NLP

Introduction to NLP

221.

Hidden Markov Models

Markov Models

- Sequence of random variables that aren’t independent
- Examples
  - Weather reports
  - Text
  - Stock market numbers

Definition

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} a_{12} \ldots a_{n1} \ldots a_{nn}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
$\pi = \pi_1, \pi_2, \ldots, \pi_N$	an **initial probability distribution** over states. π_i is the probability that the Markov chain will start in state i. Some states j may have $\pi_j = 0$, meaning that they cannot be initial states. Also, $\sum_{i=1}^{n} \pi_i = 1$

Properties

Andrey Markov

- Limited horizon:

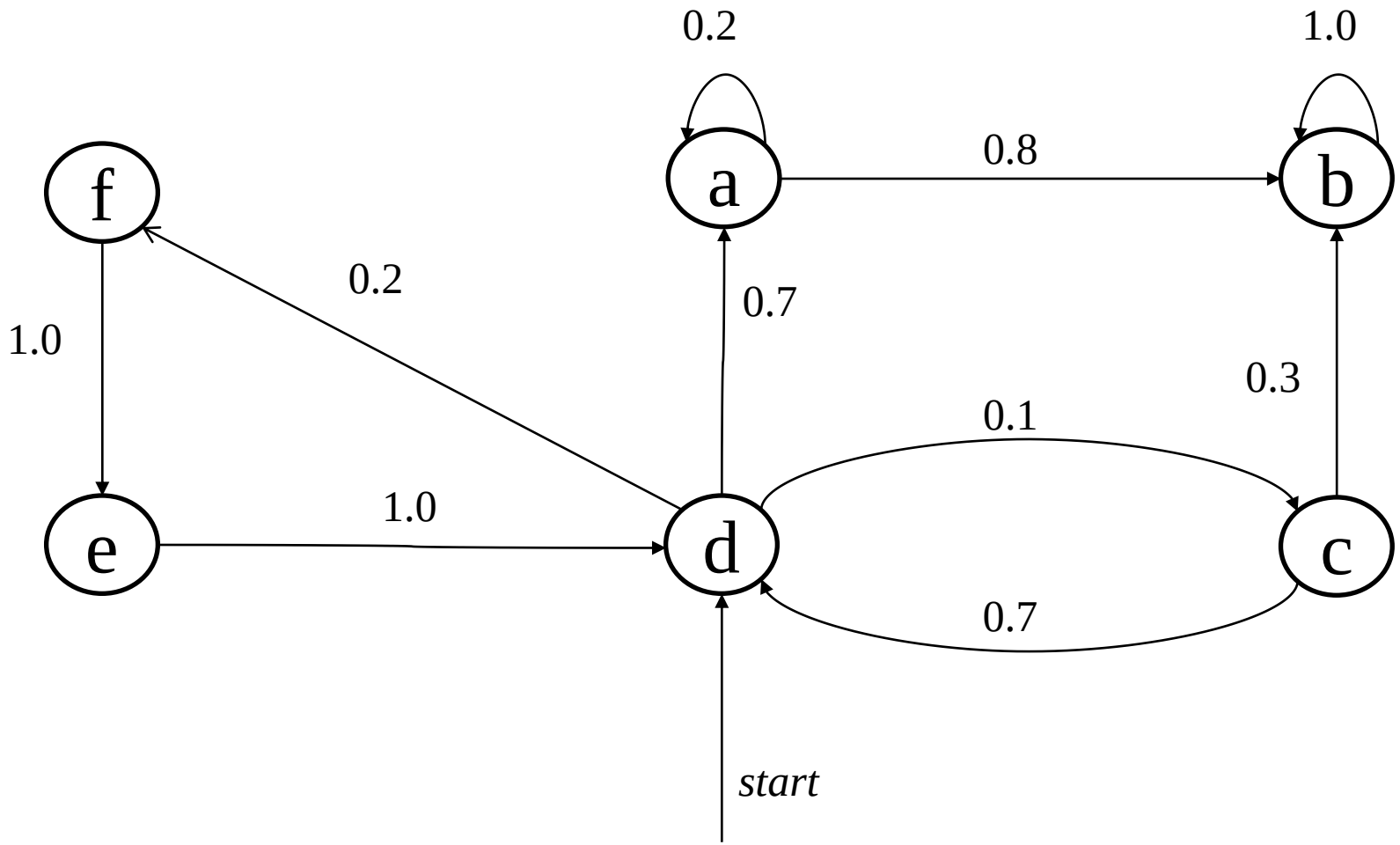
  $P(X_{t+1} = s_k | X_1,\ldots,X_t) = P(X_{t+1} = s_k | X_t)$

- Time invariant (stationary)

  $P(X_{t+1} = s_k | X_t) = P(X_2 = s_k | X_1)$

- Definition
  - in terms of a transition matrix A and initial state probabilities Π.

Example



Visible MM

$P(X_1,\ldots X_T) = P(X_1)\, P(X_2|X_1)\, P(X_3|X_1,X_2) \ldots P(X_T|X_1,\ldots,X_{T-1})$

$= P(X_1)\, P(X_2|X_1)\, P(X_3|X_2) \ldots P(X_T|X_{T-1})$

$$= \rho_{X_1} \tilde{\prod}_{t=1}^{T-1} a_{X_t X_{t+1}}$$

$P(d, a, b) = P(X_1=d)\, P(X_2=a|X_1=d)\, P(X_3=b|X_2=a)$

$= 1.0 \times 0.7 \times 0.8$

$= 0.56$

Hidden Markov Models

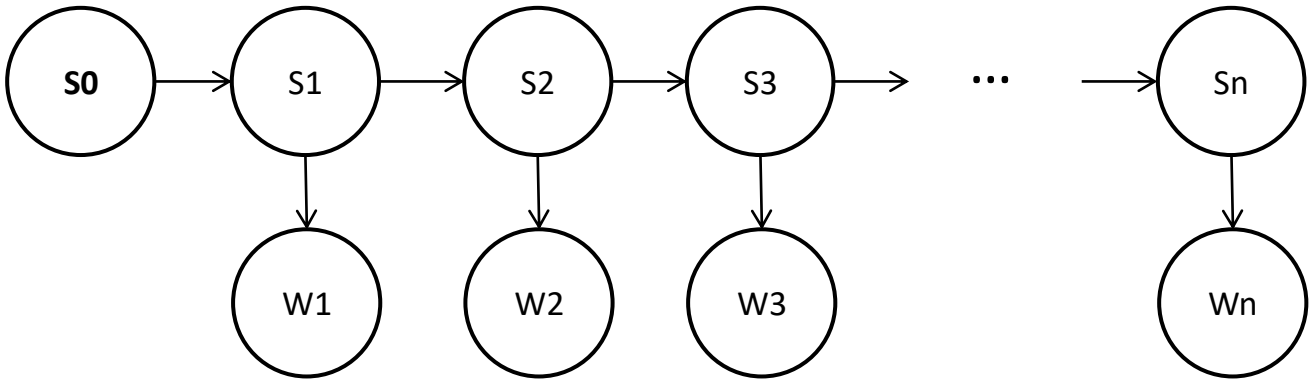
- Motivation
  - Observing a sequence of symbols
  - The sequence of states that led to the generation of the symbols is hidden
  - The states correspond to hidden (latent) variables
- Definition
  - Q = states
  - O = observations, drawn from a vocabulary
  - q_0,q_f = special (start, final) states
  - A = state transition probabilities
  - B = symbol emission probabilities
  - Π = initial state probabilities
  - μ = (A,B,Π) = complete probabilistic model

Hidden Markov Models

- Uses
  - Part of speech tagging
  - Speech recognition
  - Gene sequencing

Hidden Markov Models

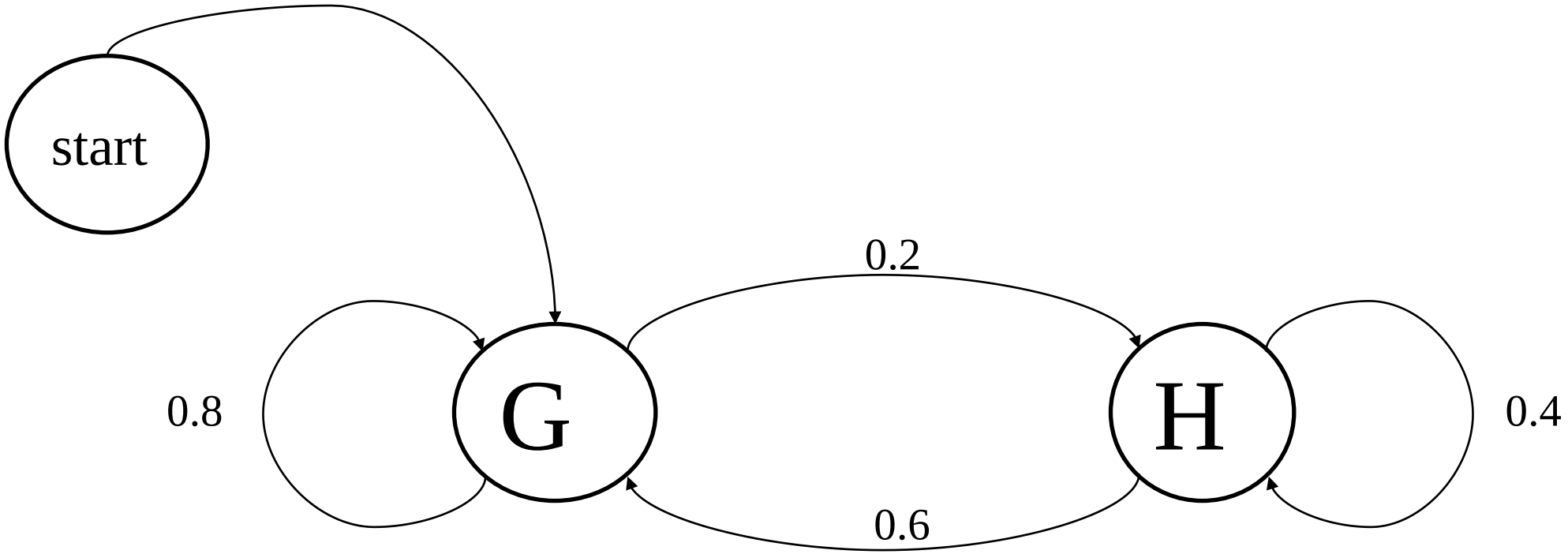
- Can be used to model state sequences and observation sequences
- Example:
  - $P(\mathbf{s},\mathbf{w}) = \Pi_i P(s_i|s_{i-1})P(w_i|s_i)$



Generative Algorithm

- Pick start state from Π
- For t = 1..T
  - Move to another state based on A
  - Emit an observation based on B

State Transition Probabilities



Emission Probabilities

- $P(O_t=k|X_t=s_i,X_{t+1}=s_j) = b_{ijk}$

	x	**y**	**z**
G	0.7	0.2	0.1
H	0.3	0.5	0.2

All Parameters of the Model

- Initial
  - P(G|start) = 1.0, P(H|start) = 0.0
- Transition
  - P(G|G) = 0.8, P(G|H) = 0.6, P(H|G) = 0.2, P(H|H) = 0.4
- Emission
  - P(x|G) = 0.7, P(y|G) = 0.2, P(z|G) = 0.1
  - P(x|H) = 0.3, P(y|H) = 0.5, P(z|H) = 0.2

Observation sequence “yz”

- Starting in state G (or H), P(yz) = ?
- Possible sequences of states:
  - GG
  - GH
  - HG
  - HH
- P(yz) = P(yz|GG) + P(yz|GH) + P(yz|HG) + P(yz|HH) =

  = .8 x .2 x .8 x .1

  + .8 x .2 x .2 x .2

  + .2 x .5 x .4 x .2

  + .2 x .5 x .6 x .1

  = .0128+.0064+.0080+.0060 =.0332

Hidden Markov Model

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

States and Transitions

- An HMM is essentially a weighted finite-state transducer
  - The states encode the most recent history
  - The transitions encode likely sequences of states
    - e.g., Adj-Noun or Noun-Verb
    - or perhaps Art-Adj-Noun
  - Use MLE to estimate the probabilities
- Another way to think of an HMM
  - It’s a natural extension of Naïve Bayes to sequences

