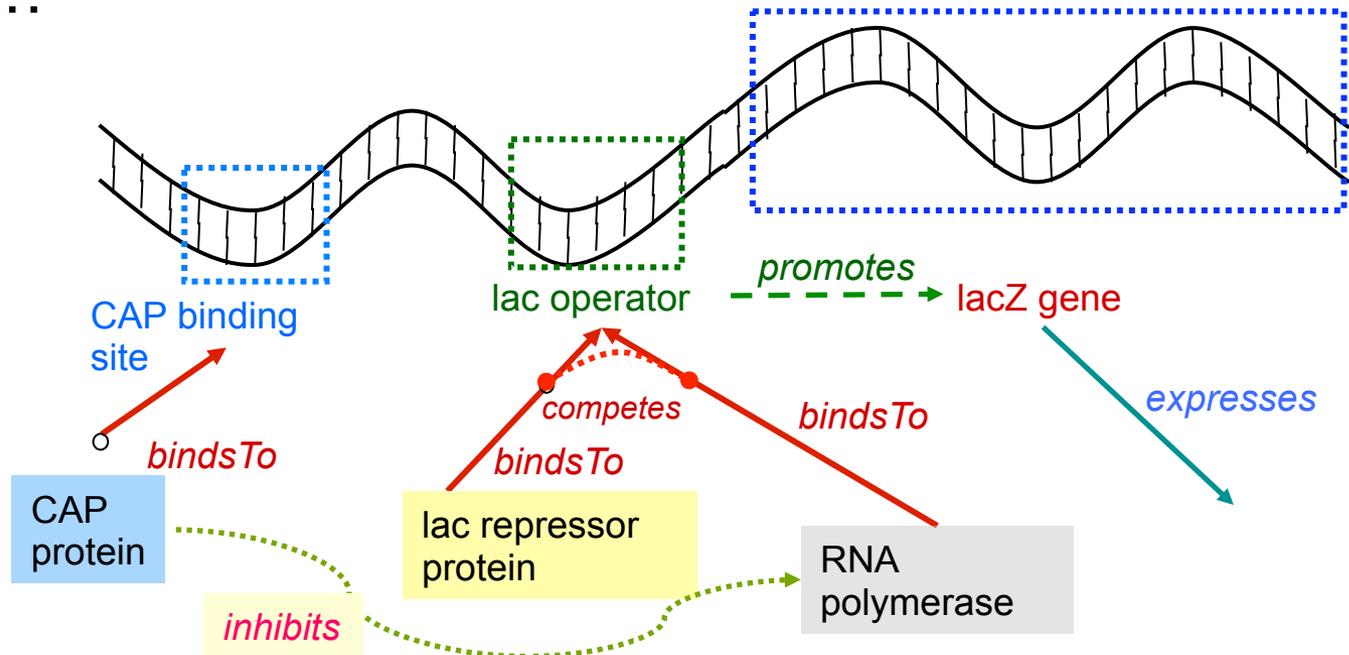
uygarlaştıramadıklarımızdanmışsınızcasına, or “(behaving) as if you are among those whom we could not civilize”

- Document analysis: finding tables in plain-text documents
- Video segmentation: splitting videos into naturally meaningful sections
- Converting text to speech (TTS)
- Converting speech to text (ASR)
- ...

Other nice problems for HMMS

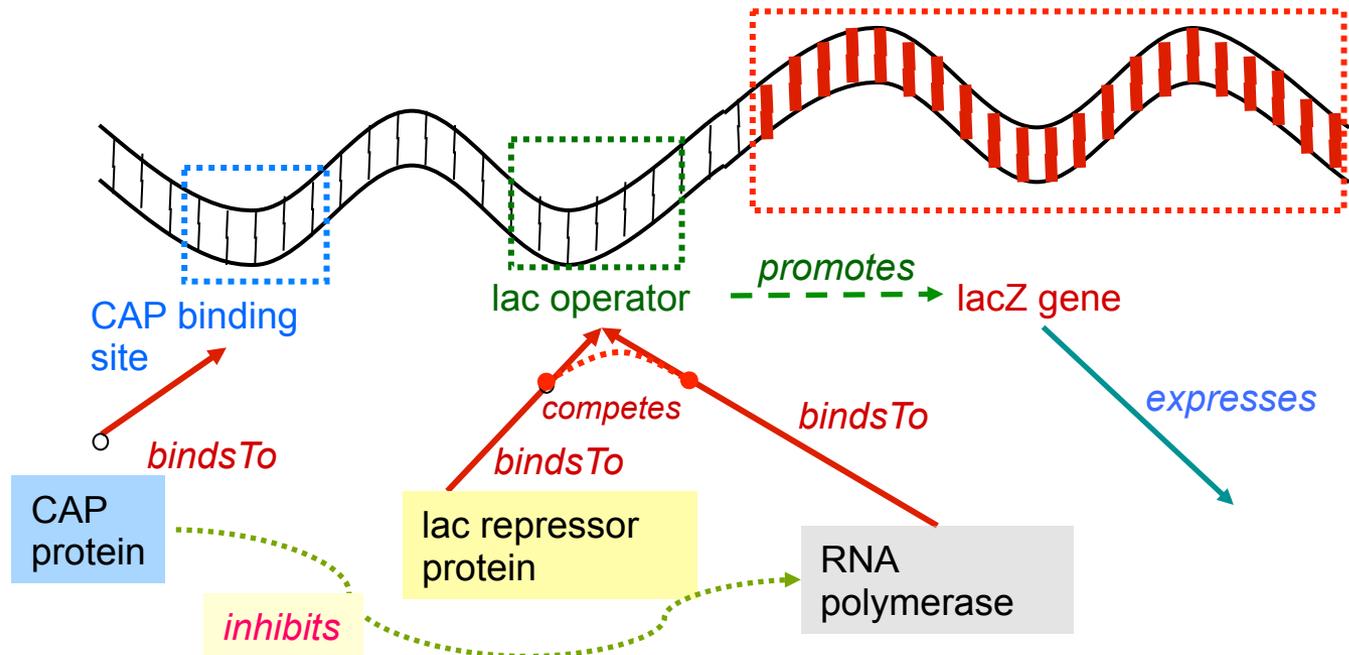
- Modeling **biological sequences**

- e.g., segmenting DNA into genes (transcribed into proteins or not), promoters, TF binding sites, ...
- identifying variants of a single gene
- ...



Other nice problems for HMMS

- Eg **gene finding**: which parts of DNA are genes, versus binding sites for gene regulators, junk DNA, ... ?

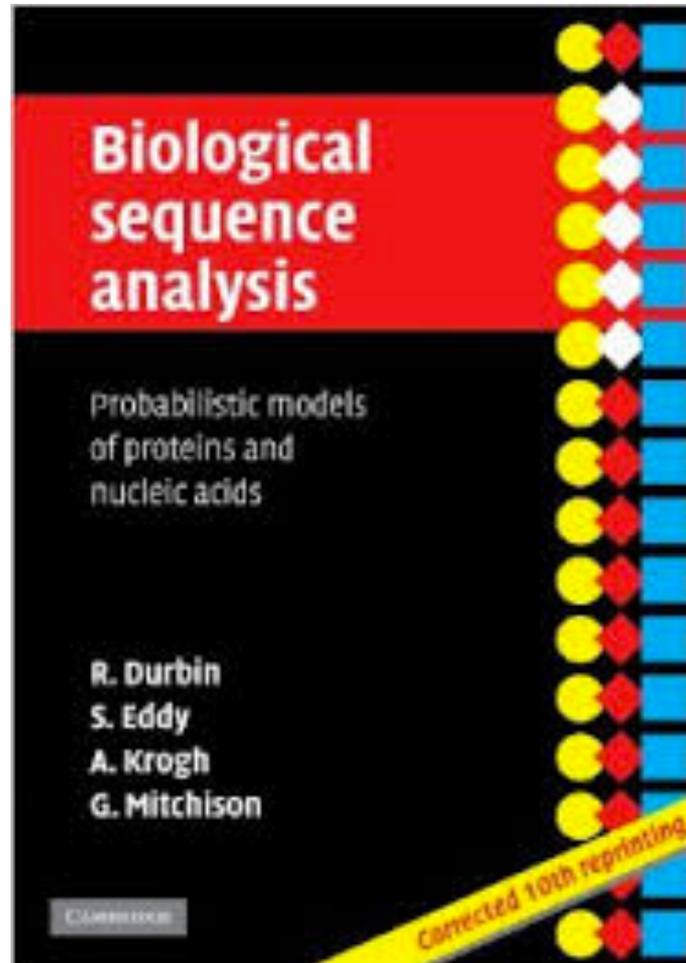


[BOOK] **Biological sequence analysis: probabilistic models of proteins and nucleic acids**

[R Durbin](#), [SR Eddy](#), [A Krogh](#), [G Mitchison](#) - 1998 - [books.google.com](#)

Probabilistic models are becoming increasingly important in analysing the huge amount of data being produced by large-scale DNA-sequencing efforts such as the Human Genome Project. For example, hidden Markov models are used for analysing **biological** sequences, ...

[Cited by 5878](#) [Related articles](#) [All 15 versions](#) [Cite](#) [Saved](#) [More](#)



Aside: relax, we will not test you on biology for this class 😊

Sequence alignment for proteins (done by “pair HMMs”)

R Y D S R T T I F S P . . E G R L Y Q V E Y A M E A I G N A . G S A I G I L S
R Y D S R T T I F S P L R E G R L Y Q V E Y A M E A I S H A . G T C L G I L S
R Y D S R T T I F S P . . E G R L Y Q V E Y A Q E A I S N A . G T A I G I L S
R Y D S R T T I F S P . . E G R L Y Q V E Y A M E A I S H A . G T C L G I L A
R Y D S R T T I F S P . . E G R L Y Q V E Y A M E A I G H A . G T C L G I L A
R Y D S R T T I F S P . . E G R L Y Q V E Y A M E A I G N A . G S A L G V L A
R Y D S R T T T F S P . . E G R L Y Q V E Y A L E A I N N A . S I T I G L I T
S Y D S R T T I F S P . . E G R L Y Q V E Y A L E A I N H A . G V A L G I V A

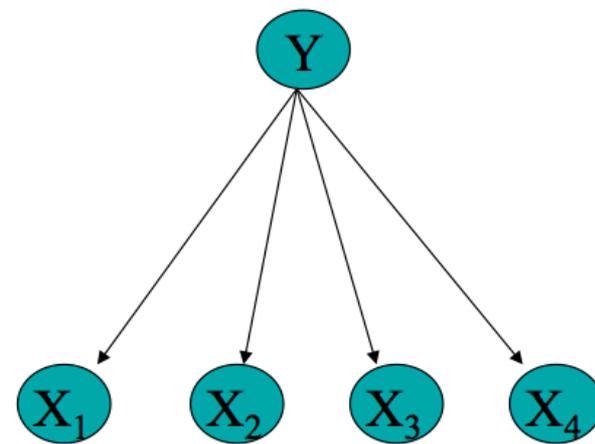
HMM warmup: a model of aligned sequences

```

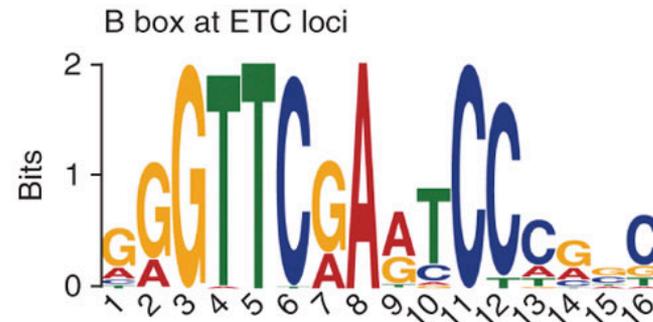
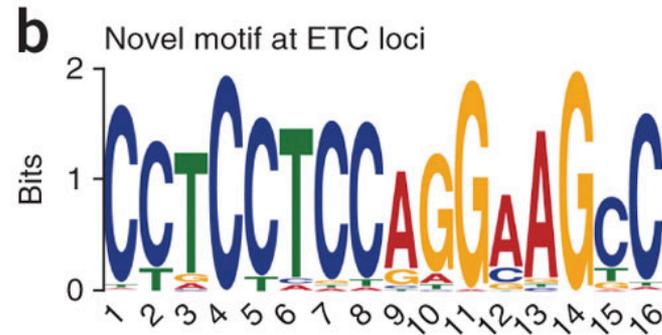
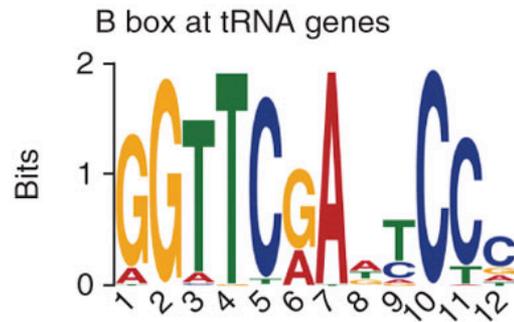
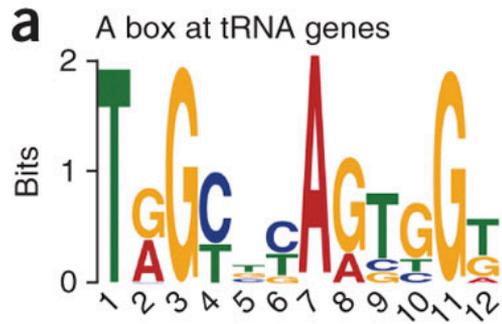
GNA . GSA IG
SHA . GTC LG
SNA . GTA IG
SHA . GTC LG
GHA . GTC LG
GNA . GSALG
NNA . SIT IG
NHA . GVALG
    
```

	S1	S2	S3	...
A	0.01	0.03	0.89
G	0.3	0.01	0.01	...
H	0.01	0.5	0.01	...
N	0.2	0.4	0.01	...
S	0.3	0.01	0.01	...
...				