Emissions

- Estimating the emission probabilities
  - Harder than transition probabilities (why?)
  - There may be novel uses of word/POS combinations
- Suggestions
  - It is possible to use standard smoothing
  - As well as heuristics (e.g., based on the spelling of the words)

Sequence of Observations

- The observer can only see the emitted symbols
- Observation likelihood
  - Given the observation sequence S and the model $\mu = (A,B,\Pi)$, what is the probability $P(S|\mu)$ that the sequence was generated by that model.
- Being able to compute the probability of the observations sequence turns the HMM into a language model

Tasks with HMM

- Given $\mu = (A,B,\Pi)$, find $P(O|\mu)$
  - Uses the Forward Algorithm
- Given O, μ, find $(X_1,\ldots X_{T+1})$
  - Uses the Viterbi Algorithm
- Given O and a space of all possible $\mu_{1..m}$, find model μ_i that best describes the observations
  - Uses Expectation-Maximization

Inference

- Find the most likely sequence of tags, given the sequence of words
  - $t^* = \text{argmax}_t P(t|w)$
- Given the model μ, it is possible to compute P (t|w) for all values of t
  - In practice, there are way too many combinations
- Greedy Search
- Beam Search
  - One possible solution
  - Uses partial hypotheses
  - At each state, only keep the k best hypotheses so far
  - May not work

Viterbi Algorithm

- Find the best path up to observation i and state s
- Characteristics
  - Uses dynamic programming
  - Memoization
  - Backpointers

The **Viterbi** algorithm was first applied to speech and language processing in the context of speech recognition by Vintsyuk (1968) but has what Kruskal (1983) calls a "remarkable history of multiple independent discovery and publication". Kruskal and others give at least the following independently-discovered variants of the algorithm published in four separate fields:

Citation	Field
Viterbi (1967)	information theory
Vintsyuk (1968)	speech processing
Needleman and Wunsch (1970)	molecular biology
Sakoe and Chiba (1971)	speech processing
Sankoff (1972)	molecular biology
Reichert et al. (1973)	molecular biology
Wagner and Fischer (1974)	computer science

Algorithm 12 Generative process for the hidden Markov model

$y_0 \leftarrow \lozenge, \quad m \leftarrow 1$

repeat

$y_m \sim \text{Categorical}(\boldsymbol{\lambda}_{y_{m-1}})$ ▷ sample the current tag

$w_m \sim \text{Categorical}(\boldsymbol{\phi}_{y_m})$ ▷ sample the current word

until $y_m = \blacklozenge$ ▷ terminate when the stop symbol is generated

Viterbi Algorithm

Finally, we can give a formal definition of the Viterbi recursion as follows:

1. **Initialization:**

$$\begin{aligned} v_1(j) &= \pi_j b_j(o_1) \qquad 1 \le j \le N \\ bt_1(j) &= 0 \qquad\qquad\quad 1 \le j \le N \end{aligned}$$

2. **Recursion**

$$\begin{aligned} v_t(j) &= \max_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \le j \le N, 1 < t \le T \\ bt_t(j) &= \operatorname*{argmax}_{i=1}^{N} v_{t-1}(i)\, a_{ij}\, b_j(o_t); \quad 1 \le j \le N, 1 < t \le T \end{aligned}$$

3. **Termination:**

$$\text{The best score:} \quad P* = \max_{i=1}^{N} v_T(i)$$

$$\text{The start of backtrace:} \quad q_T* = \operatorname*{argmax}_{i=1}^{N} v_T(i)$$

Algorithm 11 The Viterbi algorithm. Each $s_m(k, k')$ is a local score for tag $y_m = k$ and $y_{m-1} = k'$.

---

for $k \in \{0, \ldots K\}$ **do**

  $v_1(k) = s_1(k, \Diamond)$

for $m \in \{2, \ldots, M\}$ **do**

  for $k \in \{0, \ldots, K\}$ **do**

    $v_m(k) = \max_{k'} s_m(k, k') + v_{m-1}(k')$

    $b_m(k) = \text{argmax}_{k'} s_m(k, k') + v_{m-1}(k')$

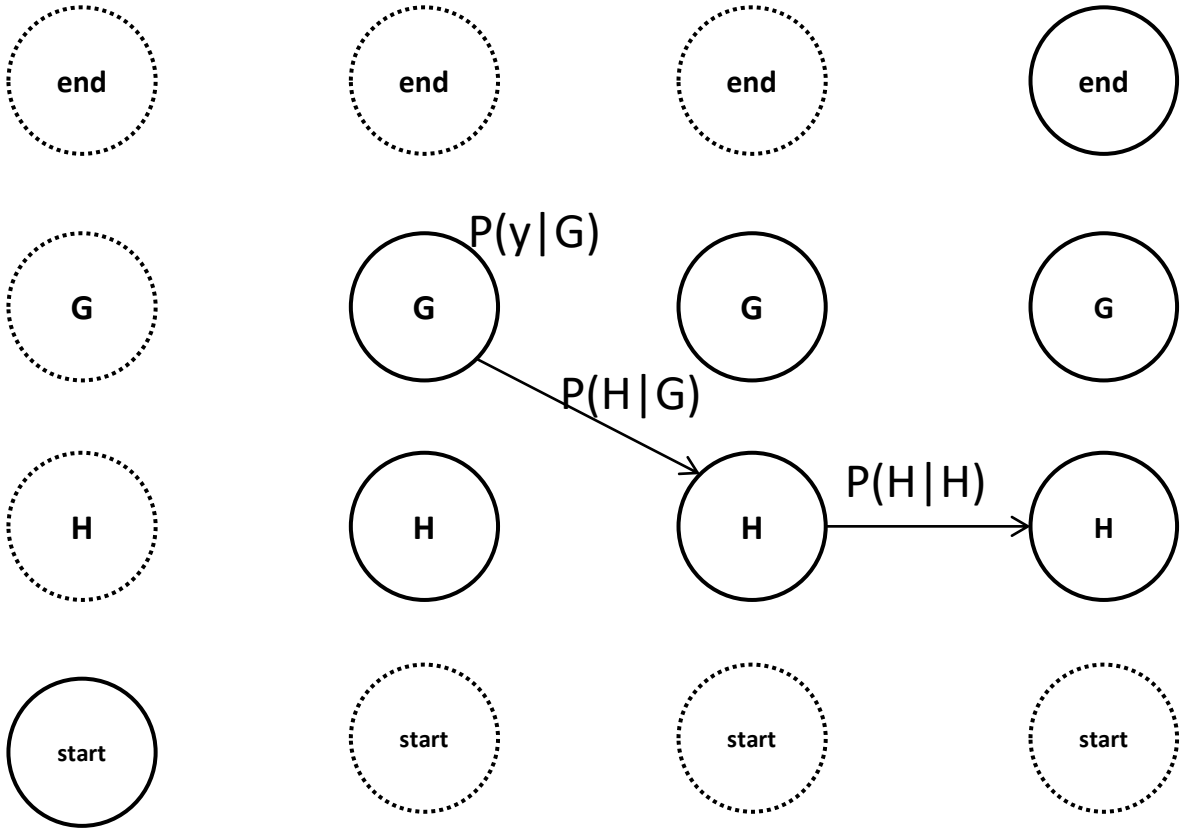
$y_M = \text{argmax}_k s_{M+1}(\blacklozenge, k) + v_M(k)$