Learn $P(Y|X)$



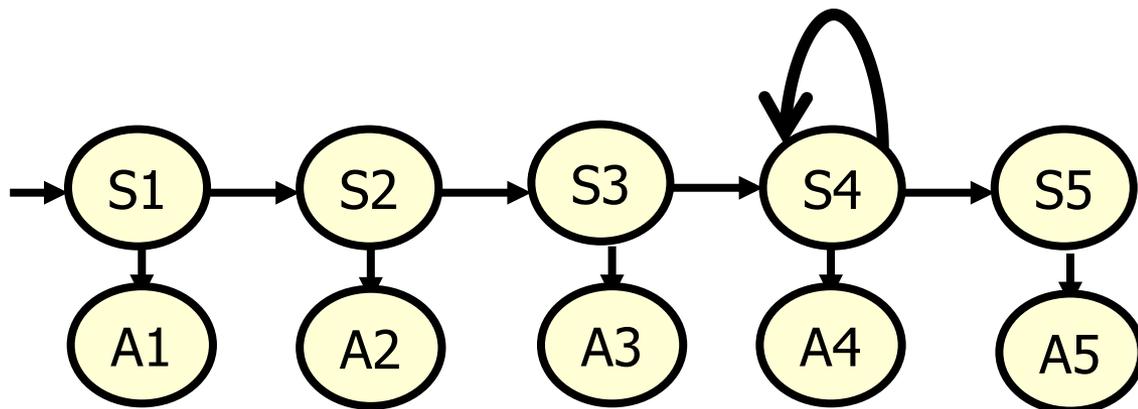
E.g.: Motifs



HMM warmup: a model of aligned sequences

```

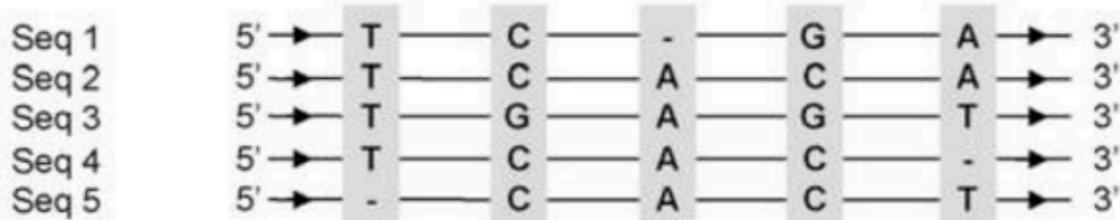
GNA . GSA IIG
SHA . GTC LIG
SNA . GTA IIG
SHA . GTC LIG
GHA . GTC LIG
GNA . GSA LIG
NNA . SIT IIG
NHA . GVA LIG
  
```



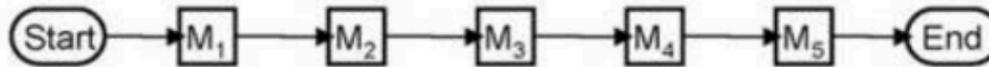
	S1	S2	S3	S4	S5
A	0.01	0.03	0.89	0.05	0.01
G	0.3	0.01	0.01	0.05	0.82
H	0.01	0.5	0.01	0.05	0.01
N	0.2	0.4	0.01	0.05	0.01
S	0.3	0.01	0.01	0.05	0.15
...					

Profile HMMs

(a) Sequence Alignment

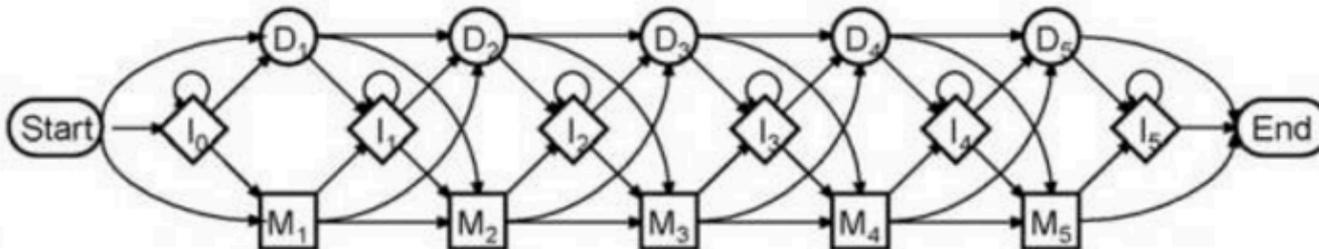


(b) Ungapped HMM



M_k Match states

(c) Profile-HMM

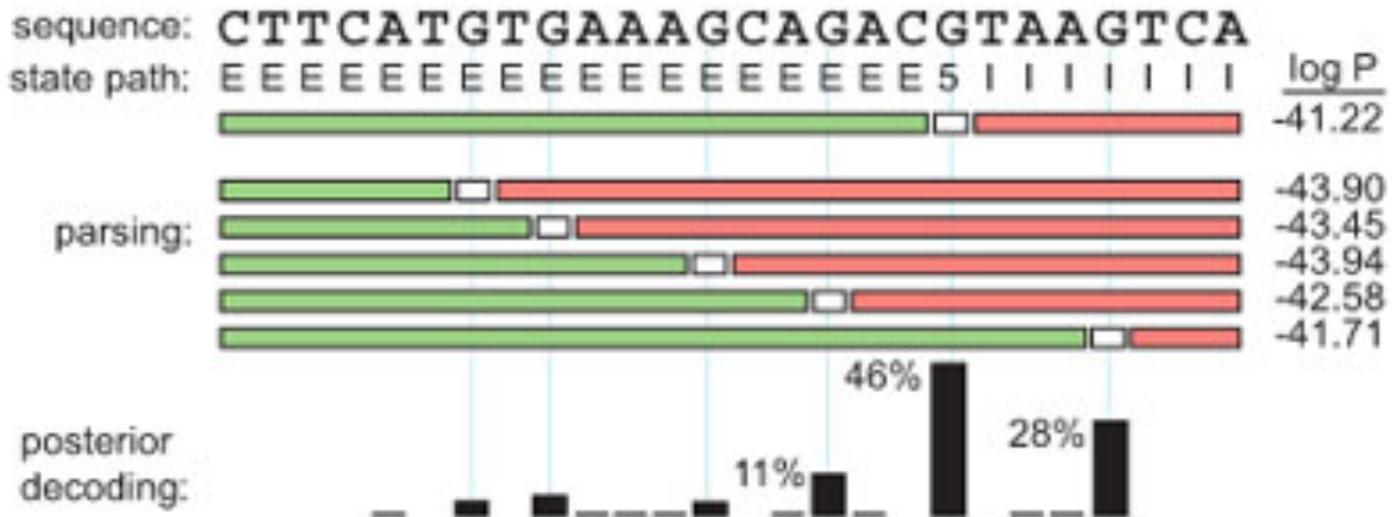
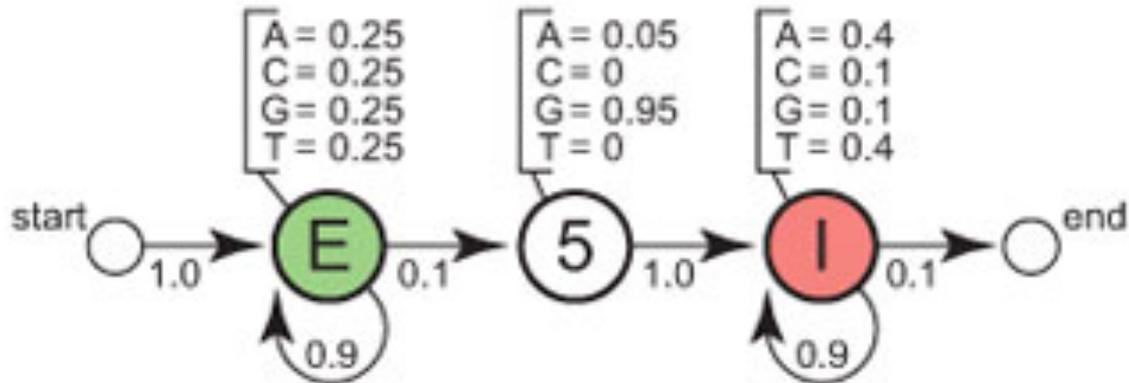


M_k Match states

I_k Insert states

D_k Delete states

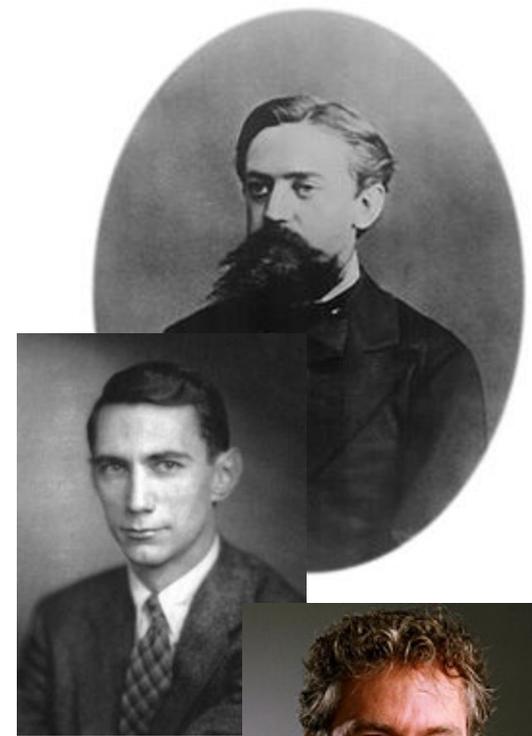
Gene Finding



WHAT IS AN HMM?

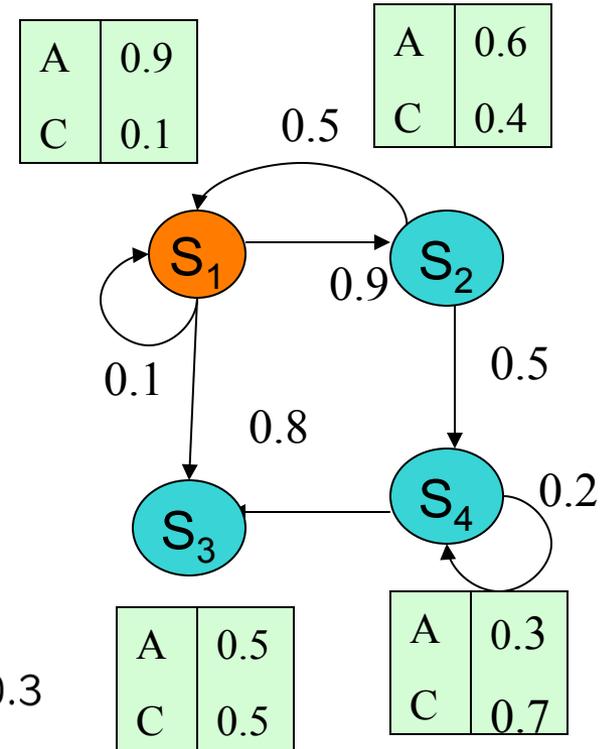
HMMs: History

- Markov chains: Andrey Markov (1906)
 - Random walks and Brownian motion
- Used in Shannon's work on information theory (1948)
- Baum-Welsh learning algorithm: late 60's, early 70's.
 - Used mainly for speech in 60s-70s.
- Late 80's and 90's: David Haussler (major player in learning theory in 80's) began to use HMMs for modeling biological sequences
- Mid-late 1990's: Dayne Freitag/Andrew McCallum
 - Freitag thesis with Tom Mitchell on IE from Web using logic programs, grammar induction, etc.
 - McCallum: multinomial Naïve Bayes for text
 - With McCallum, IE using HMMs on CORA
- ...



What is an HMM?

- Generative process:
 - Choose a *start state* S_1 using $Pr(S_1)$
 - For $i=1 \dots n$:
 - Emit a symbol x_i using $Pr(x|S_i)$
 - Transition from S_i to S_j using $Pr(S_j|S_i)$

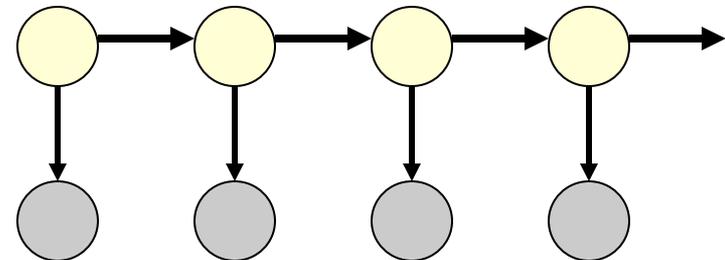


$$Pr(AACA) = \sum_{ijkl} Pr(AACA, S_i S_j S_k S_l)$$

$$Pr(AACA, S_i S_j S_k S_l) = Pr(S_i) Pr(A|S_i) Pr(S_j|S_i) \dots Pr(A|S_l)$$

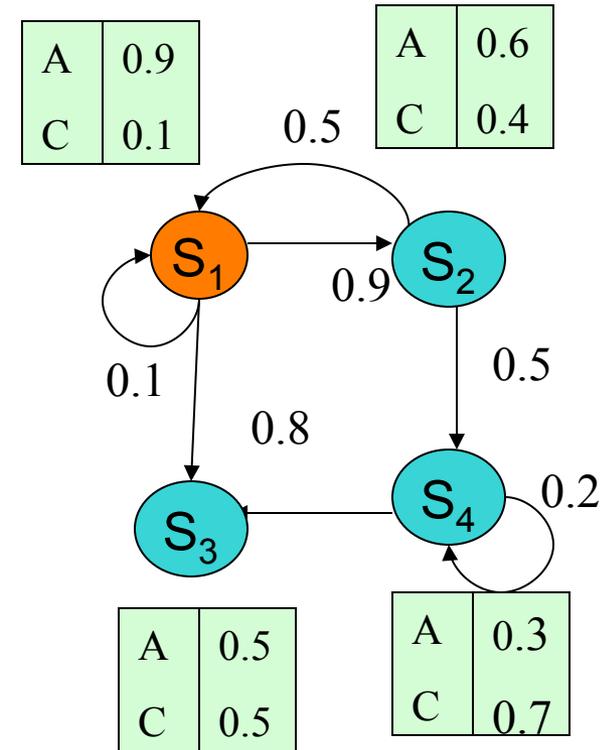
$$Pr(AACA, S_1 S_2 S_4 S_4) = 1 \times 0.9 \times 0.9 \times 0.6 \times 0.5 \times 0.7 \times 0.2 \times 0.3$$

- Usually the token sequence $x_1 x_2 x_3 \dots$ is observed and the state sequence $S_1 S_2 S_3 \dots$ is not (“hidden”)
- An HMM is a special case of a Bayes net



What is an HMM?

- Generative process:
 - Choose a *start state* S_1 using $Pr(S_1)$
 - For $i=1 \dots n$:
 - *Emit a symbol* x_i using $Pr(x|S_i)$
 - *Transition* from S_i to S_j using $Pr(S_j|S_i)$

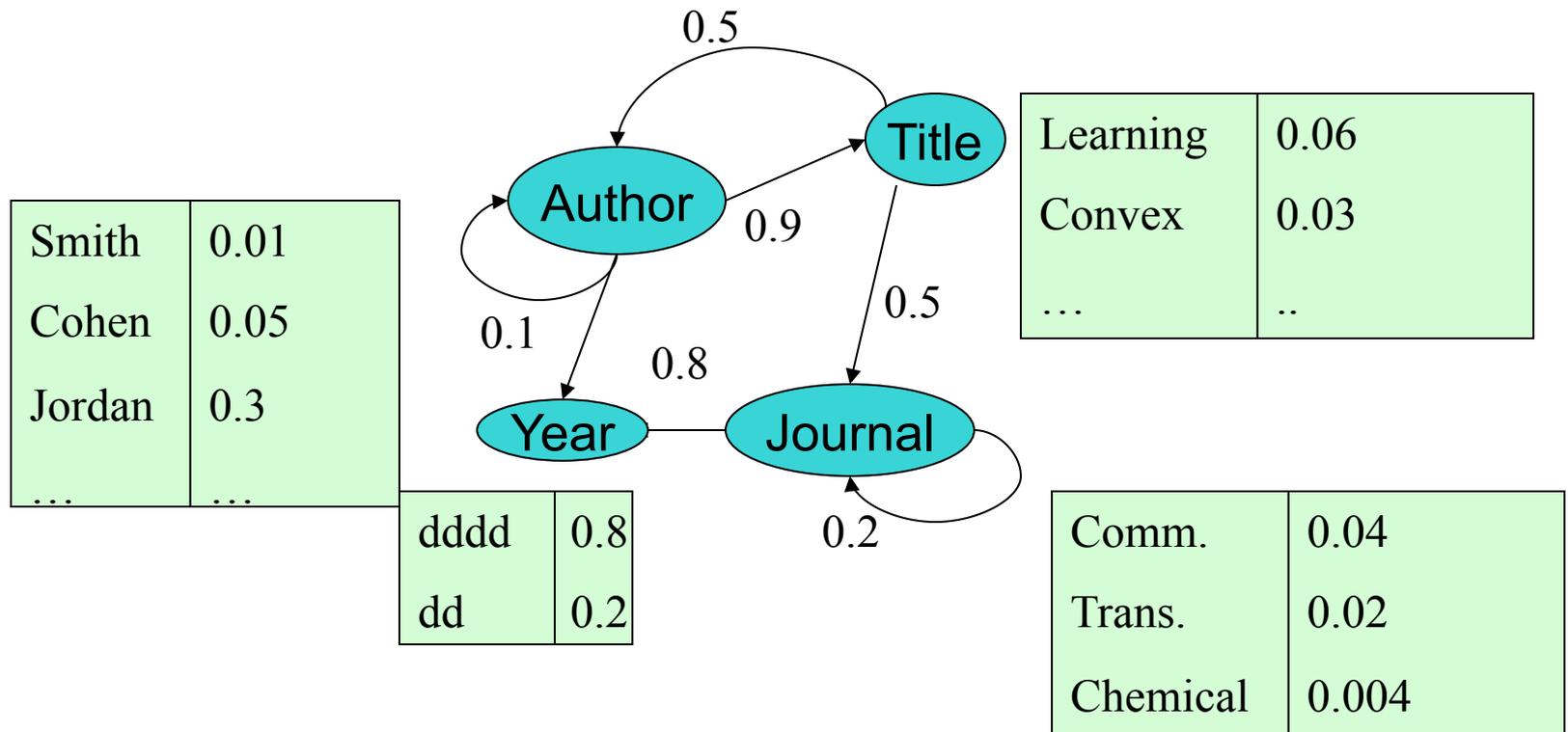


- Some key operations:
 - Given sequence $x_1x_2x_3 \dots$ find the *most probable* hidden state sequence $S_1S_2S_3 \dots$
 - We can do this efficiently! **Viterbi**
 - Given sequence $x_1x_2x_3 \dots$ find $Pr(S_j=k|X_1 \dots X_i \dots = x_1 \dots)$
 - We can do this efficiently! **Forward-Backward**

HMMS FOR NER

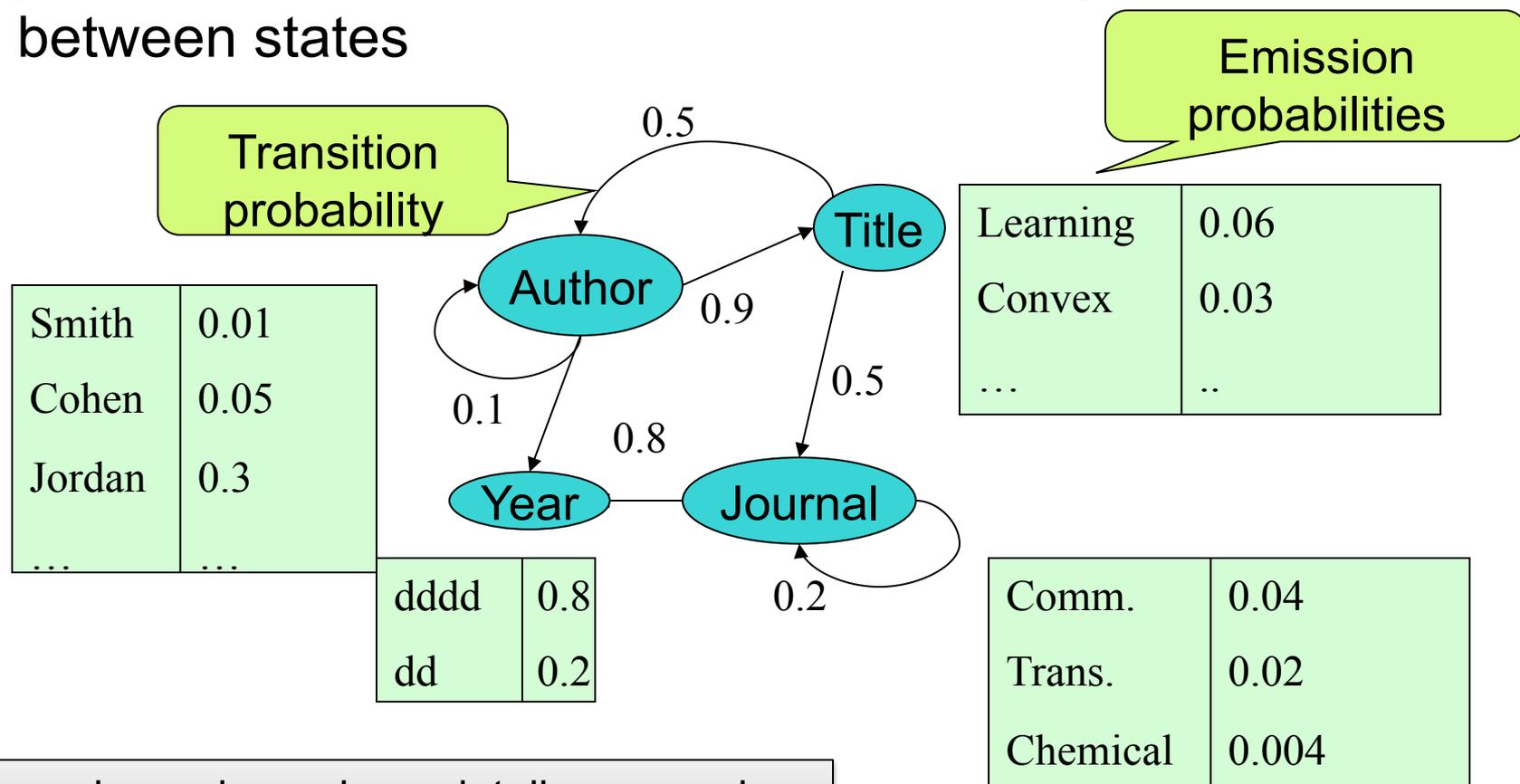
NER with Hidden Markov Models: Learning

- We usually are **given the structure** of the HMM: the vocabulary of states and symbols



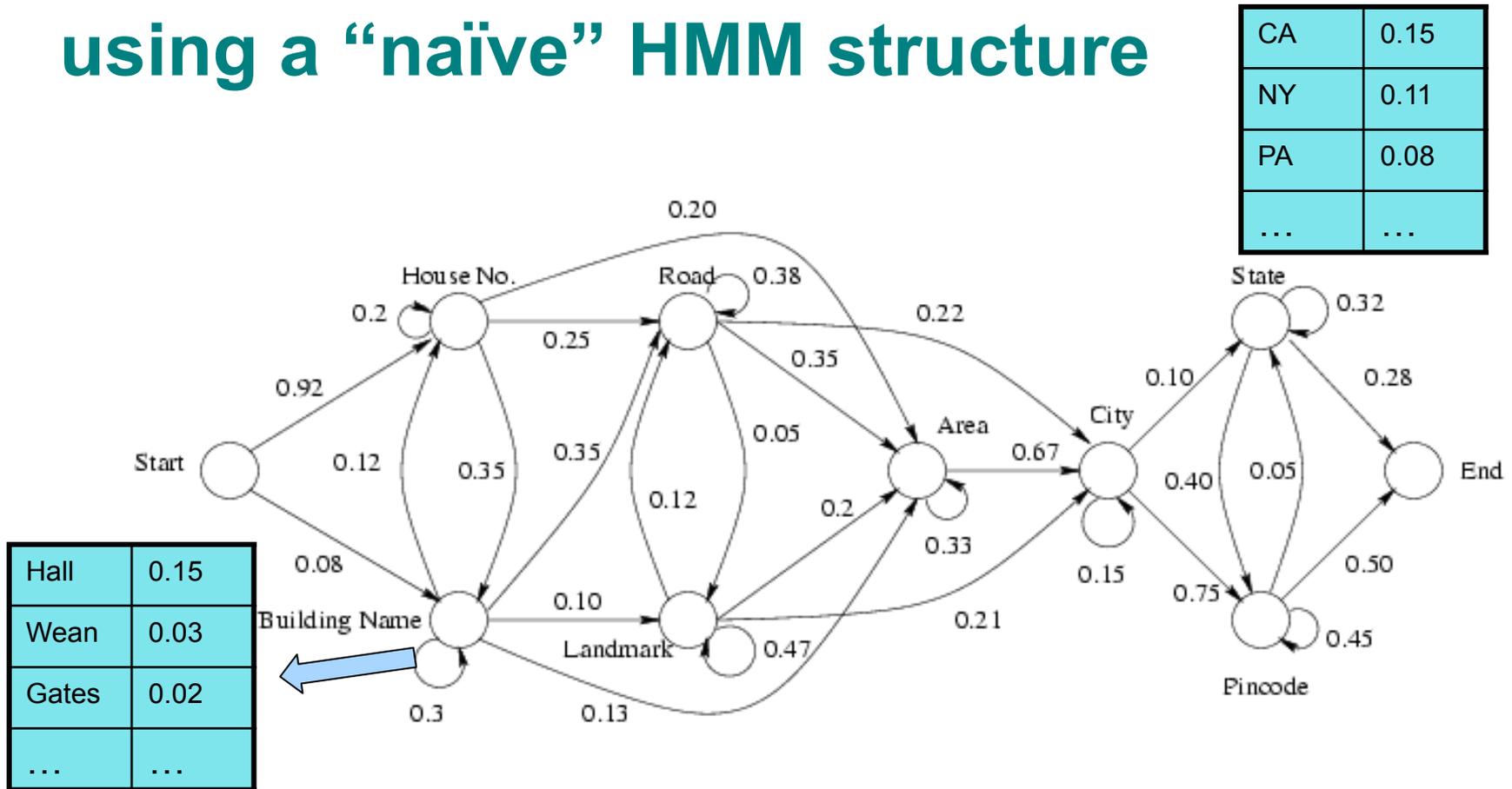
NER with Hidden Markov Models: Learning

- We **learn** the tables of numbers: *emission probabilities* for each state and *transition probabilities* between states



How we learn depends on details concerning the training data and the HMM structure.

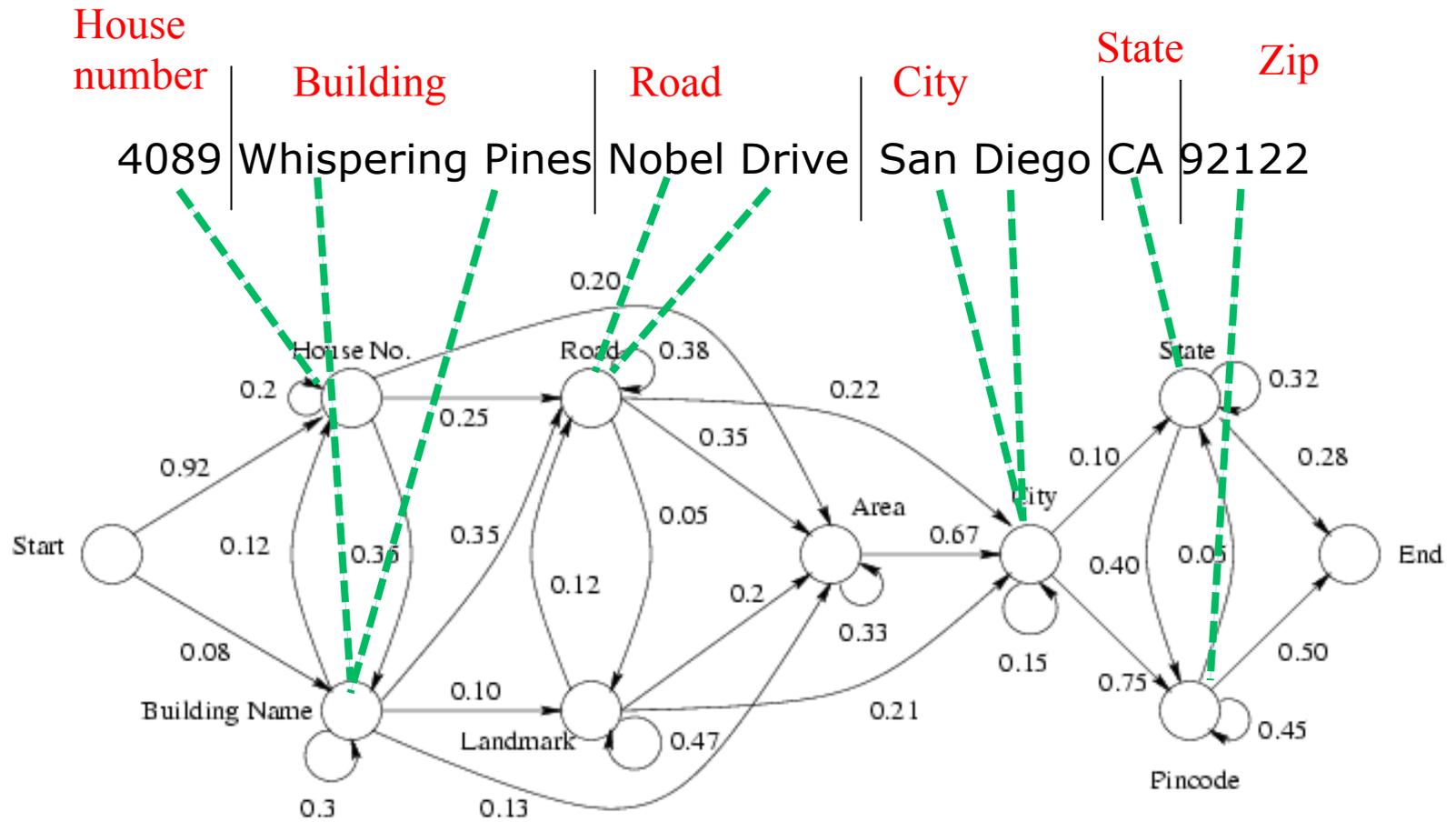
An HMM for Addresses using a “naïve” HMM structure