for $m \in \{M - 1, \ldots 1\}$ **do**

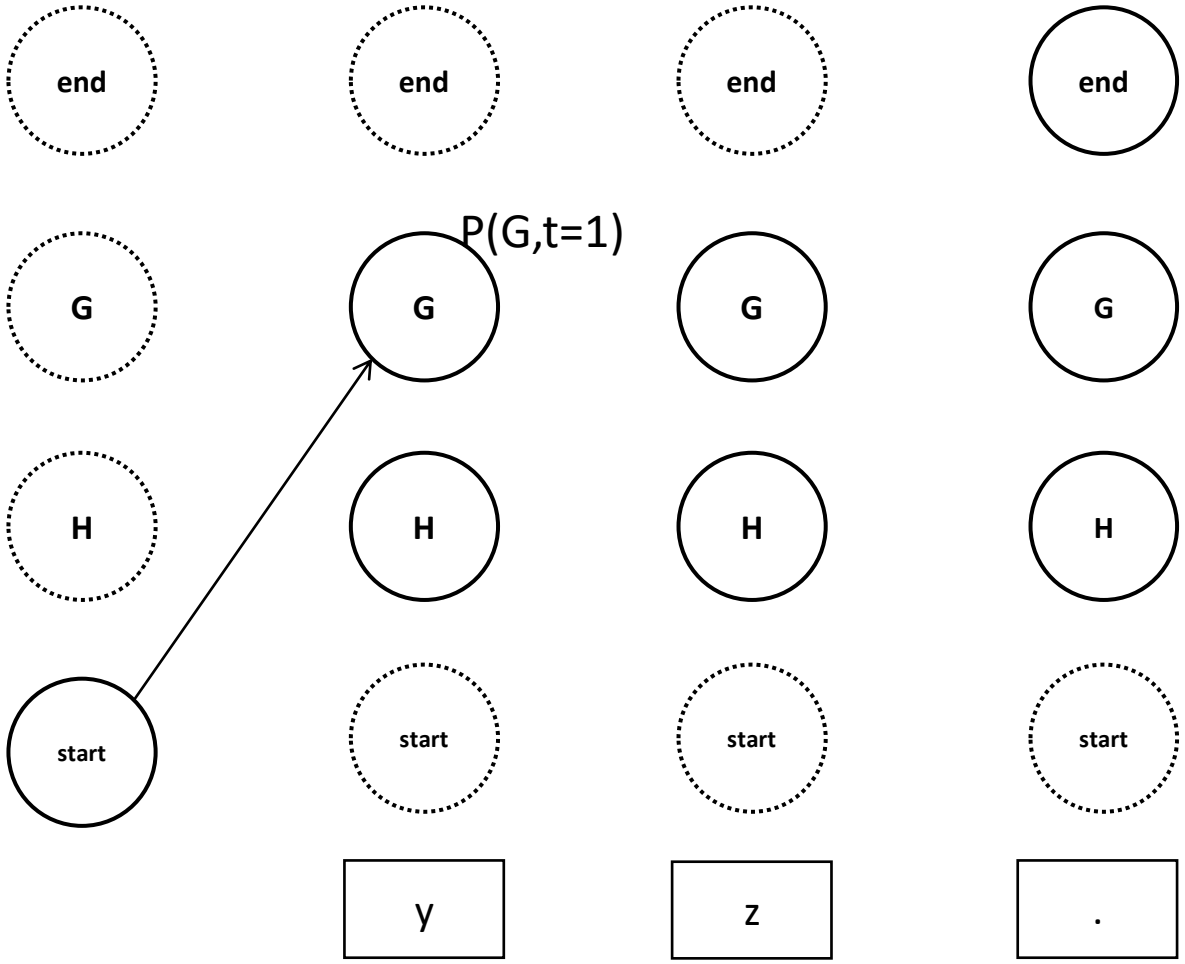
  $y_m = b_m(y_{m+1})$

return $\boldsymbol{y}_{1:M}$

HMM Trellis

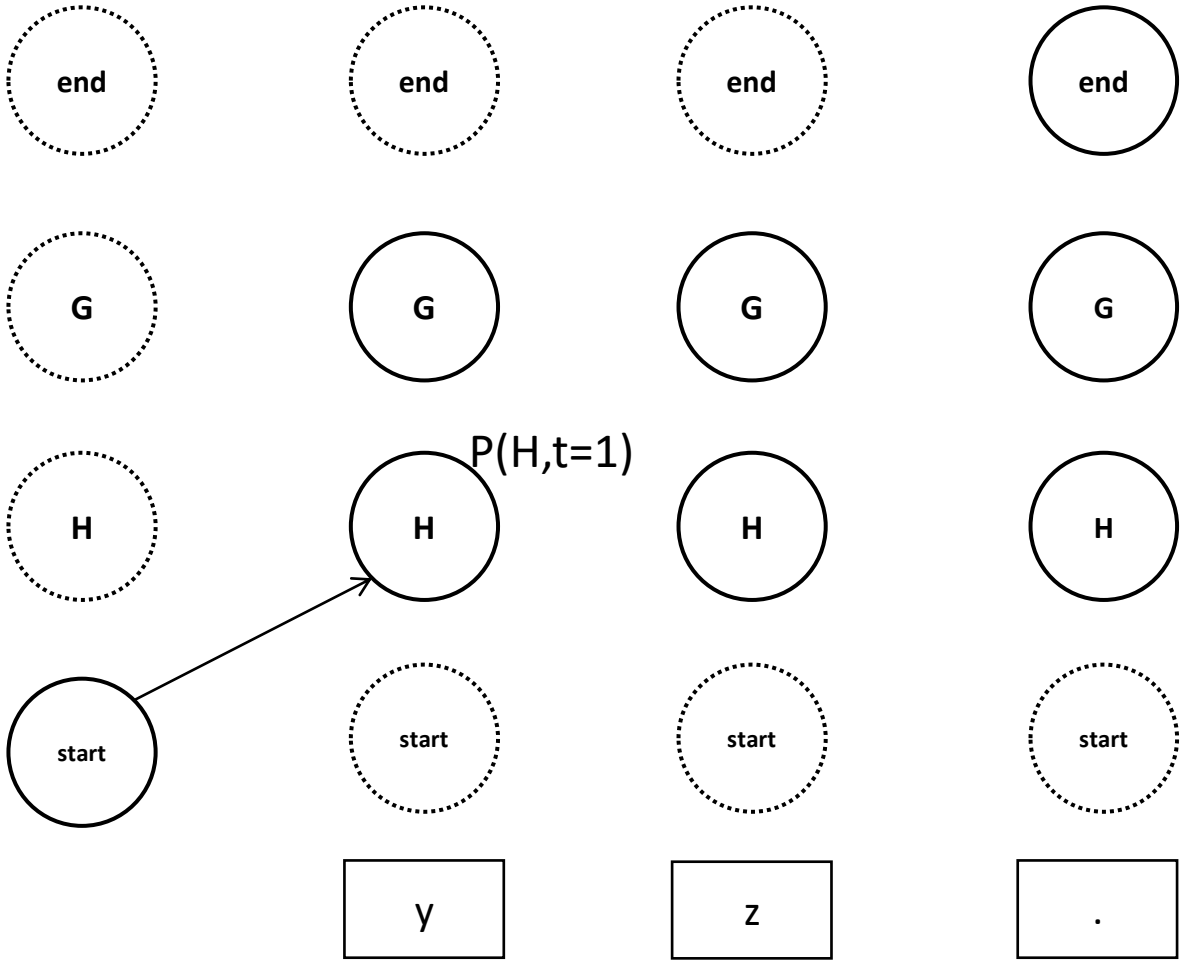


HMM Trellis



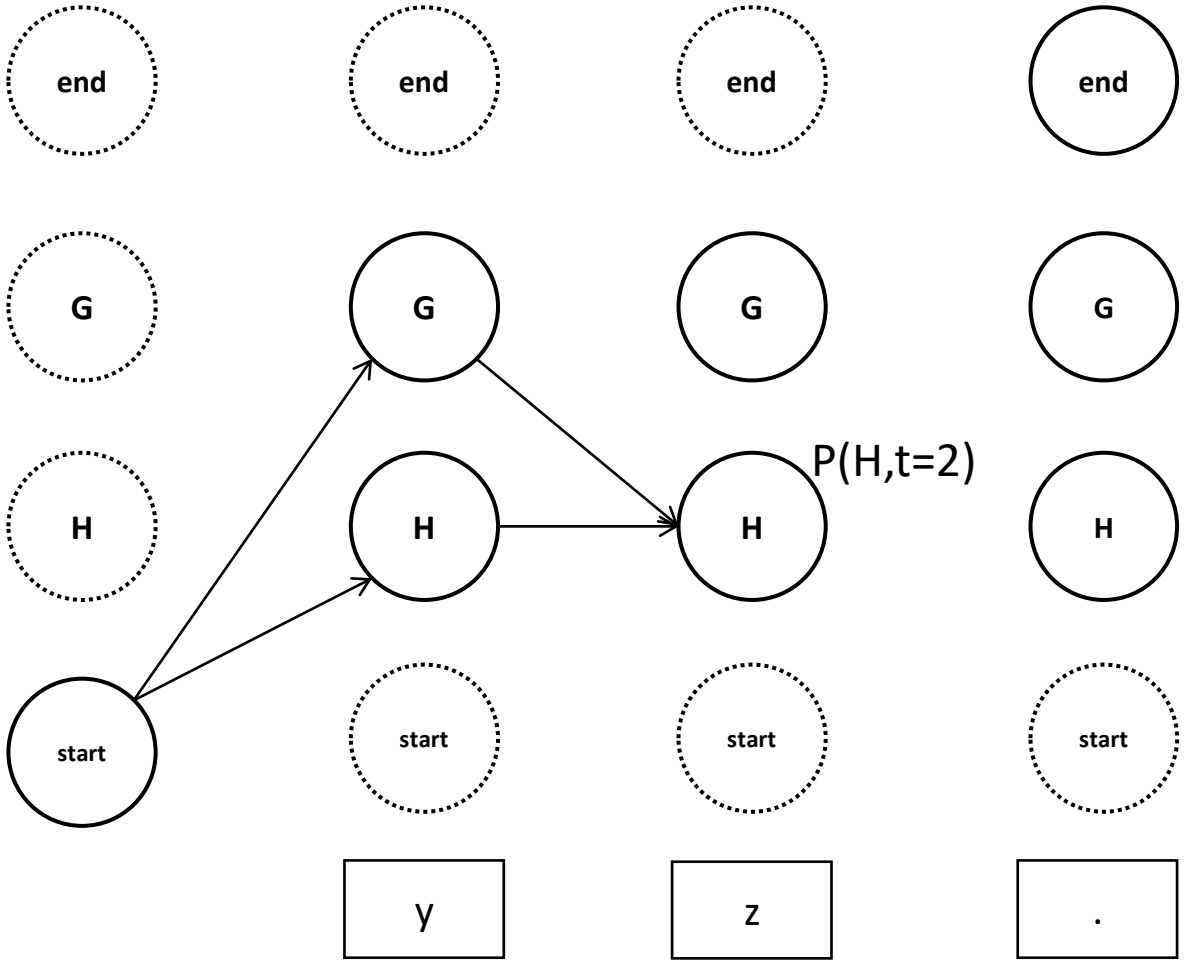
P(G,t=1) =
P(start) x P(G|start) x P(y|G)

HMM Trellis



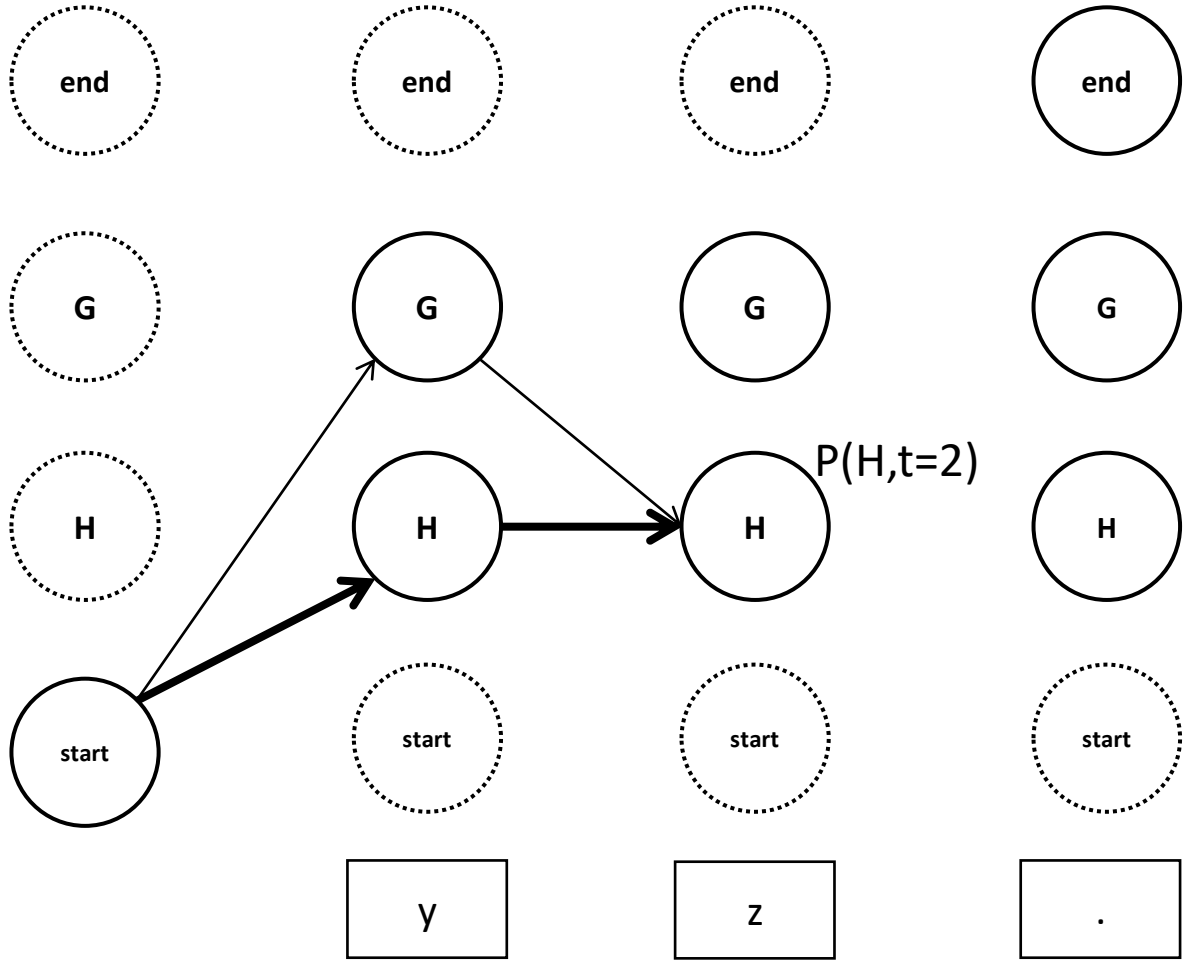
P(H,t=1) =
P(start) x P(H|start) x P(y|H)