- “Naïve” HMM Structure: One state per entity type, and all transitions are possible

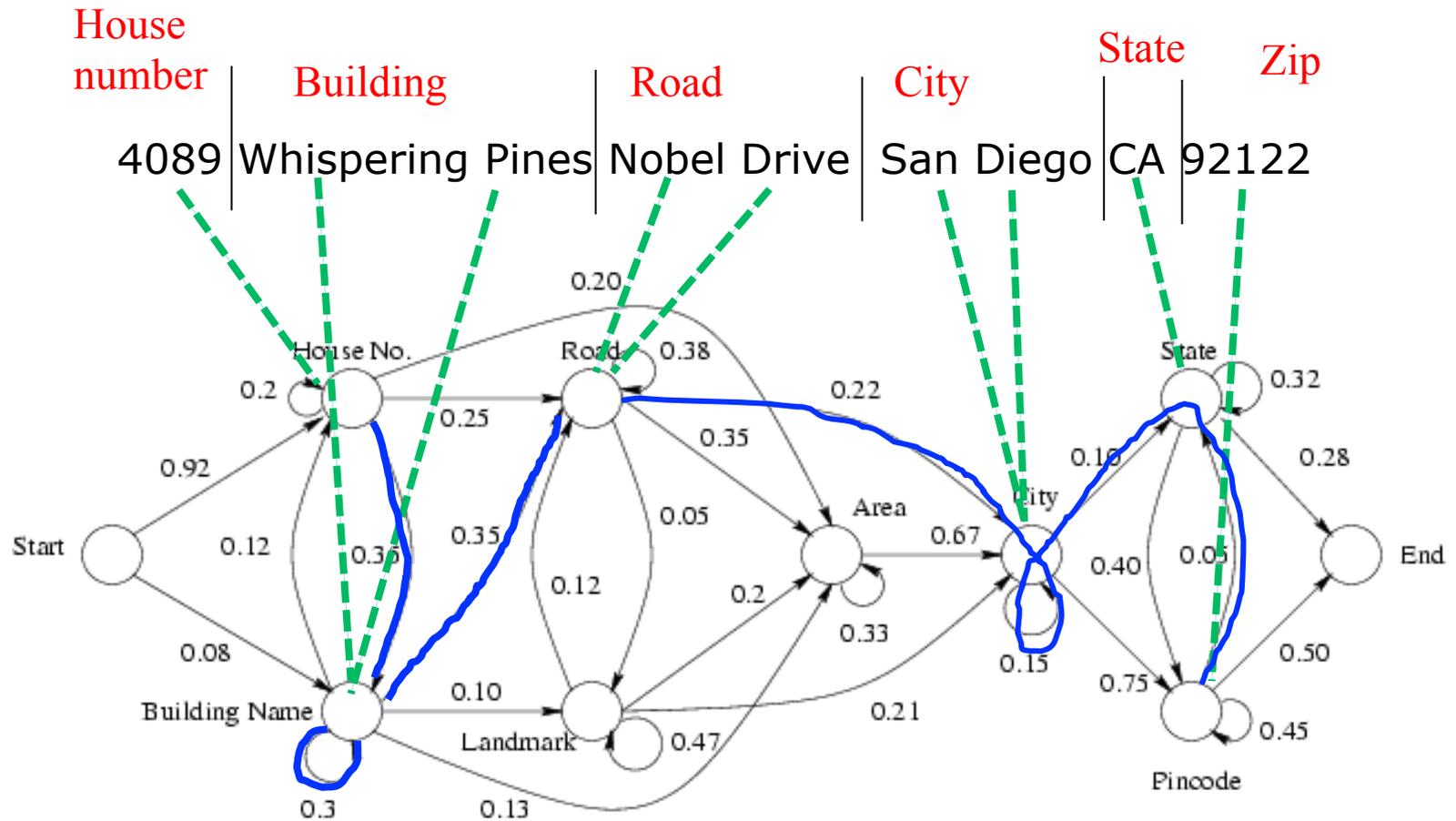
[Pilfered from Sunita Sarawagi, IIT/Bombay]





A key point: with labeled data, we know exactly **which state** emitted **which token**.

This makes it easy to learn the emission probability tables



And: with labeled data, we know exactly **which state transitions** happened.

This makes it easy to learn the transition tables

Breaking it down:

Learning parameters for the “naïve” HMM

- Training data defines unique path through HMM!

- Transition probabilities

- Probability of transitioning from state i to state j =

$$\frac{\text{number of transitions from } i \text{ to } j}{\text{total transitions from state } i}$$

with
smoothing,
of course

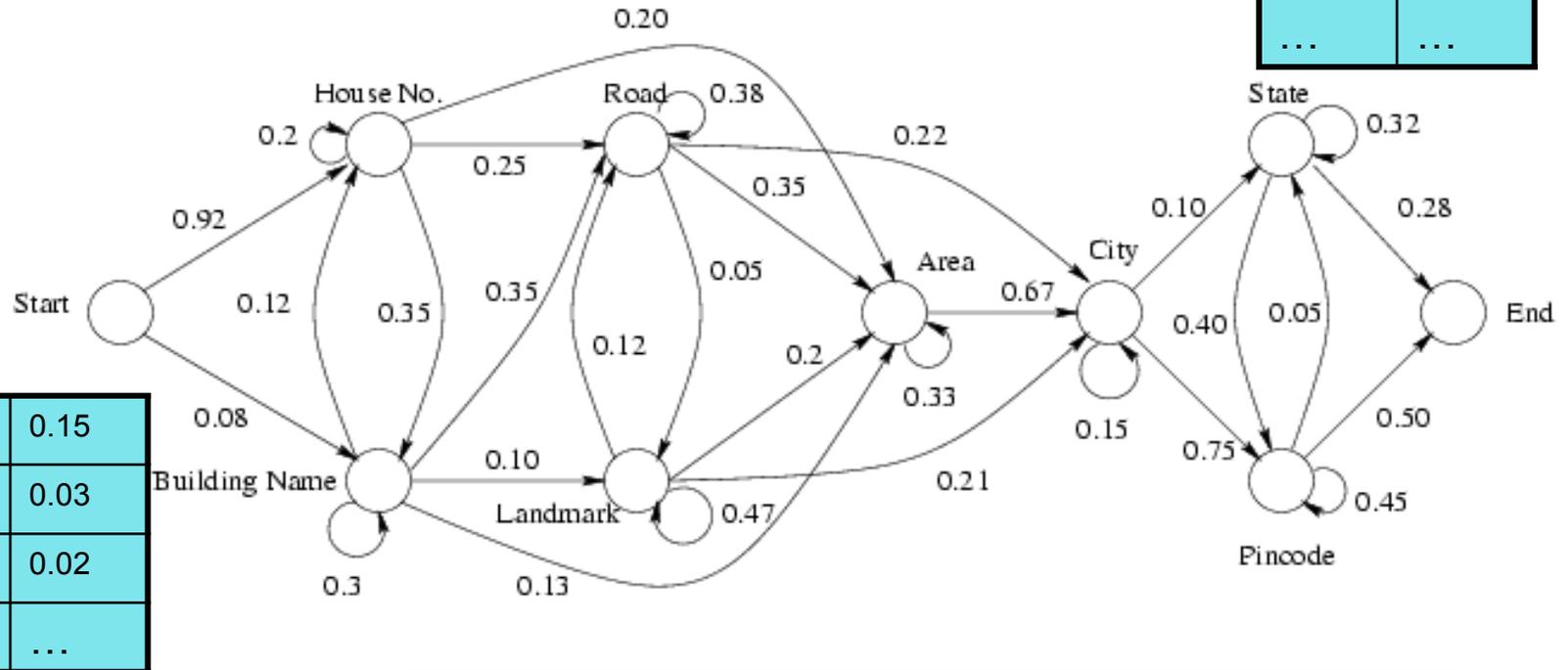
- Emission probabilities

- Probability of emitting symbol k from state i =

$$\frac{\text{number of times } k \text{ generated from } i}{\text{number of transitions from } i}$$

Result of learning: states, transitions, and emissions

CA	0.15
NY	0.11
PA	0.08
...	...

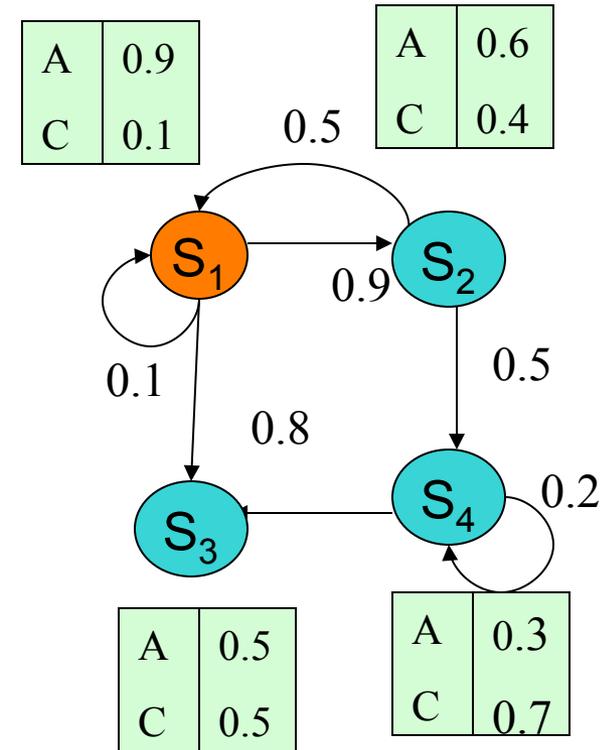


How do we use this to classify a test sequence?

House number | Building | Road | City | State | Zip
 4089 | Whispering Pines | Nobel Drive | San Diego | CA | 92122

What is an HMM?

- Generative process:
 - Choose a *start state* S_1 using $Pr(S_1)$
 - For $i=1 \dots n$:
 - Emit a symbol x_i using $Pr(x|S_i)$
 - Transition from S_i to S_j using $Pr(S_j|S_i)$
- Some key operations:
 - Given sequence $x_1x_2x_3 \dots$ find the *most probable* hidden state sequence $S_1S_2S_3 \dots$
 - We can do this efficiently! **Viterbi**



- Given sequence $x_1x_2x_3 \dots$ find $Pr(S_j=k|X_1 \dots X_i \dots = x_1 \dots)$
 - We can do this efficiently! **Forward-Backward**

VITERBI FOR HMMS

Viterbi in pictures

s1	s2	s3	s4	s5	s6
4089	Nobel	Drive	San	Diego	92122

Four states: HouseNum, Road, City, Zip

The slow way: test every possible hidden state sequence $\langle s_1 s_2 \dots s_6 \rangle$ and see which makes the text most probable (6^4 sequences).

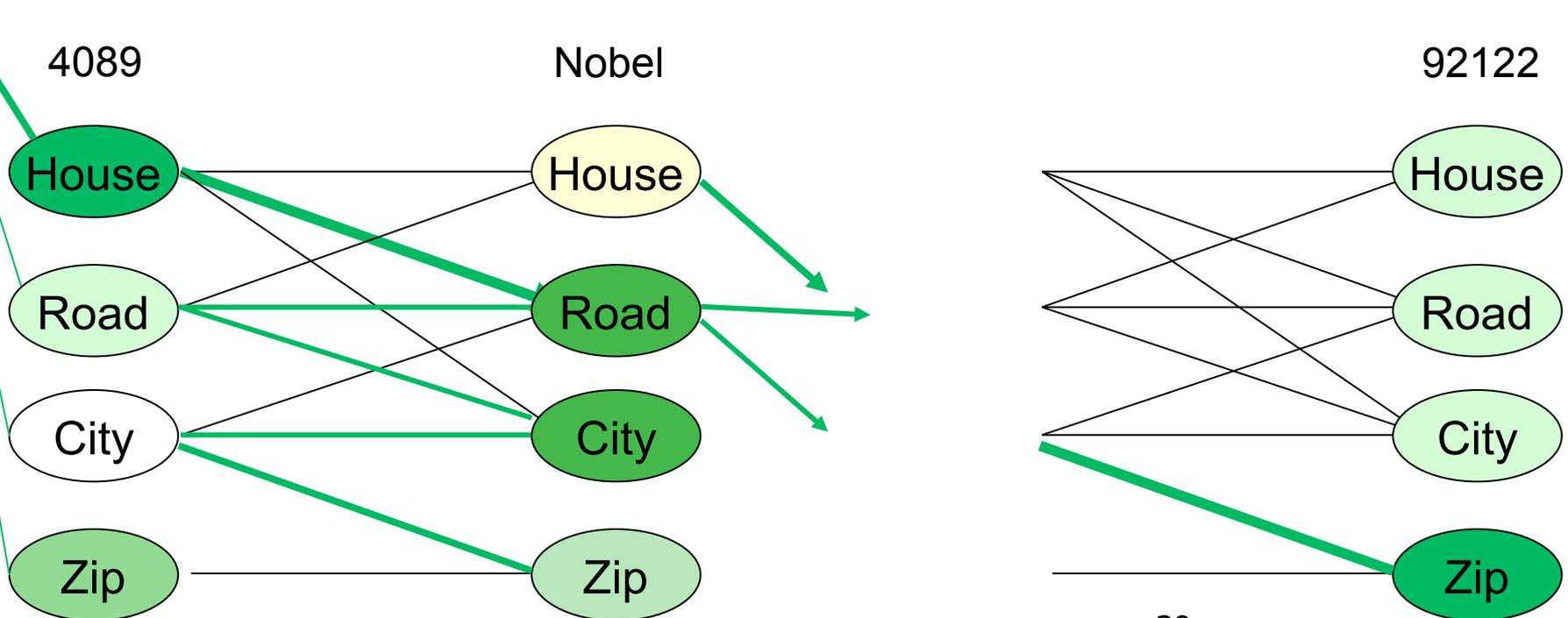
$$\Pr(4089 \text{ Nobel Drive San Diego } 92122 \mid s_1 s_2 \dots s_6) = \\ \Pr(s_1) \Pr(4089 \mid s_1) \Pr(s_2 \mid s_1) \Pr(\text{Nobel} \mid s_2) \dots \Pr(s_6 \mid s_5)$$

The fast way: dynamic programming: reduces time from $O(|S|^{|x|})$ to $O(|x||S|^2)$

Viterbi in pictures

4089 Nobel Drive San Diego 92122

Circle color indicates $\Pr(x|s)$, line width indicates $\Pr(s'|s)$



Viterbi algorithm

- Let V be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns.
- Let ptr be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns..
 - $V(k,j)$ will be: max over all $s_1 \dots s_{j-1}: s_{j-1}=k$ of $\text{Prob}(x_1 \dots x_j | s_1 \dots s_{j-1})$

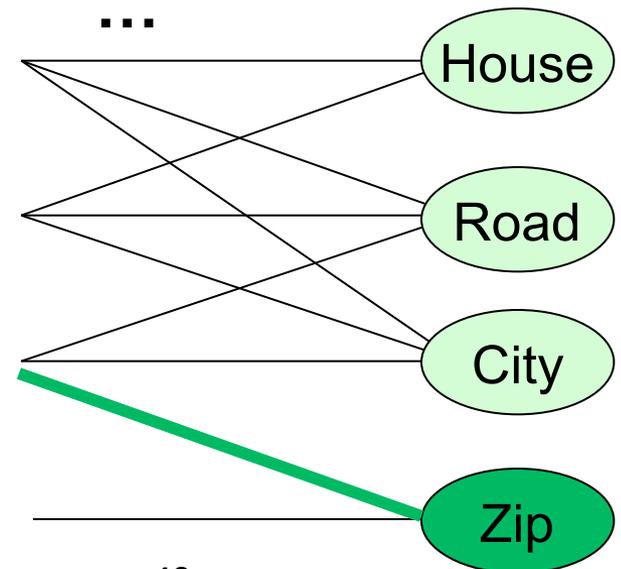
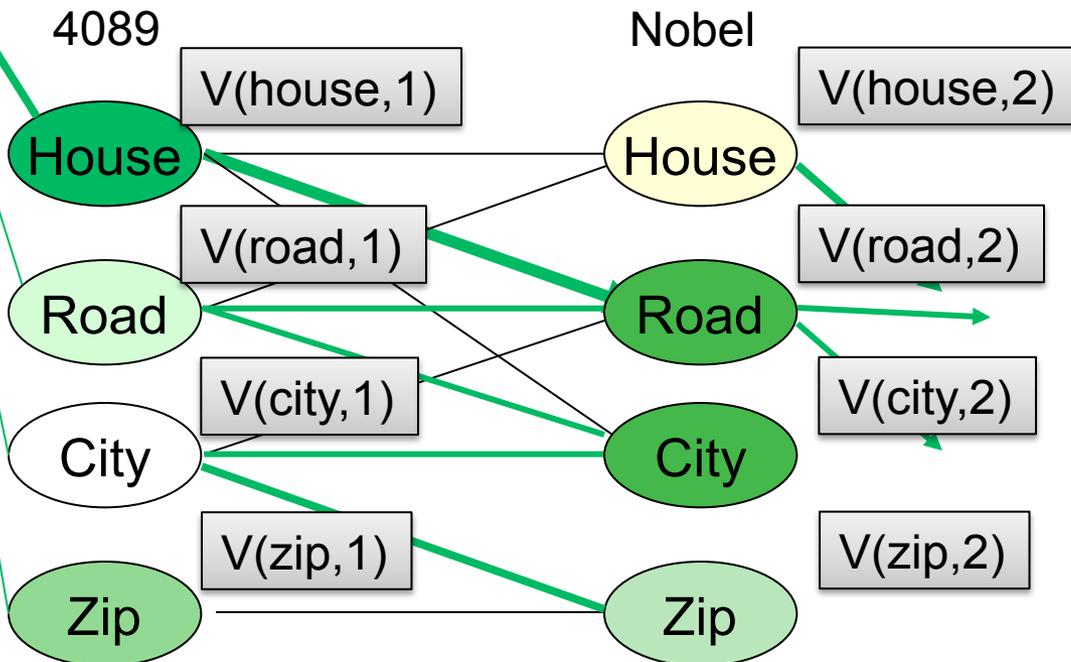
• For all k : $V(k,1) = \text{Pr}(S_1=k) * \text{Pr}(x_1|S=k)$

$\text{Pr}(\text{start}) * \text{Pr}(\text{first emission})$

• For $j=1, \dots, |\mathbf{x}|$

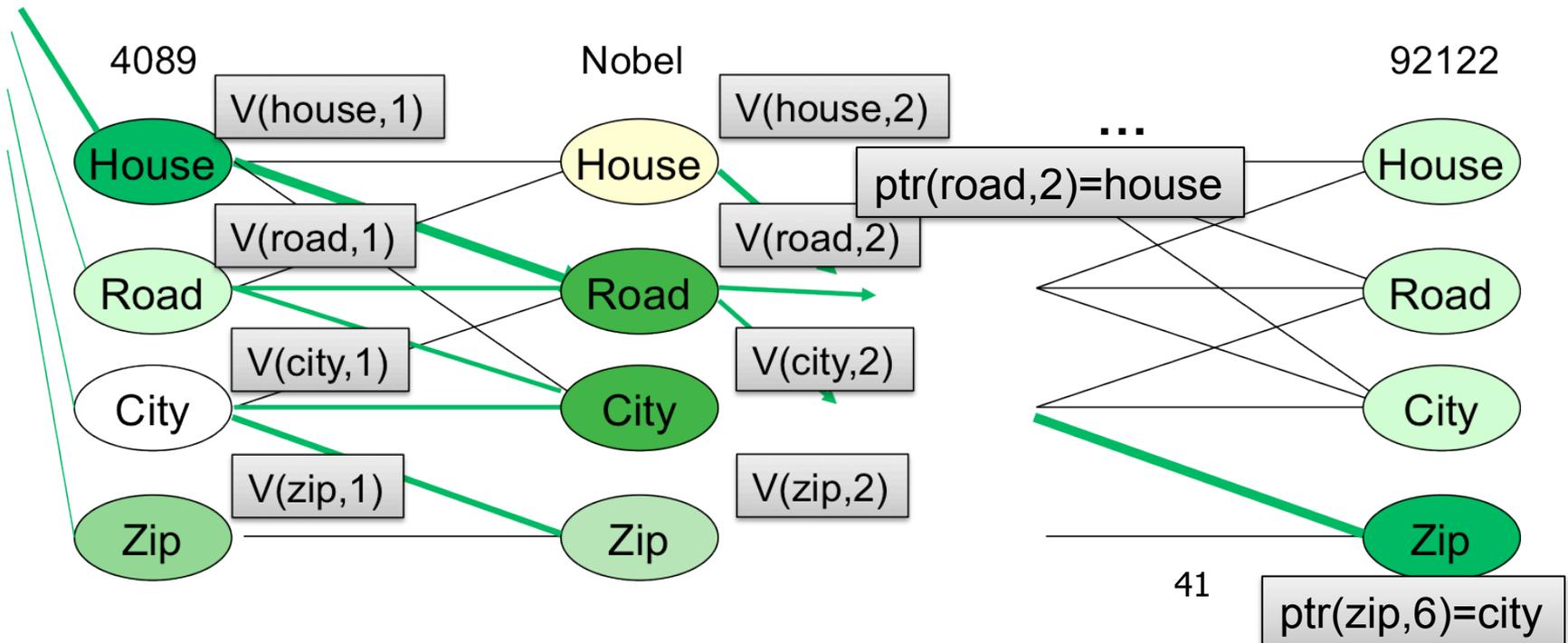
• $V(k,j+1) = \text{Pr}(x_j|S=k) * \max_{k'} [\text{Pr}(S=k|S'=k') * V(k',j)]$

$\text{Pr}(\text{transition}) * \text{Pr}(\text{emission})$



Viterbi algorithm

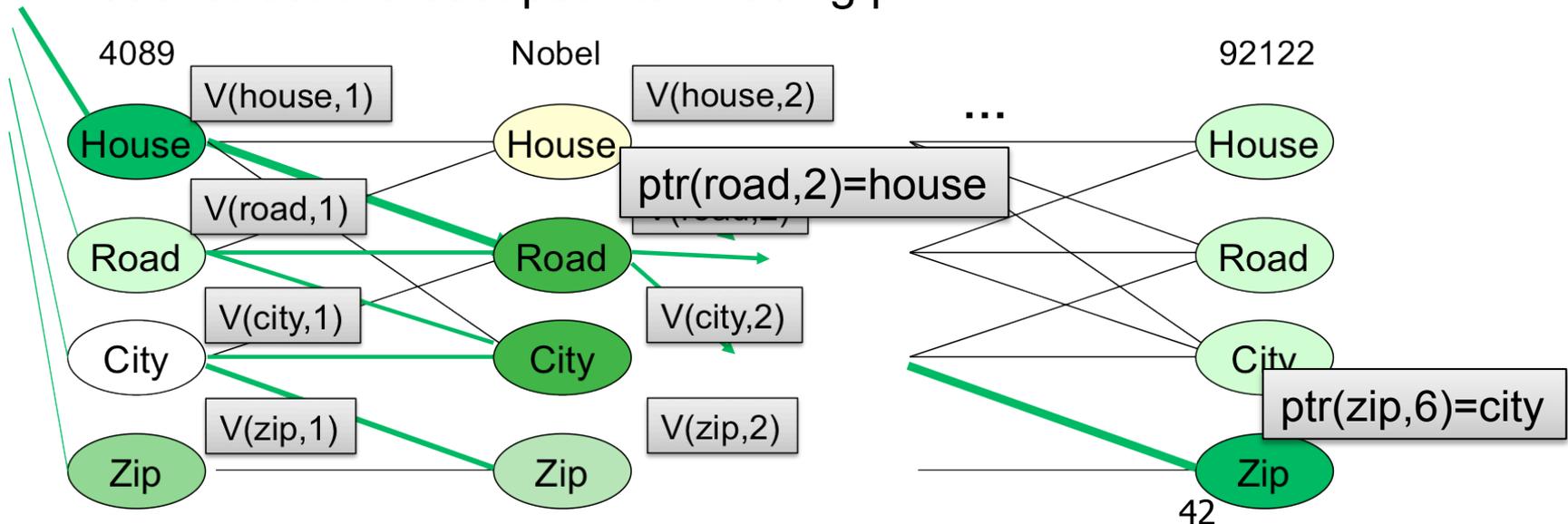
- Let V be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns.
 - Let ptr be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns..
- For all k : $V(k,1) = \Pr(S_1=k) * \Pr(x_1|S=k)$
 - For $j=1, \dots, |\mathbf{x}|$
 - $V(k,j+1) = \Pr(x_j|S=k) * \max_{k'} \Pr(S'=k|S=k') * V(k',j)$
 - $\text{ptr}(k,j+1) = \text{argmax}_{k'} \Pr(x_j|S=k) * \Pr(S=k|S'=k') * V(k',j)$



Viterbi algorithm

- Let V be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns.
- Let ptr be a matrix with $|S|$ rows and $|\mathbf{x}|$ columns..
- For all k : $V(k,1) = \Pr(S_1=k) * \Pr(x_1|S=k)$
- For $j=1, \dots, |\mathbf{x}|-1$
 - $V(k,j+1) = \Pr(x_j|S=k) * \max_{k'} \Pr(S'=k|S=k') * V(k',j)$
 - $\text{ptr}(k,j+1) = \text{argmax}_{k'} \Pr(x_j|S=k) * \Pr(S=k|S'=k') * V(k',j)$
- Let $k^* = \text{argmax}_k V(k,|\mathbf{x}|)$ -- *the best final path*
- Reconstruct the best path to k^* using ptr

Implement this in log space with addition instead of multiplication



Breaking it down:

NER using the “naïve” HMM

- Define the HMM structure:
 - one state per entity type
- Training data defines unique path through HMM for each labeled example
 - Use this to estimate transition and emission probabilities
- At test time for a sequence \mathbf{x}
 - Use Viterbi to find sequence of states \mathbf{s} that maximizes $\Pr(\mathbf{x}|\mathbf{s})$
 - Use \mathbf{s} to derive labels for the sequence \mathbf{x}

What forward-backward computes

Parsing addresses

Like probabilistic inference:
 $\Pr(X|E)$

House number	Building	Road	City	State	Zip
4089	Whispering	Pines	Nobel Drive	San Diego	CA 92122

What is the best prediction for this token?

for this token?

Parsing citations

Author

P.P.Wangikar, T.P. Graycar, D.A. Estell, D.S. Clark, J.S. Dordick (1993)
Protein and Solvent Engineering of Subtilising BPN' in Nearly
Anhydrous Organic Media J.Amer. Chem. Soc. 115, 12231-12237.

Title

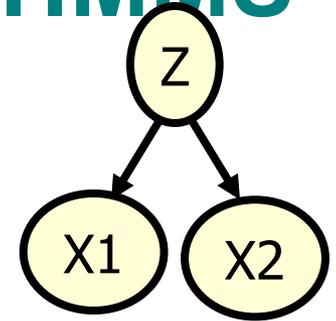
Journal

Volume

THE FORWARD-BACKWARD ALGORITHM FOR HMMS

F-B could also be used to learn HMMs with *hidden variables*

Hidden variables: what if some of your data is not completely observed?



Method (Expectation-Maximization, EM):

1. Estimate parameters somehow or other.
2. Predict unknown values from your estimated parameters (**Expectation step**)
3. Add pseudo-data corresponding to these predictions, *weighting each example by confidence in its correctness.*
4. Re-estimate parameters using the extended dataset (real + pseudo-data).
 - You find the MLE or MAP values of the parameters. (**Maximization step**)
5. Repeat starting at step 2....

Z	X1	X2
ugrad	<20	facebook
ugrad	20s	facebook
grad	20s	thesis
grad	20s	facebook
prof	30+	grants
?	<20	facebook
?	30s	thesis

Possible application of F-B: partially labeled data

McCallum & Culotta, AAI 2005

1. System proposes a segmentation:

House number	Road	City	State	Zip
4089	Whispering Pines Nobel Drive	San Diego	CA	92122

2. User corrects errors in segmentation:

House number	Building	Road	Didn't really look at this
4089	Whispering Pines	Nobel Drive	San Diego CA 92122

3. Depending on how careful the user is – you might want to use only some of the labeled data

For EM: what is the prediction for this token?

More applications of F-B

User, or some other source, gives a partial segmentation:

 O B I O
Mid-late 1990's : Dayne Freitag and A.
McCallum.