HMM Trellis

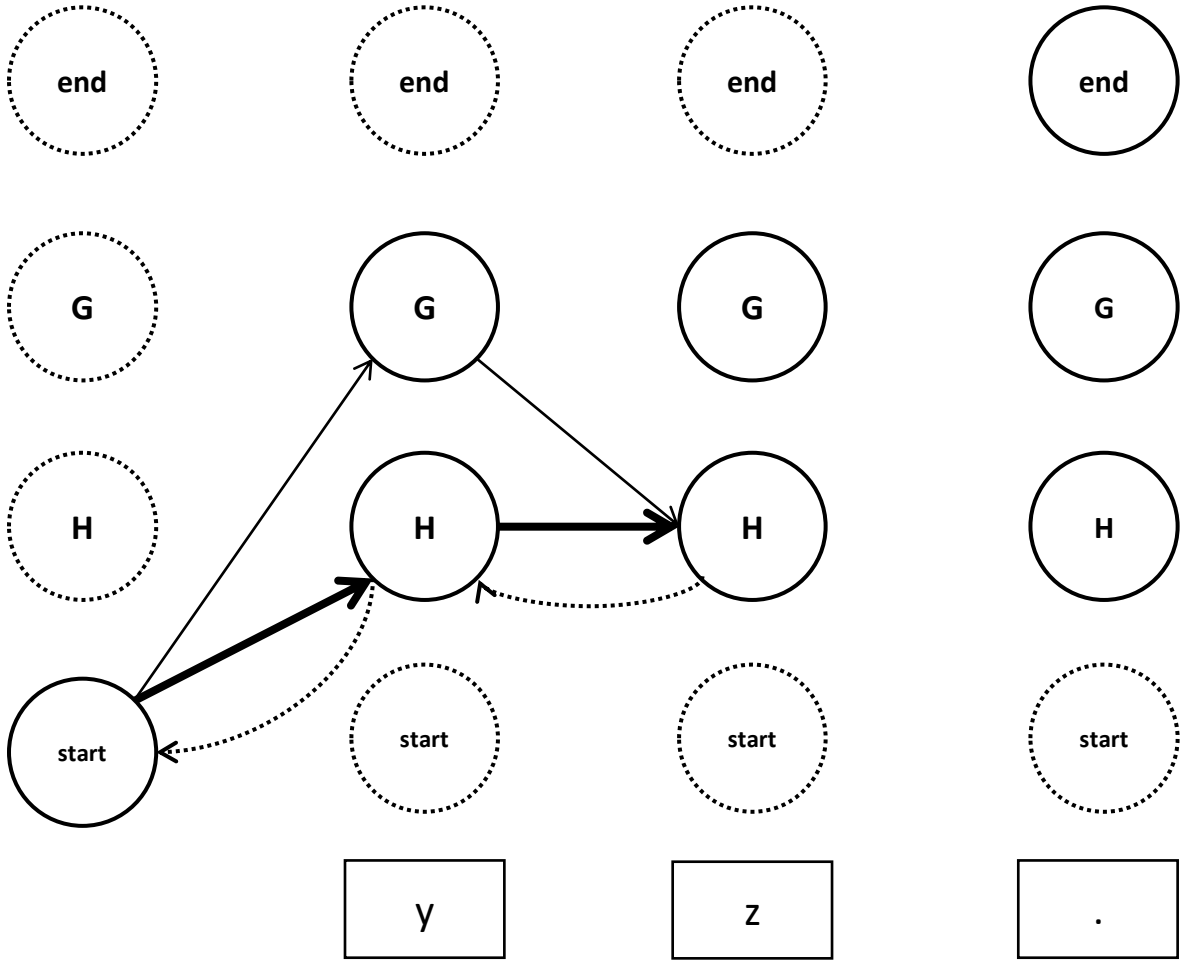


P(H,t=2) =
max (P(G,t=1) x P(H|G) x P(z|H),
P(H,t=1) x P(H|H) x P(z|H))

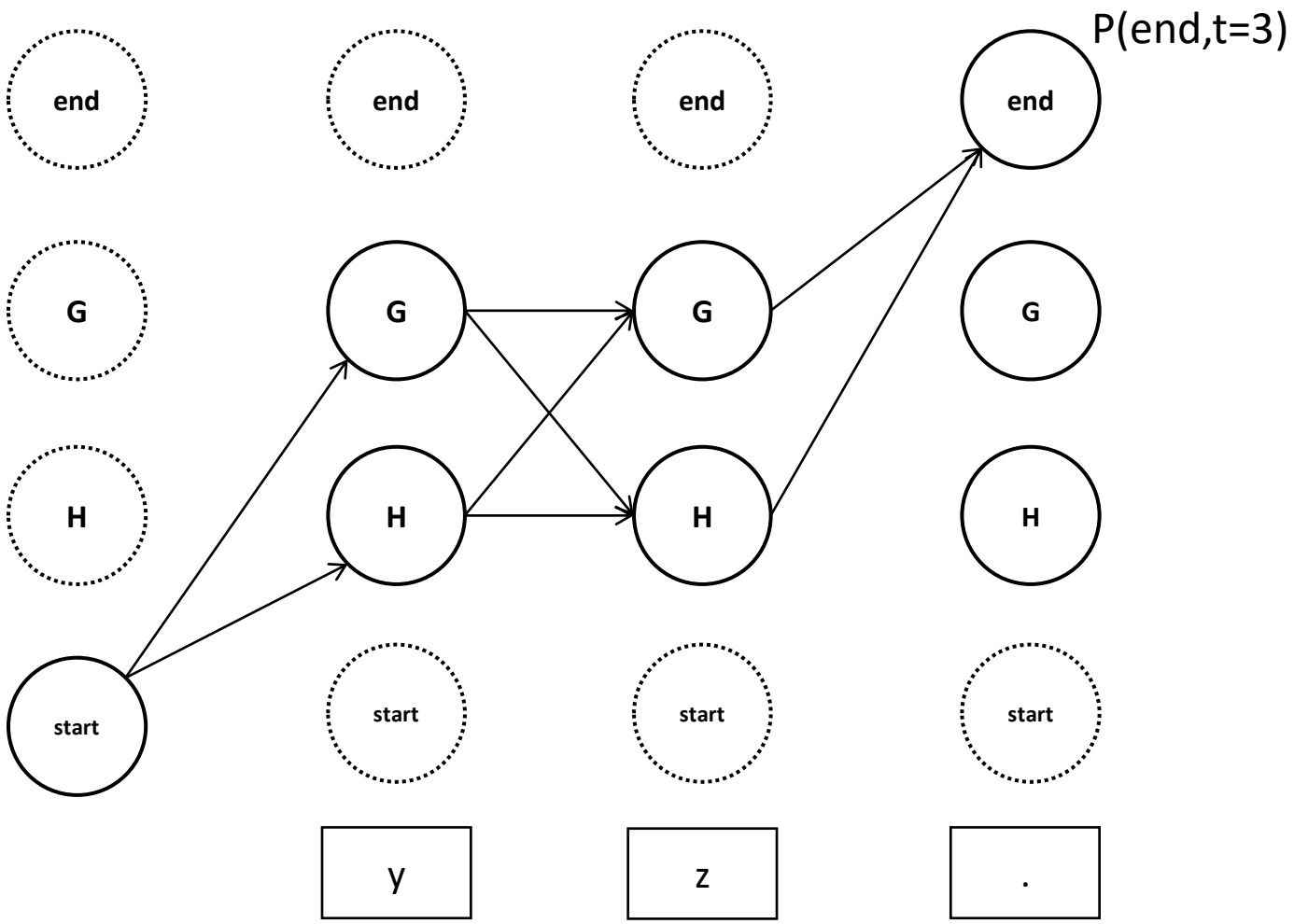
HMM Trellis



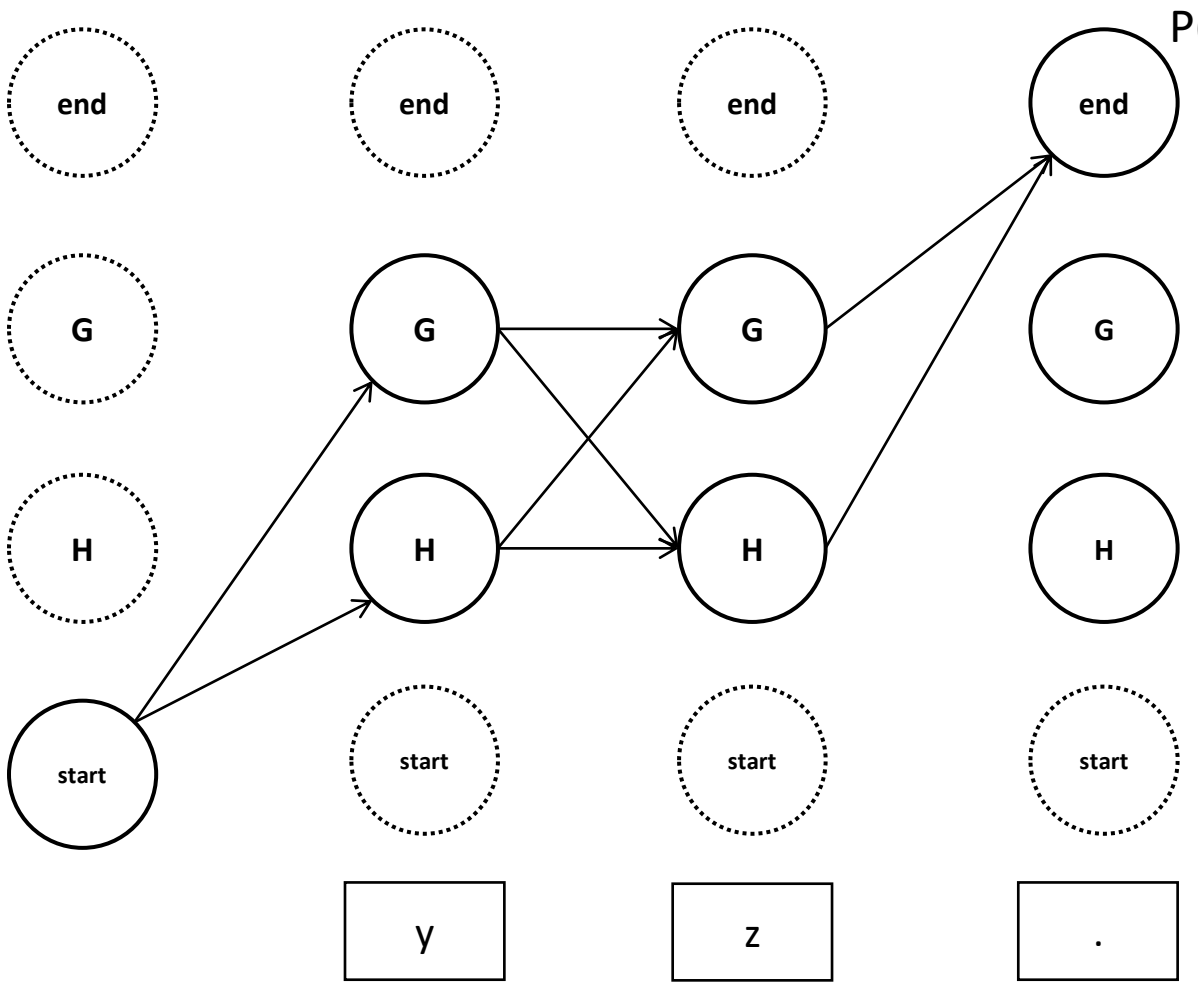
HMM Trellis



HMM Trellis



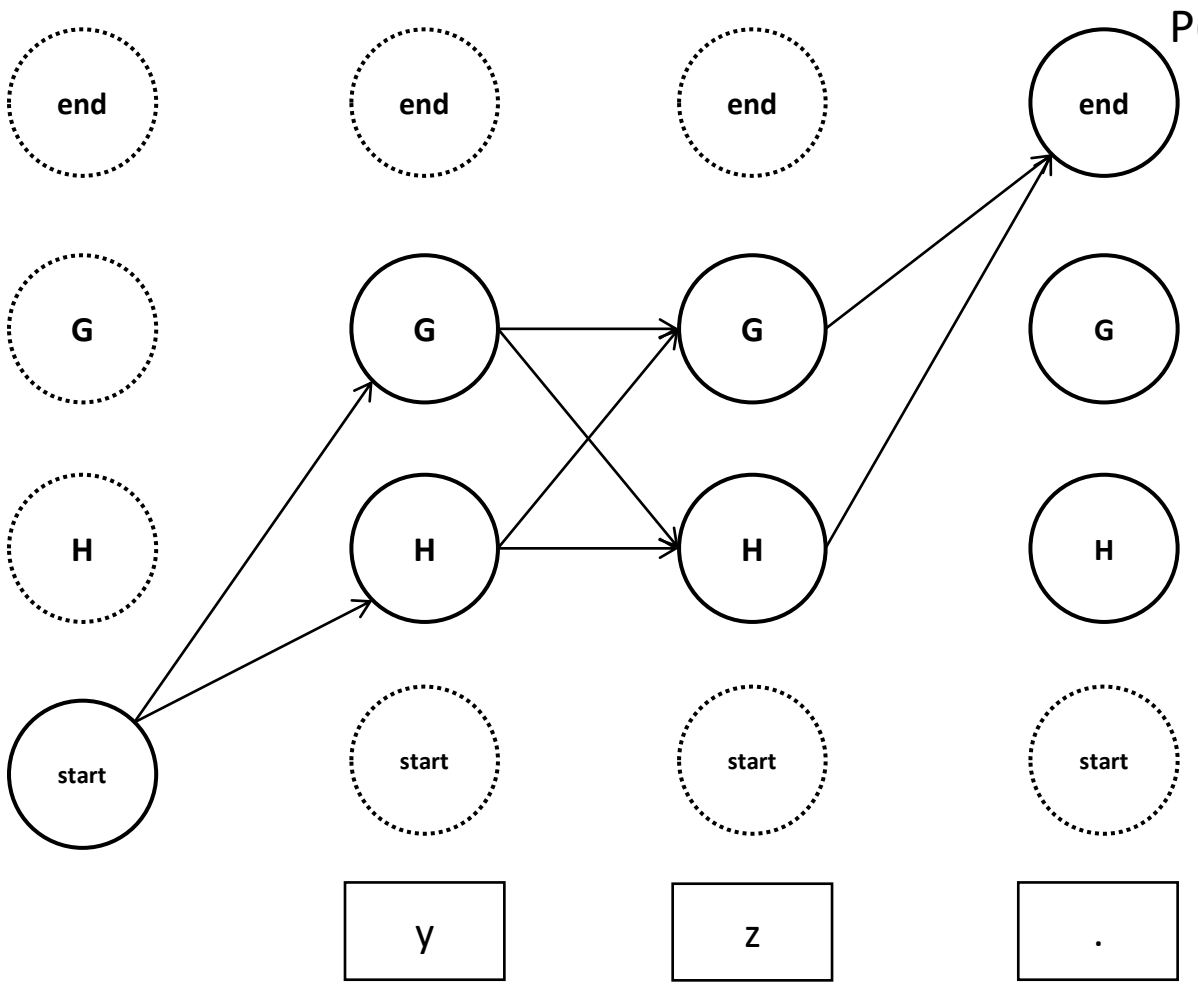
HMM Trellis



P(end,t=3)

P(end,t=3) =
max (P(G,t=2) x P(end|G),
P(H,t=2) x P(end|H))

HMM Trellis



P(end,t=3)

P(end,t=3) =
max (P(G,t=2) x P(end|G),
P(H,t=2) x P(end|H))

P(end,t=3) = best score for the sequence

Use the backpointers to find the sequence of states.

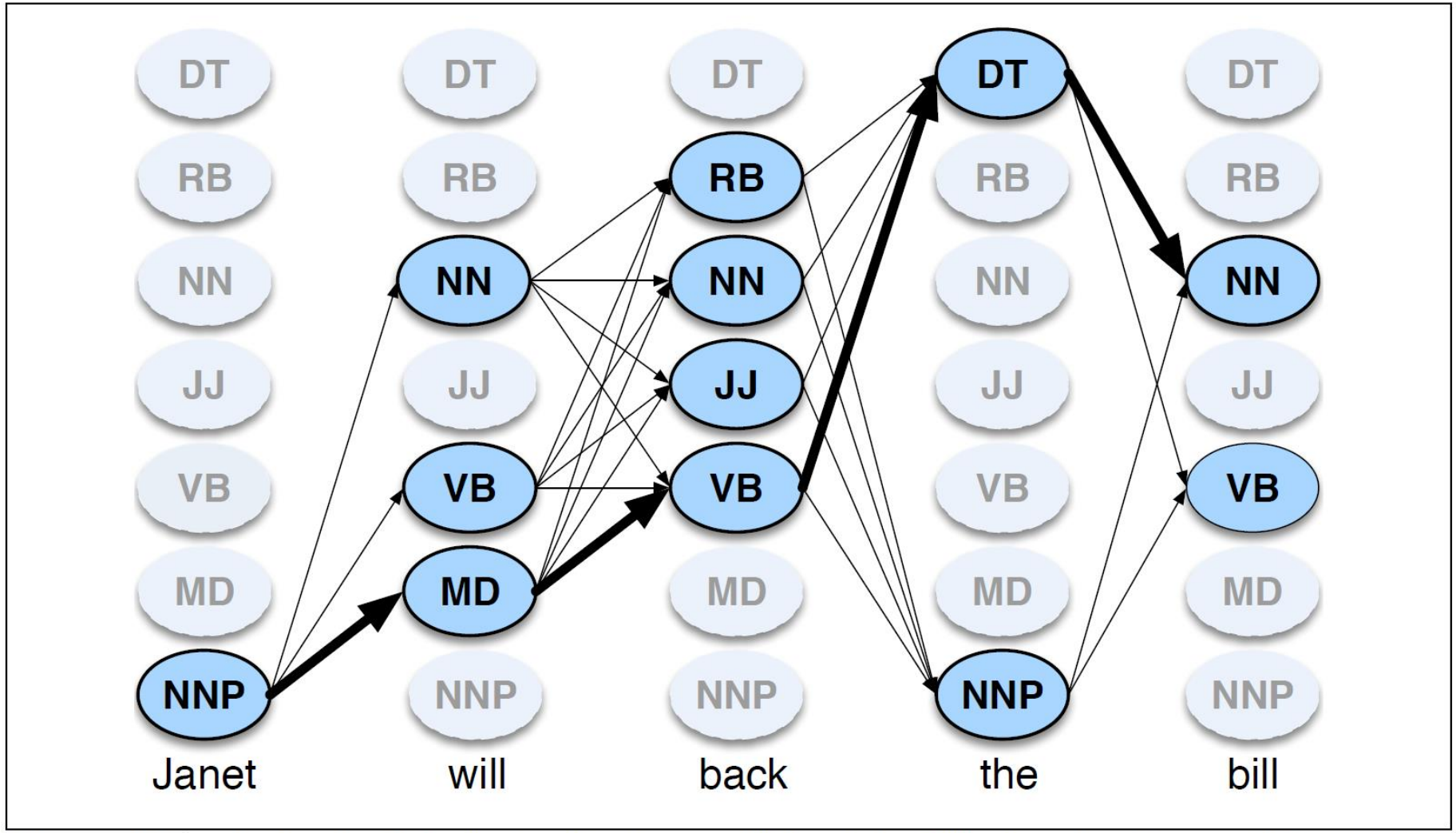


Figure 8.6 A sketch of the lattice for *Janet will back the bill*, showing the possible tags (q_i) for each word and highlighting the path corresponding to the correct tag sequence through the hidden states. States (parts of speech) which have a zero probability of generating a particular word according to the B matrix (such as the probability that a determiner DT will be realized as *Janet*) are greyed out.

Janet/NNP will/MD back/VB the/DT bill/NN

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

Figure 8.7 The A transition probabilities $P(t_i|t_{i-1})$ computed from the WSJ corpus without smoothing. Rows are labeled with the conditioning event; thus $P(VB|MD)$ is 0.7968.

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

Figure 8.8 Observation likelihoods B computed from the WSJ corpus without smoothing, simplified slightly.

Beam Search

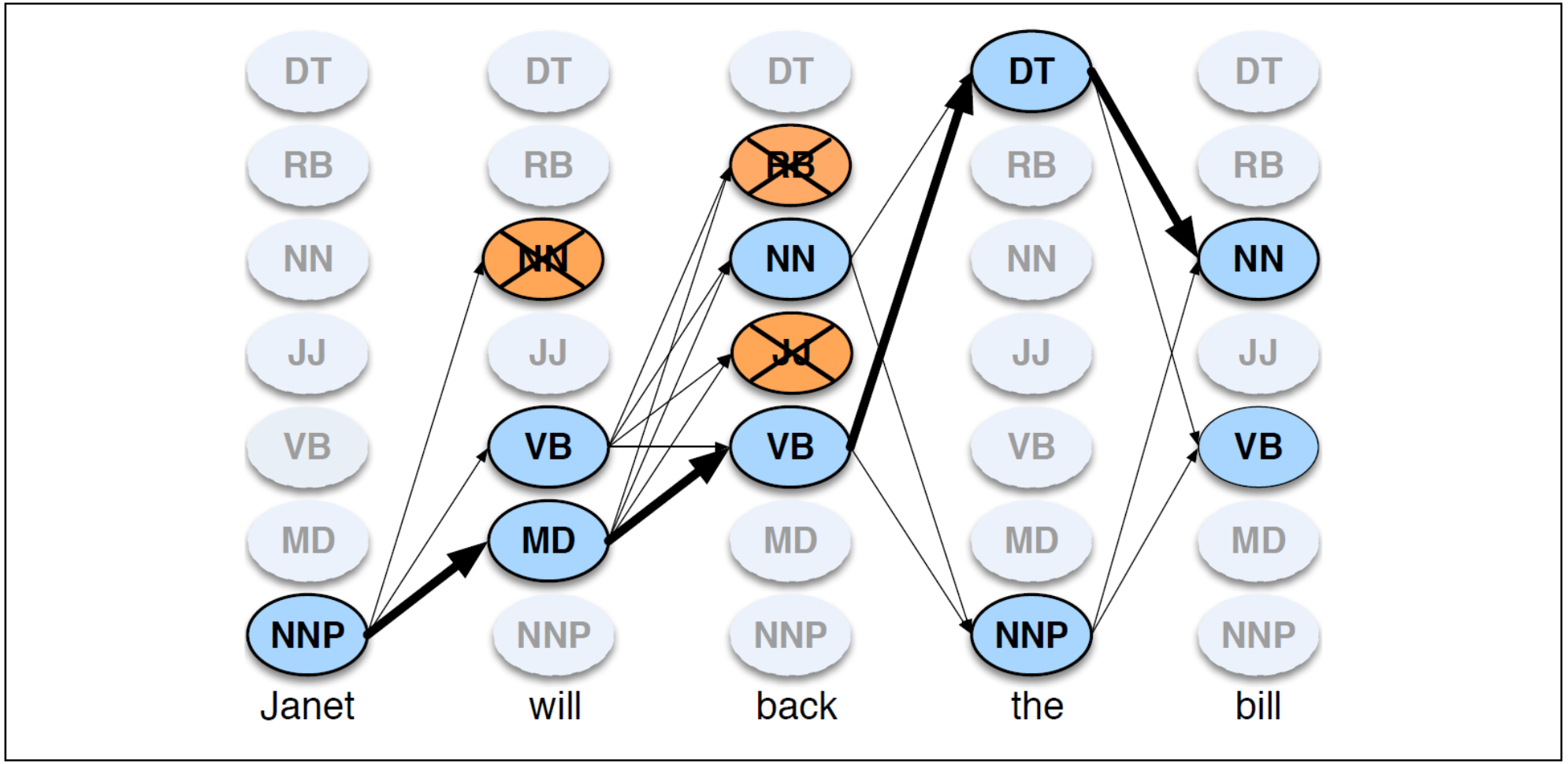


Figure 8.11 A beam search version of Fig. 8.6, showing a beam width of 2. At each time t, all (non-zero) states are computed, but then they are sorted and only the best 2 states are propagated forward and the rest are pruned, shown in orange.

Some Observations

- Advantages of HMMs
  - Relatively high accuracy
  - Easy to train
- Higher-Order HMM
  - The previous example was about bigram HMMs
  - How can you modify it to work with trigrams?

How to compute P(O)

- Viterbi was used to find the most likely sequence of states that matches the observation
- What if we want to find all sequences that match the observation
- We can add their probabilities (because they are mutually exclusive) to form the probability of the observation
- This is done using the Forward Algorithm

The Forward Algorithm

- Used to compute the probability of a sequence
- Very similar to Viterbi
- Instead of *max* we use *sum*

init $t = 0$, transition matrix x_{ij}, emission probabilities, $p(y_j|x_i)$, observed sequence, $y(1:t)$

```
for t = t + 1
    α_t(x_t) = p(y_t|x_t) Σ_{x_{t-1}} p(x_t|x_{t-1}) α_{t-1}(x_{t-1}) .
until t=T
```

return $p(y(1:t)) = \alpha_T$

NLP

Introduction to NLP

222.