 O B I O
Freitag thesis with Tom Mitchell on
IE from Web using logic programs,
grammar induction, etc.

McCallum: multinomial Naïve
Bayes for text

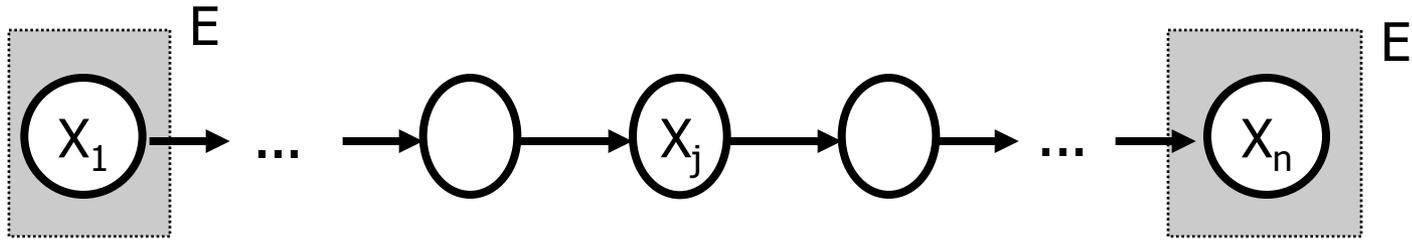
With McCallum, IE using HMMs on
CORA

...

Forward backward warmup 1

BP on chains

A special case: linear chain networks



$$P(X_1, \dots, X_n) = P(X_n | X_{n-1})P(X_{n-1} | X_{n-2}) \dots P(X_2 | X_1)P(X_1)$$

$$P(X_j | x_1, x_n) = \frac{P(x_n | X_j, x_1)P(X_j | x_1)}{P(x_n | x_1)}$$

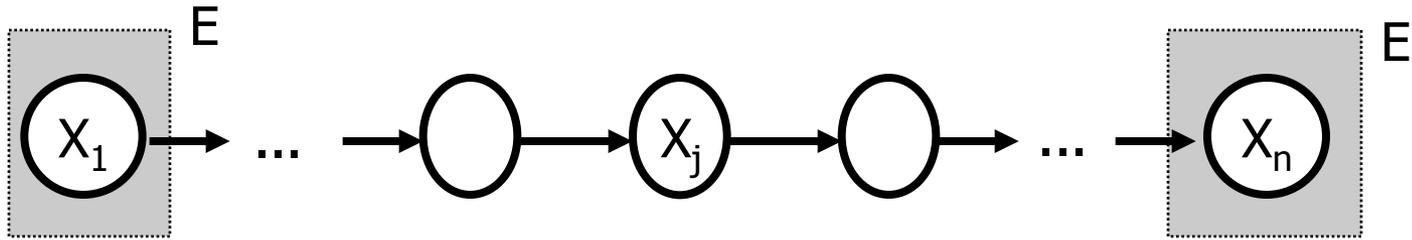
d-separation

$$= c \cdot \underbrace{P(x_n | X_j)}_{\text{backward}} \underbrace{P(X_j | x_1)}_{\text{forward}}$$

“backward”
(evidential)

“forward”
(causal)

A special case: linear chain networks



$$P(X_1, \dots, X_n) = P(X_n | X_{n-1})P(X_{n-1} | X_{n-2}) \dots P(X_2 | X_1)P(X_1)$$

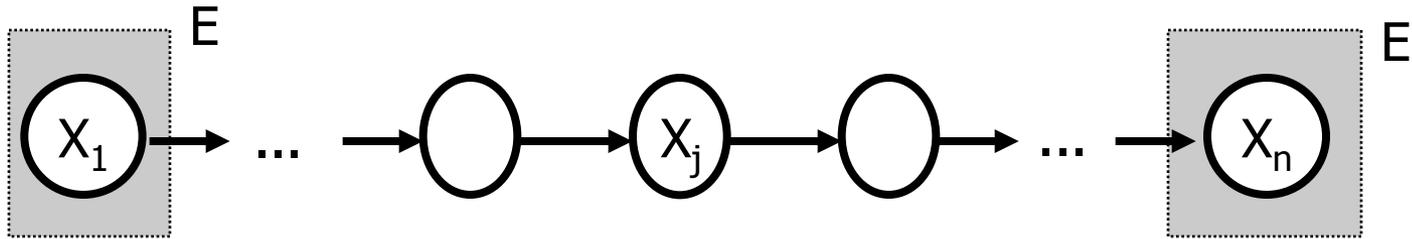
$$P(X_j | x_1, x_n) = c \cdot P(x_n | X_j)P(X_j | x_1)$$

$$\text{Fwd: } P(X_j = x | x_1) = \sum_{x'} \underbrace{P(X_{j-1} = x' | x_1)}_{\text{Recursion! (fwd)}} \underbrace{P(X_j = x | X_{j-1} = x')}_{\text{CPT entry}}$$

Recursion!
(fwd)

CPT entry

A special case: linear chain networks



$$P(X_1, \dots, X_n) = P(X_n | X_{n-1})P(X_{n-1} | X_{n-2}) \dots P(X_2 | X_1)P(X_1)$$

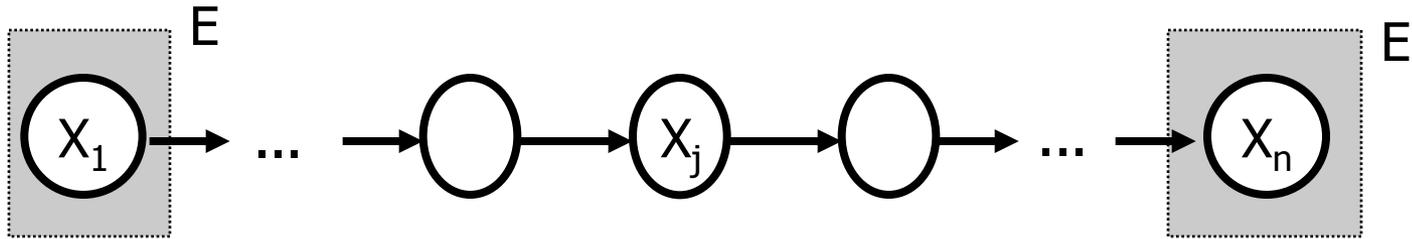
$$P(X_j | x_1, x_n) = \alpha \cdot P(x_n | X_j)P(X_j | x_1)$$

Back:

$$P(x_n | X_j) = \sum_{x'} P(x_n, X_{j+1} = x' | X_j)$$

$$= \sum_{x'} \underbrace{P(x_n | X_{j+1} = x', \cancel{X_j})}_{\text{Recursion backward}} \underbrace{P(X_{j+1} = x' | X_j)}_{\text{CPT}} \quad \text{Chain rule}$$

A special case: linear chain networks



$$P(X_1, \dots, X_n) = P(X_n | X_{n-1})P(X_{n-1} | X_{n-2}) \dots P(X_2 | X_1)P(X_1)$$

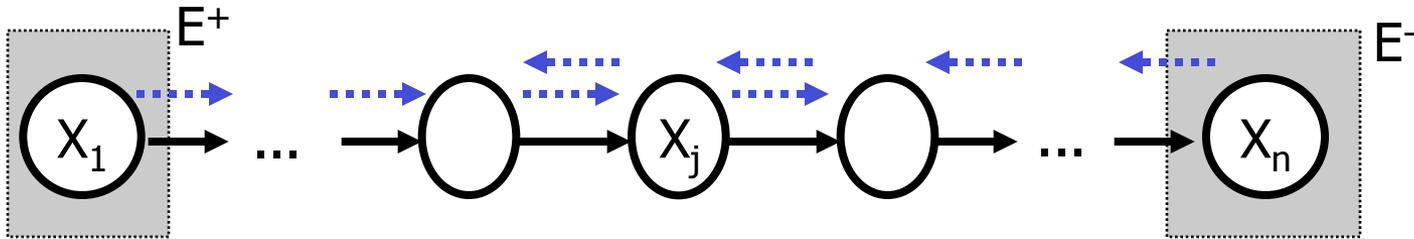
$$P(X_j | x_1, x_n) = \underbrace{\alpha \cdot P(x_n | X_j)}_{\text{“backward”}} \underbrace{P(X_j | x_1)}_{\text{“forward”}}$$

Instead of recursion:

- iteratively compute $P(X_j | x_1)$ from $P(X_{j-1} | x_1)$ – the forward probabilities
- iteratively compute $P(x_n | X_j)$ from $P(x_n | X_{j+1})$ – the backward probabilities
- can view the forward computations as passing a “message” forward
 - and vice versa

Linear-chain message passing

$$P(X_j|E) = P(X|E^+)P(X|E^-) \dots \text{true by d-separation}$$



Pass forward: $P(X_j|E^+)$...computed from $P(X_{j-1}|E^+)$ and CPT for X_j

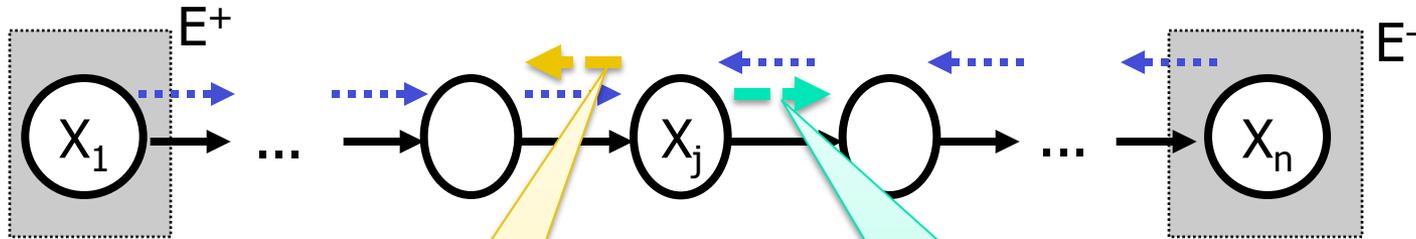
$$P(X_j = x | x_1) = \sum_{x'} P(X_{j-1} = x' | x_1) P(X_j = x | X_{j-1} = x')$$

Pass backward: $P(X_j|E^-)$...computed from $P(X_{j+1}|E^-)$ and CPT for X_{j+1}

$$P(x_n | X_j = x) = \sum_{x'} P(x_n | X_{j+1} = x') P(X_{j+1} = x' | X_j = x)$$

Linear-chain message passing

$$P(X_j|E) = P(X|E^+)P(X|E^-) \dots \text{true by d-separation}$$



$\Pr(E^-|X_j=b_person)=0.15$
 $\Pr(E^-|X_j=i_person)=0.731$
...

$\Pr(X_j=b_person|E^+)=0.2$
 $\Pr(X_j=i_person|E^+)=0.36$
...