Learning in Hidden Markov Models

HMM Learning

- Supervised
  - Training sequences are labeled
- Unsupervised
  - Training sequences are unlabeled
  - Known number of states
- Semi-supervised
  - Some training sequences are labeled

Supervised HMM Learning

- Estimate the static transition probabilities using MLE

$$a_{ij} = \frac{Count(q_t = s_i, q_{t+1} = s_j)}{Count(q_t = s_i)}$$

- Estimate the observation probabilities using MLE

$$b_j(k) = \frac{Count(q_i = s_j, o_i = v_k)}{Count(q_i = s_j)}$$

- Use smoothing

Unsupervised HMM Training

- Given:
  - observation sequences
- Goal:
  - build the HMM
- Use EM (Expectation Maximization) methods
  - forward-backward (Baum-Welch) algorithm
  - Baum-Welch finds an approximate solution for $P(O|\mu)$

Outline of Baum-Welch

- Algorithm
  - Randomly set the parameters of the HMM
  - Until the parameters converge repeat:
    - E step – determine the probability of the various state sequences for generating the observations
    - M step – reestimate the parameters based on these probabilities
- Notes
  - the algorithm guarantees that at each iteration the likelihood of the data $P(O|\mu)$ increases
  - it can be stopped at any point and give a partial solution
  - it converges to a local maximum

NLP

Introduction to NLP

231.

Statistical POS Tagging

HMM Tagging

- T = argmax P(T|W)
  - where $T=t_1,t_2,\ldots,t_n$
- By Bayes' theorem
  - P(T|W) = P(T)P(W|T)/P(W)
- Thus we are attempting to choose the sequence of tags that maximizes the RHS of the equation
  - P(W) can be ignored
  - P(T) is called the prior, P(W|T) is called the likelihood.

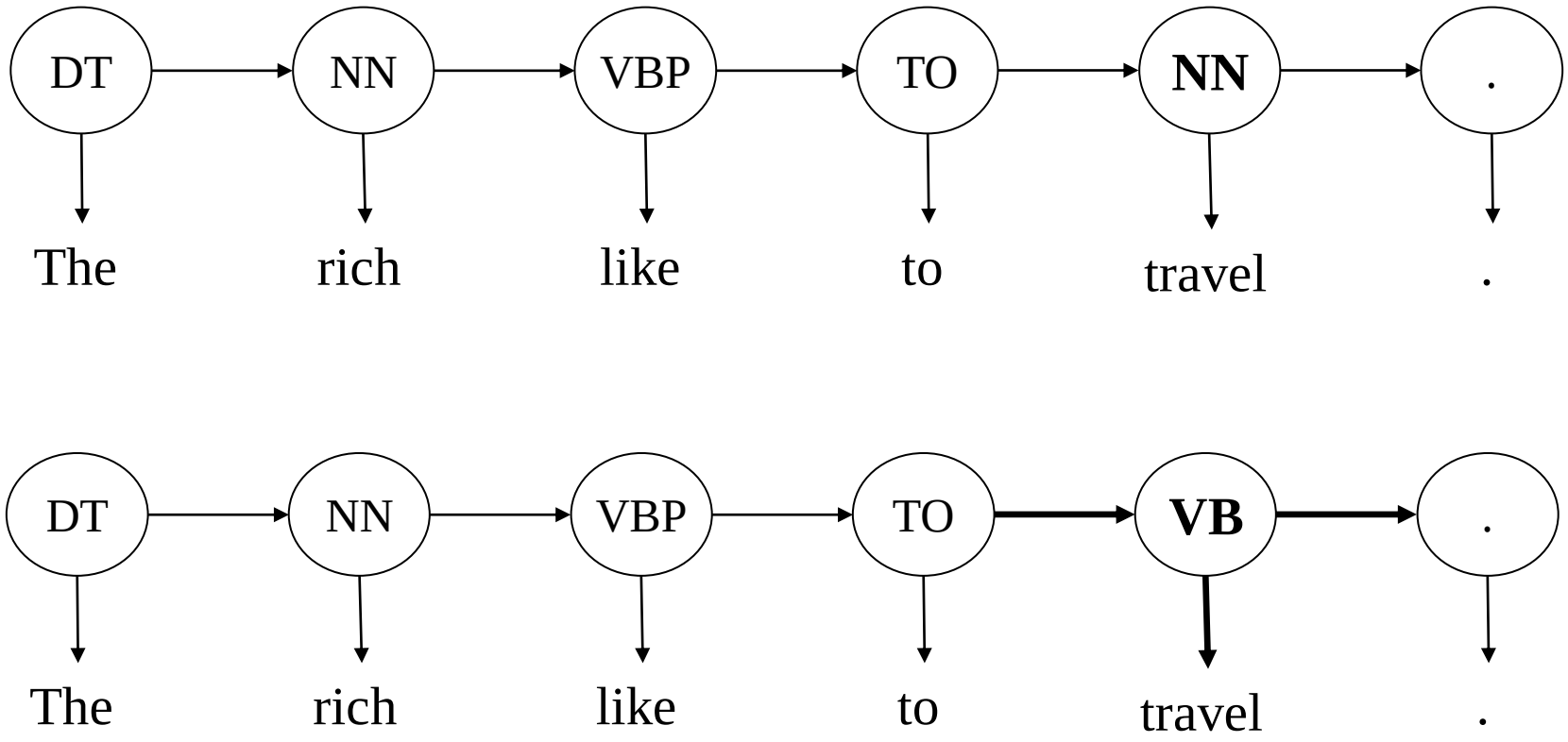
HMM Tagging

- Complete formula
  - $P(T)P(W|T) = \prod P(w_i|w_1t_1\ldots w_{i-1}t_{i-1}t_i)P(t_i|t_1\ldots t_{i-2}t_{i-1})$
- Simplification 1:
  - $P(W|T) = \prod P(w_i|t_i)$
- Simplification 2:
  - $P(T)= \prod P(t_i|t_{i-1})$
- Bigram approximation
  - $T = \text{argmax } P(T|W) = \text{argmax } \prod P(w_i|t_i)\ P(t_i|t_{i-1})$

Example

- The/DT rich/JJ like/VBP to/TO travel/VB ./.

Example


DT
NN
VBP
TO
NN
.
The
rich
like
to
travel
.
DT
NN
VBP
TO
VB
.
The
rich
like
to
travel
.

Maximum Likelihood Estimates

- Transition probabilities

  $P(NN|JJ) = C(JJ,NN)/C(JJ) = 22301/89401 = .249$

- Emission probabilities

  $P(this|DT) = C(DT,this)/C(DT) = 7037/103687 = .068$

Evaluating Taggers

- Data set
  - Training set
  - Development set
  - Test set
- Tagging accuracy
  - how many tags right

HMM POS Results

- Assigning each word its most likely tag: 90%
- Trigram HMM: 95% (55% on unknown words)
- Tuned HMM (Brants 1998): 96.2% (86.0%)
- SOTA (Bi-LSTM CRF): 97.5% (89+%)

Numbers thanks to Dan Klein and Greg Durrett

Remaining Errors

- Words not seen with that tag in training: 4.5%
- Unknown word: 4.5%
- Could get right: 16% (needs parsing)
- Difficult decision: 20% (“set” = VBP or VBD?)
- Underspecified/unclear, gold standard inconsistent/wrong: 58% (e.g., is “discontinued” JJ or VBN)

[Manning 2011]

Confusion Matrix

	JJ	NN	NNP	NNPS	RB	RP	IN	VB	VBD	VBN	VBP	Total
JJ	0	**177**	**56**	0	**61**	2	5	10	15	**108**	0	488
NN	**244**	0	**103**	0	12	1	1	29	5	6	19	525
NNP	**107**	**106**	0	**132**	5	0	7	5	1	2	0	427
NNPS	1	0	**110**	0	0	0	0	0	0	0	0	142
RB	**72**	21	7	0	0	16	**138**	1	0	0	0	295
RP	0	0	0	0	**39**	0	**65**	0	0	0	0	104
IN	11	0	1	0	**169**	**103**	0	1	0	0	0	323
VB	17	**64**	9	0	2	0	1	0	4	7	**85**	189
VBD	10	5	3	0	0	0	0	3	0	**143**	2	166
VBN	**101**	3	3	0	0	0	0	3	**108**	0	1	221
VBP	5	34	3	1	1	0	2	**49**	6	3	0	104
Total	626	536	348	144	317	122	279	102	140	269	108	3651

JJ/NN NN
official knowledge

VBD RP/IN DT NN
made up the story

RB VBD/VBN NNS
recently sold shares

[Example from Toutanova+Manning'00 via Dan Klein]

Notes on POS

- New domains
  - Lower performance
- New languages
  - Morphology matters! Also availability of training data
- Distributional clustering
  - Combine statistics about semantically related words
  - Example: names of companies
  - Example: days of the week
  - Example: animals

Brown Clustering

- Words with similar vector representations (embeddings) are clustered together, in an agglomerative (recursive) way
- For example, “Monday”, “Tuesday”, etc. may form a new vector “Day of the week”
- Published by Brown et al. [1992]

Example

Friday Monday Thursday Wednesday Tuesday Saturday Sunday weekends Sundays
people guys folks fellows CEOs chaps doubters commies unfortunates blokes
down backwards ashore sideways southward northward overboard aloft downwards adrift
water gas coal liquid acid sand carbon steam shale iron
great big vast sudden mere sheer gigantic lifelong scant colossal
American Indian European Japanese German African Catholic Israeli Italian Arab
mother wife father son husband brother daughter sister boss uncle
machine device controller processor CPU printer spindle subsystem compiler plotter
John George James Bob Robert Paul William Jim David Mike
feet miles pounds degrees inches barrels tons acres meters bytes
had hadn't hath would've could've should've must've might've
that tha theat
head body hands eyes voice arm seat eye hair mouth

Example

- Input:
  - this is one document . it has two sentences but the program only cares about spaces .
  - here is another document . it also has two sentences .
  - and here is a third document with one sentence .
  - this document is short .
  - the dog ran in the park .
  - the cat was chased by the dog .
  - the dog chased the cat .

[code by Michael Heilman: https://github.com/mheilman/tan-clustering]

.	1011	9
the	011	7
is	110	4
document	1110	4
dog	000	3
it	101001	2
one	11111	2
sentences	1010111	2
chased	00111	2
two	1010100	2
has	1010110	2
here	111101	2
this	1000	2
cat	0010	2
and	11110010	1
sentence	11110011	1
ran	01011	1
in	0100	1
spaces	10101011011	1
another	1010001	1
cares	101010111	1
also	1010000	1
only	10101011010	1
program	10101011001	1
was	001100	1
park	01010	1
but	10101011000	1
short	1001	1
with	111100001	1
by	001101	1
a	111100000	1
about	10101010	1
third	11110001	1

[code by Michael Heilman]

Notes on POS

- British National Corpus
  - http://www.natcorp.ox.ac.uk/
- Tagset sizes
  - PTB 45, Brown 85, Universal 12, Twitter 25
- Dealing with unknown words
  - Look at features like twoDigitNum, allCaps, initCaps, containsDigitAndSlash (Bikel et al. 1999)

HMM Spreadsheet

- Jason Eisner's awesome interactive spreadsheet about learning HMMs
  - http://cs.jhu.edu/~jason/papers/#eisner-2002-tnlp
  - http://cs.jhu.edu/~jason/papers/eisner.hmm.xls

NLP

Introduction to NLP

232.

Information Extraction

Information Extraction

- Usually from unstructured (or semi-structured) data
- Examples
  - News stories
  - Scientific papers
  - Resumes
- Entities
  - Who did what, when, where, why
- Build knowledge base (KBP Task)

Named Entities

- Types:
  - People
  - Locations
  - Organizations
    - Teams, Newspapers, Companies
  - Geo-political entities
- Ambiguity:
  - London can be a person, city, country (by metonymy) etc.
- Useful for interfaces to databases, question answering, etc.