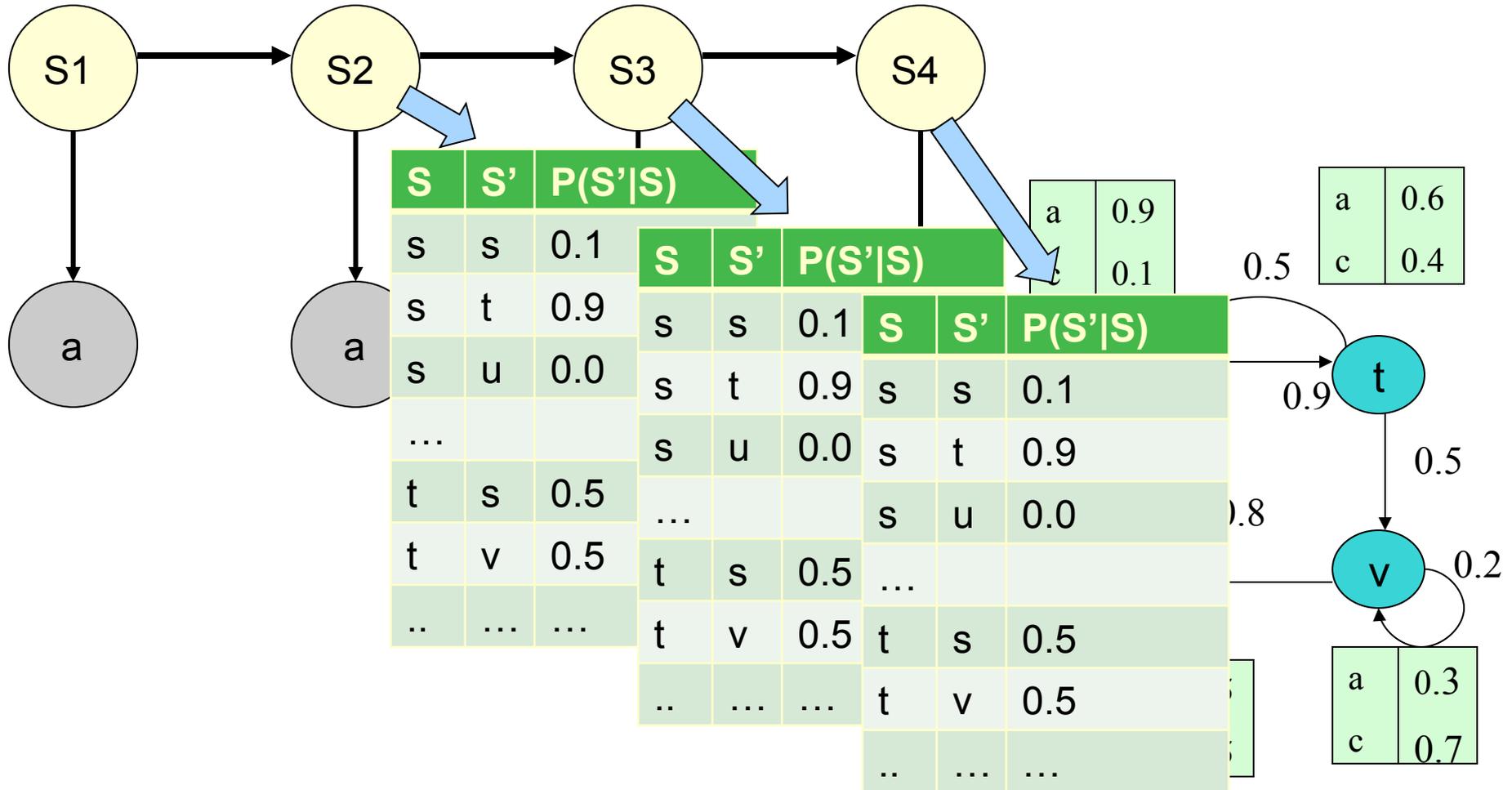
Forward backward warmup 2

simple “dynamic” Bayes nets

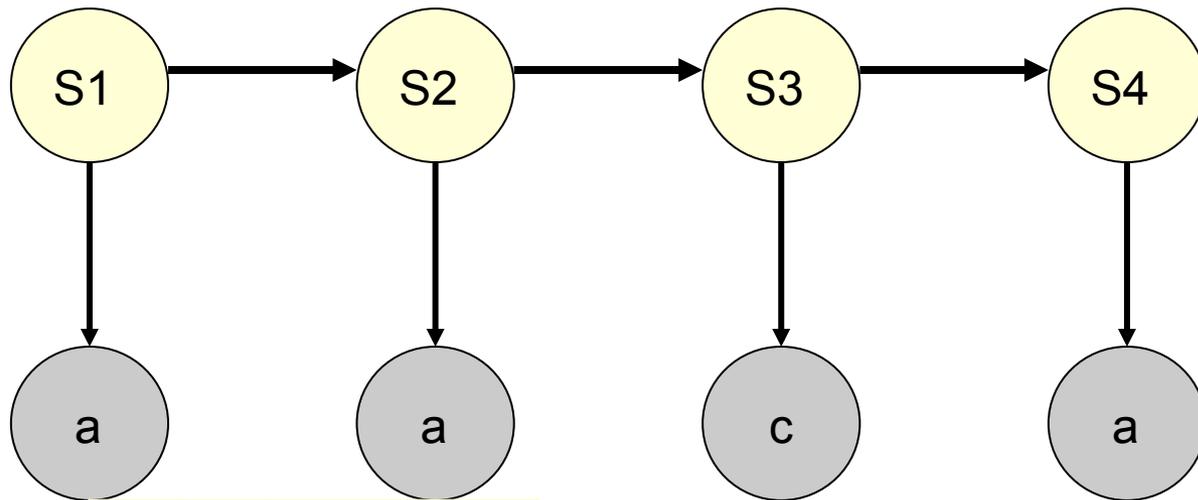
An HMM-like Bayes Net

S	P(S)
s	1.0
t	0.0
u	0.0
v	0.0

“Tied” parameters

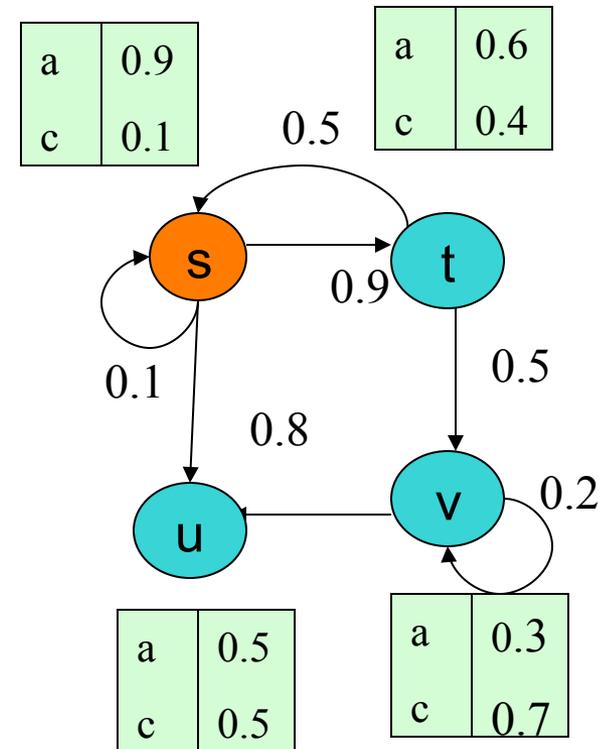


An HMM-like Bayes Net



S	X	P(X S)	S	X	P(X S)
s	a	0.9	s	a	0.9
s	c	0.1	s	c	0.1
t	a	0.6	t	a	0.6
t	c		t	c	0.4
..

“Tied” parameters



...

Forward-backward for a small example

	S1	S2	S3	S4	S5	S6
x	4089	Nobel	Drive	San	Diego	92122

Four states: HouseNum, Road, City, Zip

What is $\Pr(S3=\text{road}|\text{evidence})$?

The slow way: use the joint.

There are $4^6 = 4096$ entries here

Is there a faster way? Use BP – the special case for this problem for HMMs is called forward-backward

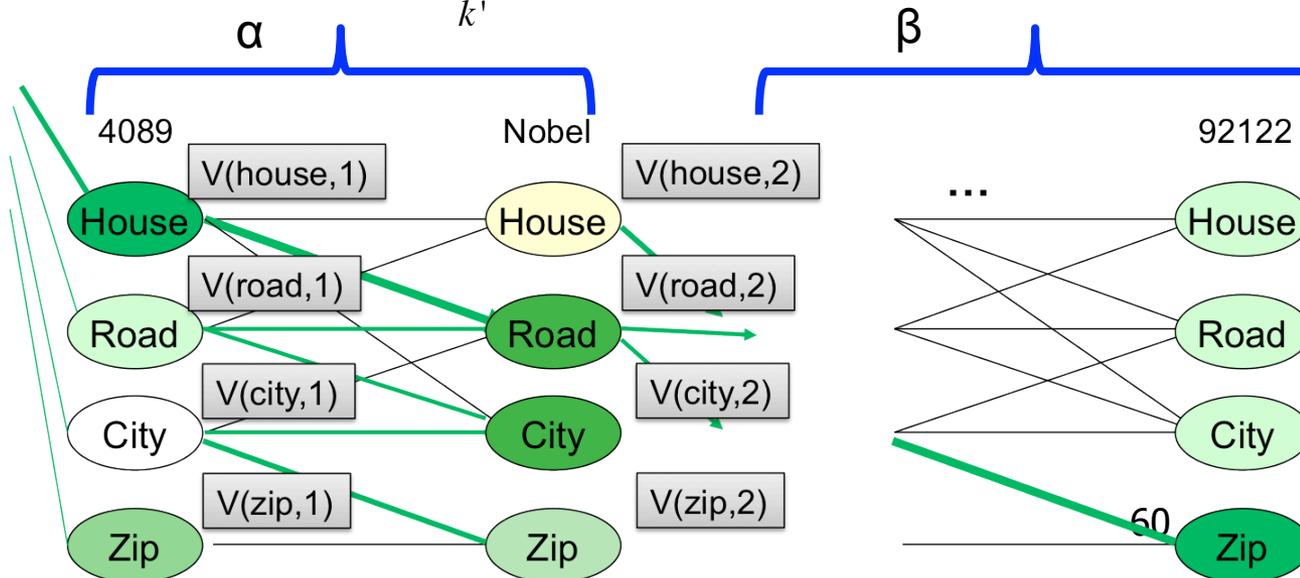
Forward Backward

- Let α and β be matrices with $|S|$ rows and $|\mathbf{x}|$ columns.
 - α is “forward probability”, β is “backward probability”
- $\alpha(k, 1) = V(k, 1) = \Pr(S_1=k) * \Pr(x_1|S=k)$
- $\beta(k, |\mathbf{x}|+1) = 1$
- For $j=1$ to $|\mathbf{x}|-1$:

$$\alpha(k, j+1) = \Pr(x_{j+1} | S_{j+1} = k) \sum_{k'} \Pr(S = k | S' = k') \alpha(k', j)$$

- For $j = |\mathbf{x}|$ down to 2:

$$\beta(k, j-1) = \sum_{k'} \Pr(x_j | S = k') \Pr(S = k | S = k') \beta(k', j)$$



Forward Backward

- Let α and β be matrices with $|S|$ rows and $|\mathbf{x}|$ columns.
 - α is “forward probability”, β is “backward probability”

- $\alpha(k, 1) = V(k, 1) = \Pr(S_1=k) * \Pr(x_1|S=k)$

- $\beta(k, |\mathbf{x}|+1) = 1$

- For $j=1$ to $|\mathbf{x}|-1$:

$$\alpha(k, j+1) = \Pr(x_{j+1} | S_{j+1} = k) \sum_{k'} \Pr(S = k | S' = k') \alpha(k', j)$$

- For $j = |\mathbf{x}|$ down to 2:

$$\beta(k, j-1) = \sum_{k'} \Pr(x_j | S = k') \Pr(S = k | S = k') \beta(k', j)$$

- Now we can compute expectations over the hidden variables:

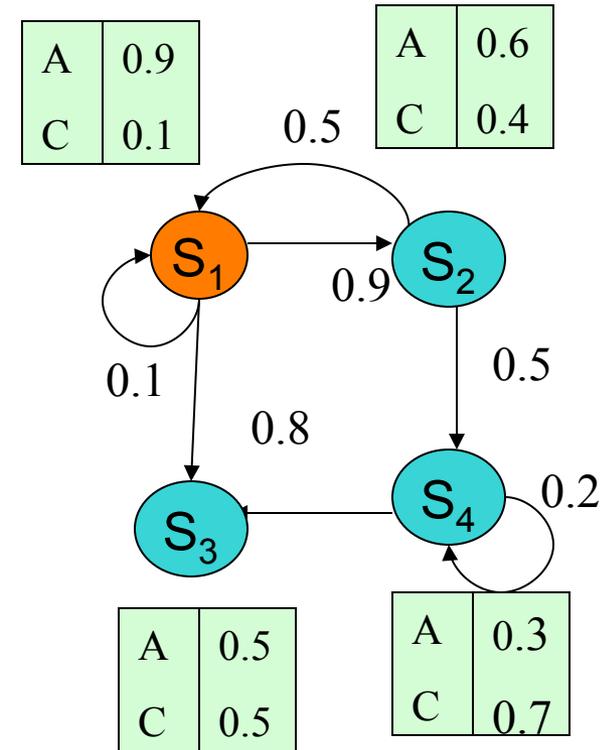
$$\Pr(S_j = k | \mathbf{x}) = \alpha(k, j) \beta(k, j)$$

- ...which lets us classify tokens, or use EM to learn (for HMMs, it's called Baum-Welch)

What is an HMM?

- Generative process:
 - Choose a *start state* S_1 using $Pr(S_1)$
 - For $i=1 \dots n$:
 - Emit a symbol x_i using $Pr(x|S_i)$
 - Transition from S_i to S_j using $Pr(S_j|S_i)$

- Some key operations:
 - Given sequence $x_1x_2x_3 \dots$ find the *most probable* hidden state sequence $S_1S_2S_3 \dots$
 - We can do this efficiently! **Viterbi**



- Given sequence $x_1x_2x_3 \dots$ find $Pr(S_j=k|X_1 \dots X_i \dots = x_1 \dots)$
 - We can do this efficiently! **Forward-Backward**

Viterbi in pictures

s1	s2	s3	s4	s5	s6
4089	Nobel Drive	San Diego	92122		

Four states: HouseNum, Road, City, Zip

Inference task: find $\operatorname{argmax}_{\mathbf{s}} \Pr(\mathbf{s}|\mathbf{x})$

$$\operatorname{argmax}_{s_1, \dots, s_6} \Pr(S_1=s_1, \dots, S_6=s_6 | X_1=4089, \dots, X_5=\text{Diego}, S_6=92122)$$

The slow way: test every possible hidden state sequence $\langle s_1 s_2 \dots s_6 \rangle$ and see which makes the text most probable (6^4 sequences).

The fast way: variant* of BP – special case for HMMs is called Viterbi. (By extension we sometimes use “Viterbi” for the analogous operation for DGMs.)

Algorithm left as an exercise for the student.

CONDITIONAL RANDOM FIELDS

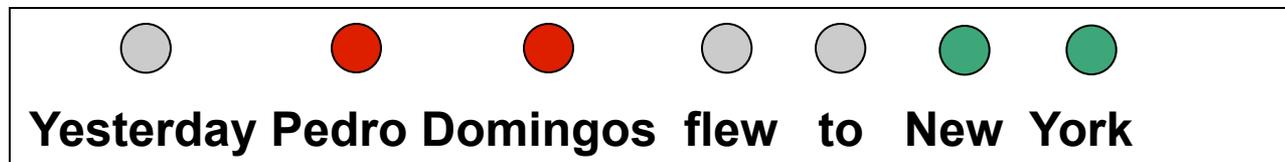
Most common approach: NER by classifying tokens

Feature	Value
isCapitalized	yes
numLetters	8
suffix2	-os
word-1-to-right	flew
word-2-to-right	to
...	
thisWord	Domingos

Given a sentence:

Yesterday Pedro Domingos flew to New

1) Break the sentence into *tokens*, and **classify** each token with a label indicating *what sort of entity* it is part of:



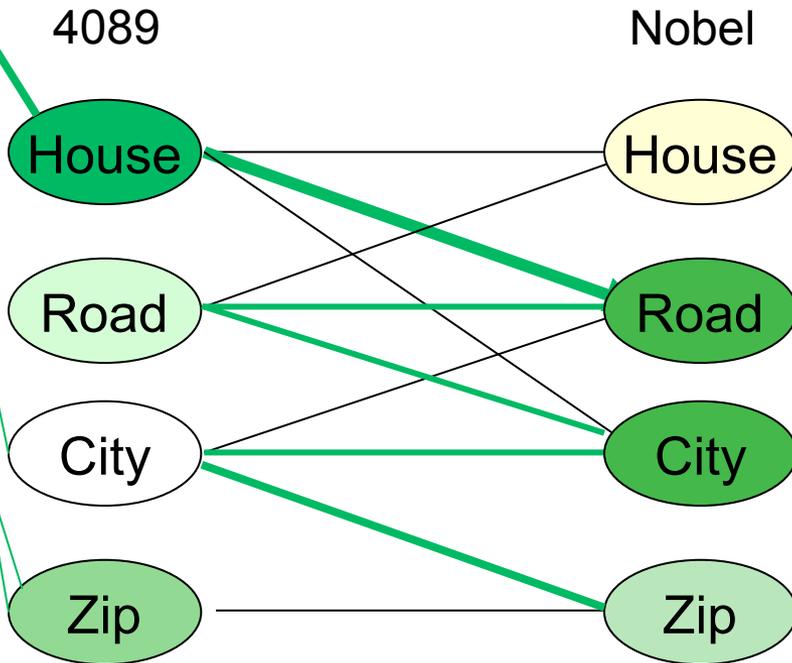
2) Identify names based on the entity labels

Person name: **Pedro Domingos**
Location name: **New York**

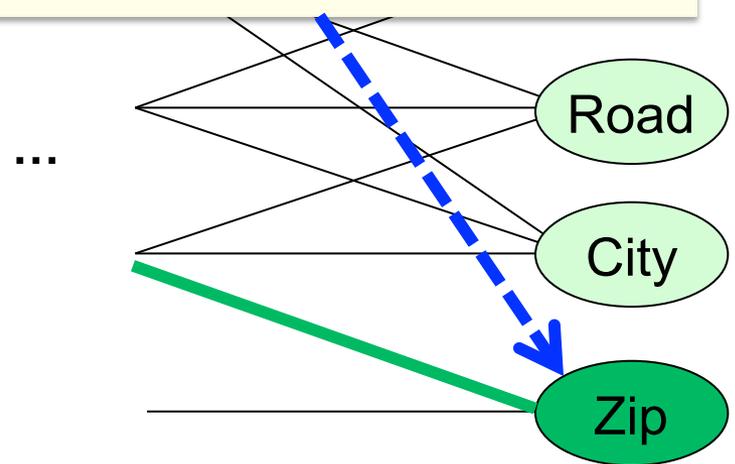
3) To learn an NER system, use YFCL and **whatever features you want....**

Back to pictures.....