Named Entities

Type	Tag	Sample Categories	Example sentences
People	PER	people, characters	**Turing** is a giant of computer science.
Organization	ORG	companies, sports teams	The **IPCC** warned about the cyclone.
Location	LOC	regions, mountains, seas	The **Mt. Sanitas** loop is in **Sunshine Canyon**.
Geo-Political Entity	GPE	countries, states, provinces	**Palo Alto** is raising the fees for parking.
Facility	FAC	bridges, buildings, airports	Consider the **Tappan Zee Bridge**.
Vehicles	VEH	planes, trains, automobiles	It was a classic **Ford Falcon**.

Figure 21.1 A list of generic named entity types with the kinds of entities they refer to.

Named Entity Recognition (NER)

- Segmentation
  - Which words belong to a named entity?
  - Brazilian football legend Pele's condition has improved, according to a Thursday evening statement from a Sao Paulo hospital.
- Classification
  - What type of named entity is it?
  - Use gazetteers, spelling, adjacent words, etc.
  - Brazilian football legend [$_{\text{PERSON}}$ Pele]'s condition has improved, according to a [$_{\text{TIME}}$ Thursday evening] statement from a [$_{\text{LOCATION}}$ Sao Paulo] hospital.

Times and Events

- Times
  - Absolute expressions
  - Relative expressions (e.g., “last night”)
- Events
  - E.g., a plane went past the end of the runway

NER, Time, and Event extraction

- Brazilian football legend [$_{\text{PERSON}}$ Pele]'s condition has improved, according to a [$_{\text{TIME}}$ Thursday evening] statement from a [$_{\text{LOCATION}}$ Sao Paulo] hospital.
- There had been earlier concerns about Pele's health after [$_{\text{ORG}}$ Albert Einstein Hospital] issued a release that said his condition was "unstable.“
- [$_{\text{TIME}}$ Thursday night]'s release said [$_{\text{EVENT}}$ Pele was relocated] to the intensive care unit because a kidney dialysis machine he needed was in ICU.

Event Extraction

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

Event Extraction

FARE-RAISE ATTEMPT:	LEAD AIRLINE:	UNITED AIRLINES
	AMOUNT:	$6
	EFFECTIVE DATE:	2006-10-26
	FOLLOWER:	AMERICAN AIRLINES

Named Entity Recognition (NER)

Name	Possible Categories
Washington	Person, Location, Political Entity, Organization, Vehicle
Downing St.	Location, Organization
IRA	Person, Organization, Monetary Instrument
Louis Vuitton	Person, Organization, Commercial Product

Figure 21.2 Common categorical ambiguities associated with various proper names.

[$_{PER}$ Washington] was born into slavery on the farm of James Burroughs.
[$_{ORG}$ Washington] went up 2 games to 1 in the four-game series.
Blair arrived in [$_{LOC}$ Washington] for what may well be his last state visit.
In June, [$_{GPE}$ Washington] passed a primary seatbelt law.
The [$_{VEH}$ Washington] had proved to be a leaky ship, every passage I made...

Figure 21.3 Examples of type ambiguities in the use of the name *Washington*.

Sample Input for NER

```
( (S
    (NP-SBJ-1
      (NP (NNP Rudolph) (NNP Agnew) )
      (, ,)
      (UCP
        (ADJP
          (NP (CD 55) (NNS years) )
          (JJ old) )
        (CC and)
        (NP
          (NP (JJ former) (NN chairman) )
          (PP (IN of)
            (NP (NNP Consolidated) (NNP Gold) (NNP Fields) (NNP PLC) ))))
      (, ,) )
    (VP (VBD was)
      (VP (VBN named)
        (S
          (NP-SBJ (-NONE- *-1) )
          (NP-PRD
            (NP (DT a) (JJ nonexecutive) (NN director) )
            (PP (IN of)
              (NP (DT this) (JJ British) (JJ industrial) (NN conglomerate) ))))))
    (. .) ))
```

Sample Output for NER (IOB format)

file_id	sent_id	word_id	iob_inner	pos	word
0002	**1**	**0**	**B-PER**	**NNP**	**Rudolph**
0002	**1**	**1**	**I-PER**	**NNP**	**Agnew**
0002	1	2	O	COMMA	COMMA
0002	1	3	B-NP	CD	55
0002	1	4	I-NP	NNS	years
0002	1	5	B-ADJP	JJ	old
0002	1	6	O	CC	and
0002	1	7	B-NP	JJ	former
0002	1	8	I-NP	NN	chairman
0002	1	9	B-PP	IN	of
0002	**1**	**10**	**B-ORG**	**NNP**	**Consolidated**
0002	**1**	**11**	**I-ORG**	**NNP**	**Gold**
0002	**1**	**12**	**I-ORG**	**NNP**	**Fields**
0002	**1**	**13**	**I-ORG**	**NNP**	**PLC**
0002	1	14	O	COMMA	COMMA
0002	1	15	B-VP	VBD	was
0002	1	16	I-VP	VBN	named
0002	1	17	B-NP	DT	a
0002	1	18	I-NP	JJ	nonexecutive
0002	1	19	I-NP	NN	director
0002	1	20	B-PP	IN	of
0002	1	21	B-NP	DT	this
0002	1	22	I-NP	JJ	British
0002	1	23	I-NP	JJ	industrial
0002	1	24	I-NP	NN	conglomerate
0002	1	25	O	.	.

NER Demos

- http://nlp.stanford.edu:8080/ner/
- http://cogcomp.org/page/demo_view/ner
- http://demo.allennlp.org/named-entity-recognition

NER Extraction Features

identity of w_i
identity of neighboring words
part of speech of w_i
part of speech of neighboring words
base-phrase syntactic chunk label of w_i and neighboring words
presence of w_i in a **gazetteer**
w_i contains a particular prefix (from all prefixes of length ≤ 4)
w_i contains a particular suffix (from all suffixes of length ≤ 4)
w_i is all upper case
word shape of w_i
word shape of neighboring words
short word shape of w_i
short word shape of neighboring words
presence of hyphen

Figure 21.5 Features commonly used in training named entity recognition systems.

NER Extraction Features

$$\text{prefix}(w_i) = \texttt{L}$$
$$\text{prefix}(w_i) = \texttt{L'}$$
$$\text{prefix}(w_i) = \texttt{L'O}$$
$$\text{prefix}(w_i) = \texttt{L'Oc}$$
$$\text{suffix}(w_i) = \texttt{tane}$$
$$\text{suffix}(w_i) = \texttt{ane}$$
$$\text{suffix}(w_i) = \texttt{ne}$$
$$\text{suffix}(w_i) = \texttt{e}$$
$$\text{word-shape}(w_i) = \texttt{X'Xxxxxxxx}$$
$$\text{short-word-shape}(w_i) = \texttt{X'Xx}$$

Feature Encoding in NER

Word	POS	Chunk	Short shape	Label
American	NNP	B-NP	Xx	B-ORG
Airlines	NNPS	I-NP	Xx	I-ORG
,	,	O	,	O
a	DT	B-NP	x	O
unit	NN	I-NP	x	O
of	IN	B-PP	x	O
AMR	NNP	B-NP	X	B-ORG
Corp.	NNP	I-NP	Xx.	I-ORG
,	,	O	,	O
immediately	RB	B-ADVP	x	O
matched	VBD	B-VP	x	O
the	DT	B-NP	x	O
move	NN	I-NP	x	O
,	,	O	,	O
spokesman	NN	B-NP	x	O
Tim	NNP	I-NP	Xx	B-PER
Wagner	NNP	I-NP	Xx	I-PER
said	VBD	B-VP	x	O
.	,	O	.	O

Figure 21.6 Word-by-word feature encoding for NER.

NER as Sequence Labeling

- Many NLP problems can be cast as sequence labeling problems
  - POS – part of speech tagging
  - NER – named entity recognition
  - SRL – semantic role labeling
- Input
  - Sequence $w_1 w_2 \ldots w_n$
- Output
  - Labeled words
- Classification methods
  - Can use the categories of the previous tokens as features in classifying the next one
  - Direction matters

NER as Sequence Labeling

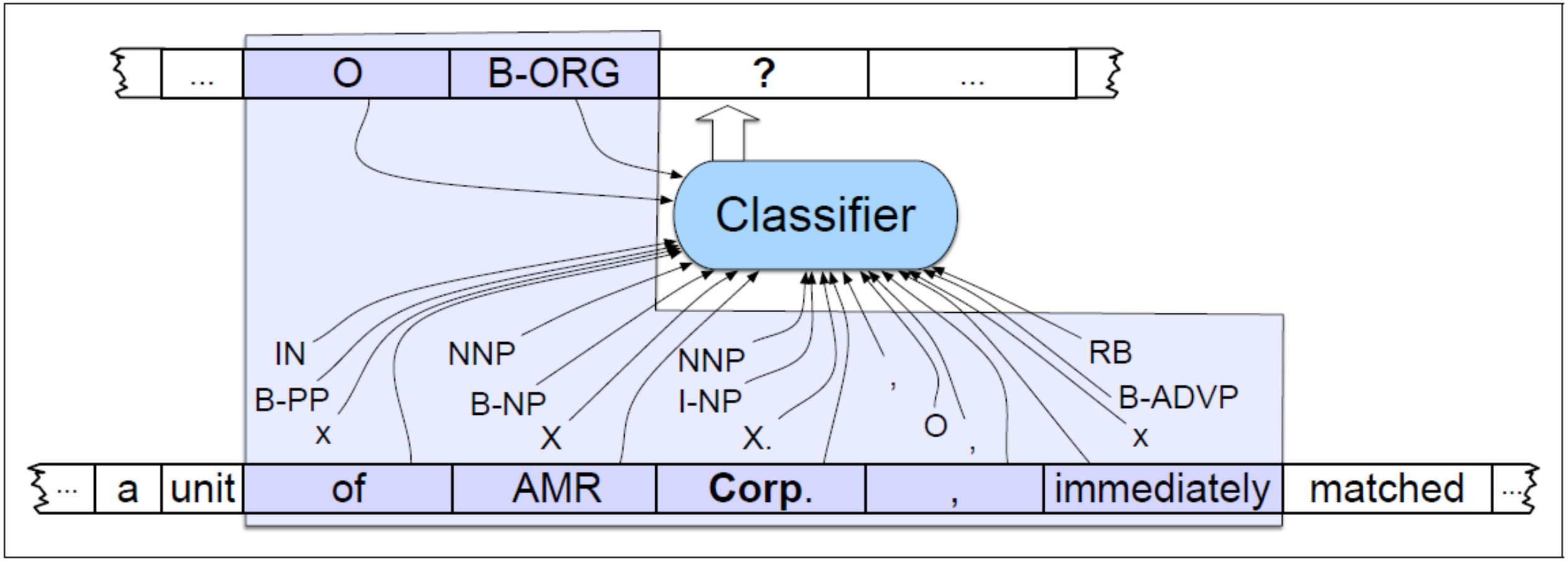


Figure 21.7 Named entity recognition as sequence labeling. The features available to the classifier during training and classification are those in the boxed area.

Temporal Expressions

Absolute	Relative	Durations
April 24, 1916	yesterday	four hours
The summer of '77	next semester	three weeks
10:15 AM	two weeks from yesterday	six days
The 3rd quarter of 2006	last quarter	the last three quarters

Figure 21.17 Examples of absolute, relational and durational temporal expressions.

Temporal Lexical Triggers

Category	Examples
Noun	*morning, noon, night, winter, dusk, dawn*
Proper Noun	*January, Monday, Ides, Easter, Rosh Hashana, Ramadan, Tet*
Adjective	*recent, past, annual, former*
Adverb	*hourly, daily, monthly, yearly*

Figure 21.18 Examples of temporal lexical triggers.