4089 Nobel Drive San Diego 92122



x_i matches regex $[0-9]^+$ and $s_i=zip$
 x_i matches regex $\{[0-9]\}^5$ and $s_i=zip$
...
 x_i starts with capital and $s_i=road$
 x_i is in a city dictionary and $s_i=city$
...



Can we featurize the way that the edges and nodes are *weighted*?

Recurrent structures.....

we don't really need **anything but edge features**, because an edge feature could ignore part of the edge

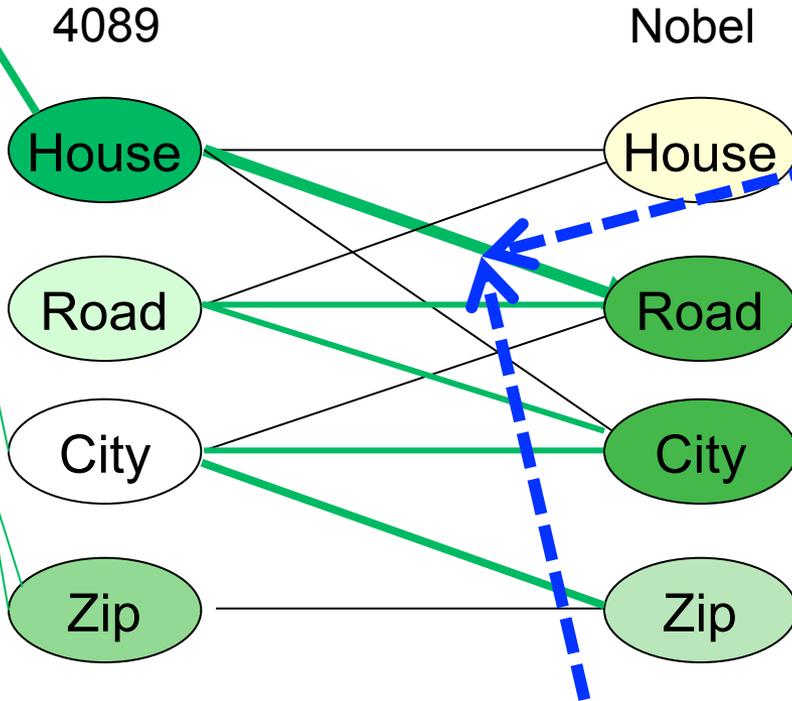
x_{i-1} is a digit and x_i is a capitalized word and $y_{i-1} = \text{house}$ and $y_i = \text{road}$ (f1)

$y_{i-1} = \text{house}$ and $y_i = \text{road}$ (f2)

...

this is the first transition and $y_{i-1} = \text{house}$ and $y_i = \text{road}$ (f37)

...



$$\text{weight}(Y_{i-1} = \text{house}, Y_i = \text{road}) = 0.23 * f1 - 0.61 * f2 + \dots$$

Zip

A possible learning algorithm....

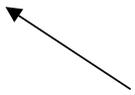
- Initialize feature weight vector λ
- For each labeled example:
 - use λ to compute edge weights, node weights for the forward-backward graph
 - use this “machine” to label \mathbf{x} with \mathbf{y}
 - e.g. using forward-backward, or something similar
 - if it gets the wrong answer, tweak λ to improve performance somehow
 - e.g. using gradient descent

The math: Multiclass logistic regression

$$\Pr(x, y) = \exp\left(\sum_i \lambda_i f_i(x, y)\right)$$

$$\Pr(y | x) = \frac{\exp\left(\sum_i \lambda_i f_i(x, y)\right)}{\sum_{y'} \exp\left(\sum_i \lambda_i f_i(x, y')\right)} = \frac{\exp\left(\sum_i \lambda_i f_i(x, y)\right)}{Z_\lambda(x)}$$

It's easy to compute this.



Gradient Descent for Logistic Regression

Old notation:

- In batch gradient descent, average the gradient over all the examples $D = \{(x_1, y_1), \dots, (x_n, y_n)\}$

$$\frac{\partial}{\partial w^j} \log P(D|\mathbf{w}) = \frac{1}{n} \sum_i (y_i - p_i) x_i^j =$$

$$= \frac{1}{n} \sum_{i: x_i^j=1} y_i - \frac{1}{n} \sum_{i: x_i^j=1} p_i$$

Multiclass Logistic Regression

New notation: $D = (x_1, y_1), \dots, (x_k, y_k), \dots$

$$\frac{\partial}{\partial w^j} \log P(D|\mathbf{w}) = \frac{1}{n} \sum_i (y_i - p_i) x_i^j$$

$$\begin{aligned} \frac{\partial}{\partial \lambda_i} \Pr(D | \vec{\lambda}) &= \sum_t \left(\delta[y_t = k] - p_\lambda(y_t = k | x_t) \right) f_i(x_t, y_t) \\ &= \sum_t f_i(x_t, y_t) - E_{p_\lambda(Y|x_t)} f_i(Y, x_t) \end{aligned}$$

From logistic regression to CRFs

Sha and Pereira, 2002

j is over positions in the sequence

$$P(\vec{y} | \vec{x}) = \frac{\prod_j \exp(\sum_i \lambda_i f_i(x_j, y_j, y_{j-1}))}{Z_\lambda(\vec{x})}$$

~~$$\Pr(y | x) = \frac{\exp(\sum_i \lambda_i f_i(x, y))}{Z_\lambda(x)}$$~~

$$= \frac{\exp(\sum_i \lambda_i F_i(\vec{x}, \vec{y}))}{Z_\lambda(\vec{x})}, \text{ where } F_i(\vec{x}, \vec{y}) = \sum_j f_i(x_j, y_j, y_{j-1})$$

Compute with forward-backward ideas

~~$$\frac{\partial}{\partial \lambda_i} \Pr(D | \vec{\lambda}) = \sum_t f_i(x_t, y_t) - E_{p_\lambda(Y|x_t)} f_i(Y, x_t)$$~~

$$\frac{\partial}{\partial \lambda_i} \Pr(D | \vec{\lambda}) = \sum_t F_i(\vec{x}_t, \vec{y}_t) - E_{p_\lambda(\vec{Y}|\vec{x}_t)} F_i(\vec{Y}, x_t)$$

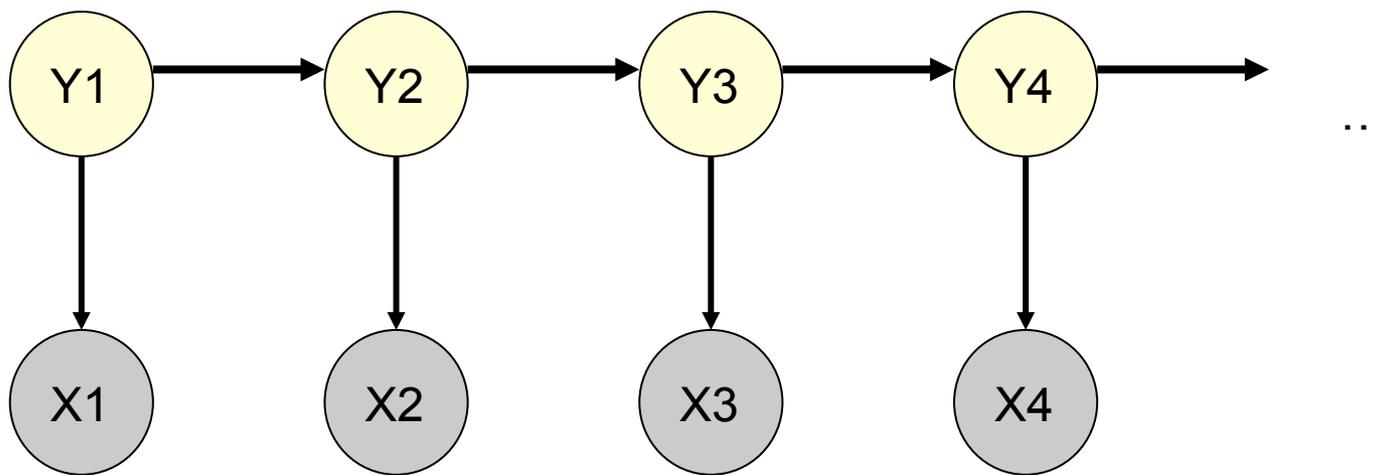
Conditional random fields

$$F_i(\vec{x}, \vec{y}) = \sum_j f_i(x_j, y_j, y_{j-1})$$

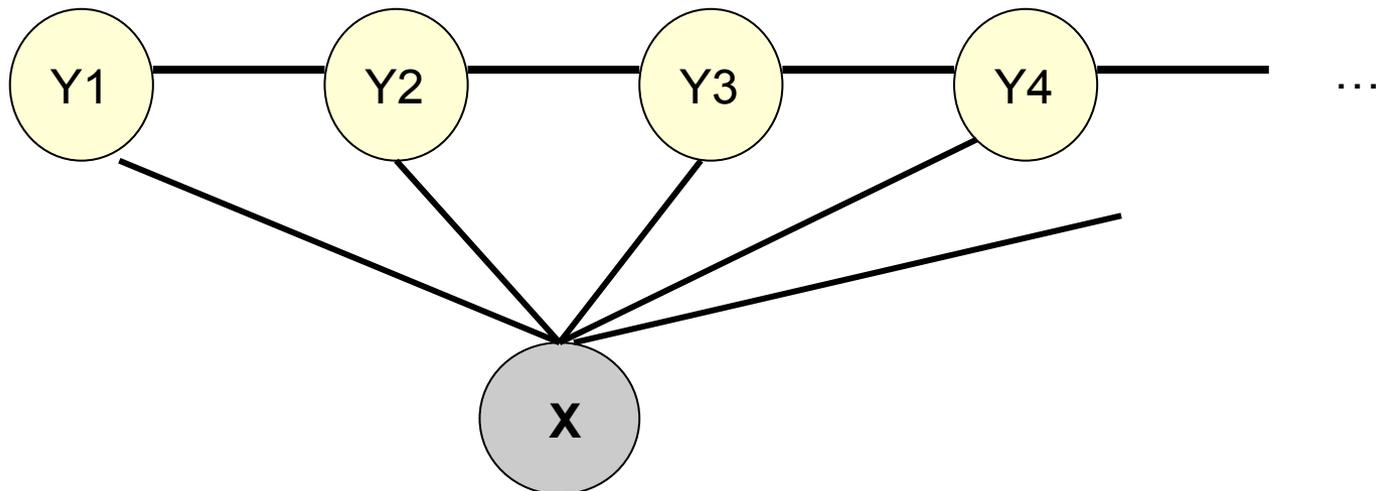
$$P(\vec{y} | \vec{x}) = \frac{\exp(\sum_i \lambda_i F_i(\vec{x}, \vec{y}))}{Z_\lambda(\vec{x})} = \frac{\exp(\sum_i \lambda_i \sum_j f_i(x_j, y_j, y_{j-1}))}{Z_\lambda(\vec{x})}$$

- Standard CRF learning method:
 - optimize λ to maximize (regularized) conditional log likelihood of the data under this model
 - computing gradient is done by running forward-backward on the data
 - instead of using the weights to “predict” each example’s score, as in logistic regression

HMM



CRF

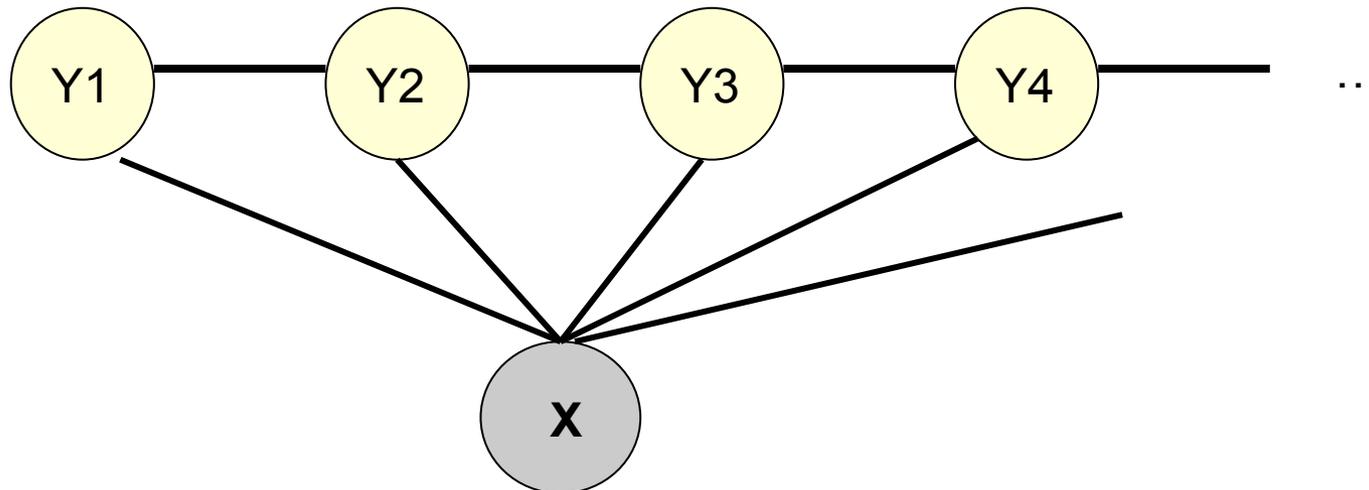


This is a special case of an *undirected graphical model*.

An undirected graphical model is also called a *Markov network*.

Independencies: every node A is independent of B given the *neighbors of A* (i.e., the nodes directly connected to A):

- For example: $\langle Y3, \{Y2, X, Y4\}, Y1 \rangle$



Conditional Random Fields: Summary

	Generative; models $\Pr(X,Y)$; estimate conditional probs directly	Conditional; models $\Pr(Y X)$; optimize data loglikelihood
Instances +labels \mathbf{x}, \mathbf{y}	Naïve Bayes	Logistic regression
Sequences of instances + sequences of labels \mathbf{x}, \mathbf{y}	HMMs	CRFs