TempEx Example

```
# yesterday/today/tomorrow
$string =~ s/(($OT+(early|earlier|later?)$CT+\s+)?(($OT+the$CT+\s+)?$OT+day$CT+\s+
$OT+(before|after)$CT+\s+)?$OT+$TERelDayExpr$CT+(\s+$OT+(morning|afternoon|
evening|night)$CT+)?)/<TIMEX2 TYPE=\"DATE\">$1<\/TIMEX2>/gio;

$string =~ s/($OT+\w+$CT+\s+)
<TIMEX2 TYPE=\"DATE\"[^>]*>($OT+(Today|Tonight)$CT+)<\/TIMEX2>/$1$2/gso;

# this/that (morning/afternoon/evening/night)
$string =~ s/(($OT+(early|earlier|later?)$CT+\s+)?$OT+(this|that|every|the$CT+\s+
$OT+(next|previous|following))$CT+\s*$OT+(morning|afternoon|evening|night)
$CT+(\s+$OT+thereafter$CT+)?)/<TIMEX2 TYPE=\"DATE\">$1<\/TIMEX2>/gosi;
```

Figure 21.19 Fragment of Perl code from MITRE's TempEx temporal tagging system.

TimeML

```
<TIMEX3 id=''t1'' type="DATE" value="2007-07-02" functionInDocument="CREATION_TIME">
July 2, 2007 </TIMEX3>  A fare increase initiated <TIMEX3 id="t2" type="DATE"
value="2007-W26"  anchorTimeID="t1">last week</TIMEX3> by UAL Corp's United  Airlines
was matched by competitors over <TIMEX3 id="t3" type="DURATION" value="P1WE"
anchorTimeID="t1"> the weekend </TIMEX3>, marking the second successful fare increase
in <TIMEX3 id="t4" type="DURATION" value="P2W" anchorTimeID="t1"> two weeks </TIMEX3>.
```

Figure 21.21 TimeML markup including normalized values for temporal expressions.

TimeBank

```
<TIMEX3 tid="t57" type="DATE" value="1989-10-26"  functionInDocument="CREATION_TIME">
10/26/89  </TIMEX3>

Delta Air Lines earnings <EVENT eid="e1" class="OCCURRENCE"> soared </EVENT> 33% to a
record in  <TIMEX3 tid="t58" type="DATE" value="1989-Q1" anchorTimeID="t57"> the
fiscal first quarter </TIMEX3>, <EVENT eid="e3"  class="OCCURRENCE">bucking</EVENT>
the industry trend toward <EVENT eid="e4" class="OCCURRENCE">declining</EVENT>
profits.
```

Figure 21.25 Example from the TimeBank corpus.

The Message Understanding Conference (MUC)

- Slot Filling

Bridgestone Sports Co. said Friday it has set up a joint venture in Taiwan with a local concern and a Japanese trading house to produce golf clubs to be shipped to Japan. The joint venture, Bridgestone Sports Taiwan Co., capitalized at 20 million new Taiwan dollars, will start production in January 1990 with production of 20,000 iron and "metal wood" clubs a month.

MUC Example

Tie-up-1		**Activity-1**:	
RELATIONSHIP	tie-up	COMPANY	Bridgestone Sports Taiwan Co.
ENTITIES	Bridgestone Sports Co.	PRODUCT	iron and "metal wood" clubs
	a local concern	START DATE	DURING: January 1990
	a Japanese trading house		
JOINT VENTURE	Bridgestone Sports Taiwan Co.		
ACTIVITY	Activity-1		
AMOUNT	NT$20000000		

Figure 21.26 The templates produced by FASTUS given the input text on page 25.

Biomedical example

- Gene labeling
- Sentence:
  - [$_{GENE}$ BRCA1] and [$_{GENE}$ BRCA2] are human genes that produce tumor suppressor proteins

Other Examples

- Job announcements
  - Location, title, starting date, qualifications, salary
- Seminar announcements
  - Time, title, location, speaker
- Medical papers
  - Drug, disease, gene/protein, cell line, species, substance

Filling the Templates

- Some fields get filled by text from the document
  - E.g., the names of people
- Others can be pre-defined values
  - E.g., successful/unsuccessful merger
- Some fields allow for multiple values

Evaluating Template-Based NER

- For each test document
  - Number of correct template extractions
  - Number of slot/value pairs extracted
  - Number of extracted slot/value pairs that are correct

NLP

Introduction to NLP

233.

Relation Extraction

Relation Extraction

- Person-person
  - ParentOf, MarriedTo, Manages
- Person-organization
  - WorksFor
- Organization-organization
  - IsPartOf
- Organization-location
  - IsHeadquarteredAt

Relation Extraction

- Core NLP task
  - Used for building knowledge bases, question answering
- Input
  - **Mazda North American Operations** *is headquartered in* **Irvine, Calif.**, and oversees the sales, marketing, parts and customer service support of Mazda vehicles in the United States and Mexico through nearly 700 dealers.
- Output (predicate)
  - IsHeadquarteredIn (Mazda North American Operations, Irvine)

Relation extraction

- Using patterns
  - Regular expressions
  - Gazetteers
- Supervised learning
- Semi-supervised learning
  - Using seeds

Relation Extraction

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

The ACE Evaluation

- Newspaper data
- Entities:
  - Person, Organization, Facility, Location, Geopolitical Entity
- Relations:
  - Role, Part, Located, Near, Social

The ACE Evaluation

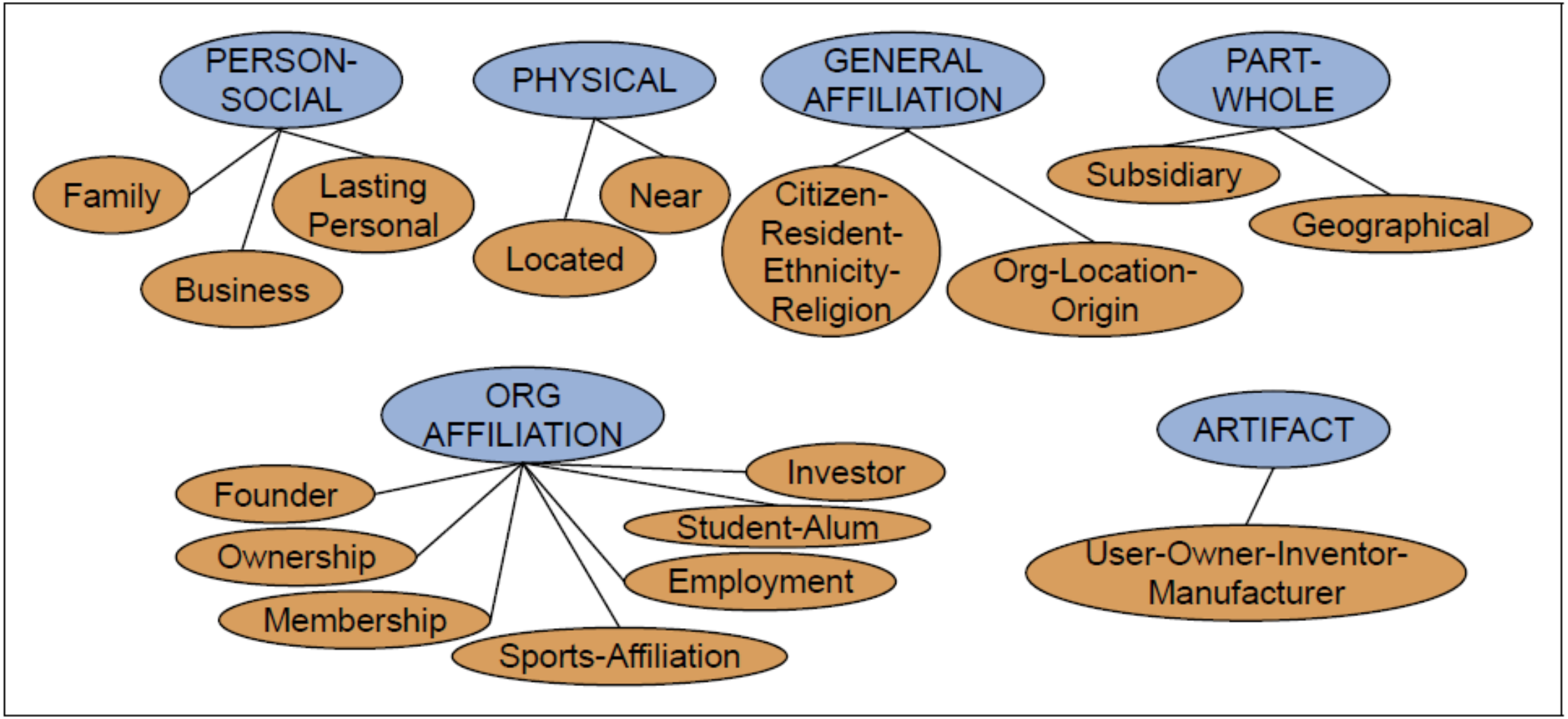


Figure 21.8 The 17 relations used in the ACE relation extraction task.

Semantic Relations

Relations	Types	Examples
Physical-Located	PER-GPE	**He** was in **Tennessee**
Part-Whole-Subsidiary	ORG-ORG	**XYZ**, the parent company of **ABC**
Person-Social-Family	PER-PER	**Yoko**'s husband **John**
Org-AFF-Founder	PER-ORG	**Steve Jobs**, co-founder of **Apple**...

Figure 21.9 Semantic relations with examples and the named entity types they involve.

Extracting IS-A Relations

- Hearst's patterns
  - X and other Y
  - X or other Y
  - Y such as X
  - Y, including X
  - Y, especially X
- Example
  - Evolutionary relationships between the platypus and other mammals

Hypernym Extraction (Hearst)

NP {, NP}* {,} (and|or) other NP_H	temples, treasuries, and other important civic buildings
NP_H such as {NP,}* {(or|and)} NP	red algae such as Gelidium
such NP_H as {NP,}* {(or|and)} NP	such authors as Herrick, Goldsmith, and Shakespeare
NP_H {,} including {NP,}* {(or|and)} NP	common-law countries, including Canada and England
NP_H {,} especially {NP}* {(or|and)} NP	European countries, especially France, England, and Spain

Figure 21.11 Hand-built lexico-syntactic patterns for finding hypernyms, using {} to mark optionality (Hearst, 1992a, 1998).

Supervised Relation Extraction

- Look for sentences that have two entities that we know are part of the target relation
- Look at the other words in the sentence, especially the ones between the two entities
- Use a classifier to determine whether the relation exists

Semi-supervised Relation Extraction

- Start with some seeds, e.g.,
  - **Beethoven** *was born* in December **1770** in Bonn
- Look for other sentences with the same words
- Look for expressions that appear nearby
- Look for other sentences with the same expressions

Bootstrapping

function BOOTSTRAP(*Relation R*) **returns** *new relation tuples*

tuples ← Gather a set of seed tuples that have relation *R*
iterate
sentences ← find sentences that contain entities in *seeds*
patterns ← generalize the context between and around entities in *sentences*
newpairs ← use *patterns* to grep for more tuples
newpairs ← *newpairs* with high confidence
tuples ← *tuples* + *newpairs*
return tuples

Figure 21.14 Bootstrapping from seed entity pairs to learn relations.

Bootstrapping

(21.6) Budget airline Ryanair, which uses Charleroi as a hub, scrapped all weekend flights out of the airport.

(21.7) All flights in and out of Ryanair's Belgian hub at Charleroi airport were grounded on Friday...

(21.8) A spokesman at Charleroi, a main hub for Ryanair, estimated that 8000 passengers had already been affected.

Bootstrapping

```
/ [ORG], which uses [LOC] as a hub /
/ [ORG]'s hub at [LOC] /
/ [LOC] a main hub for [ORG] /
```

Evaluating Relation Extraction

- Precision P
  - correctly extracted relations/all extracted relations
- Recall R
  - correctly extracted relations/all existing relations
- F1 measure
  - F1 = 2PR/(P+R)
- If there is no annotated data
  - only measure precision

NLP