More Probabilistic Models

Introduction to
Artificial Intelligence
COS302
Michael L. Littman
Fall 2001

Administration

2/3, 1/3 split for exams
Last HW due Wednesday
Wrap up Wednesday
Sample exam questions later…
Example analogies, share, etc.

Topics

Goal: Try to practice what we know about probabilistic models

- Segmentation: most likely sequence of words
- EM for segmentation
- Belief net representation
- EM for learning probabilities

Segmentation

Add spaces:

bothearthandsaturnspin

Applications:

- no spaces in speech
- no spaces in Chinese
- postscript or OCR to text

So Many Choices…

Bothearthandsaturnspin.
B O T H E A R T H A N D S A T U R N S P I N.
Bo-the-art hands at Urn's Pin.
Bot heart? Ha! N D S a turns pi N.
Both Earth and Saturn spin.
…so little time. How to choose?

Probabilistic Approach

Standard spiel:

1. Choose a generative model
2. Estimate parameters
3. Find most likely sequence

Generative Model

Choices:

- $\boxed{\text{unigram Pr(w)}}$
- bigram Pr(w|w')
- trigram Pr(w|w',w")
- tag-based HMM Pr(t|t',t"), Pr(w|t)
- probabilistic context-free grammar Pr(X Y|Z), Pr(w|Z)

Estimate Parameters

For English, can count word frequencies in text sample:

$\Pr(w) = \text{count}(w)/\text{sum}_w \text{count}(w)$

For Chinese, could get someone to segment, or use EM (next).

Search Algorithm

gotothestore

Compute the maximum probability sequence of words.

$p_0 = 1$

$p_j = \max_{i<j} p_{j-i} \Pr(w_{i:j})$

$p_5 = \max(p_0 \Pr(\text{gotot}), p_1 \Pr(\text{otot}),$
$p_2 \Pr(\text{tot}), p_3 \Pr(\text{ot}), p_4 \Pr(\text{t}))$

Get to point i, use one word to get to j.

Unigrams Probs via EM

g 0.01	go 0.78	got 0.21	goto 0.61
o 0.02			
t 0.04	to 0.76	tot 0.74	
o 0.02			
t 0.04	the 0.83	thes 0.04	
h 0.03	he 0.22	hes 0.16	hest 0.19
e 0.05	es 0.09		
s 0.04	store 0.81		
t 0.04	to 0.70	tore 0.07	
o 0.02	or 0.65	ore 0.09	
r 0.01	re 0.12	e 0.05	

EM for Segmentation

Pick unigram probabilities

Repeat until probability doesn't improve much

1. Fractionally label (like forward-backward)
2. Use fractional counts to reestimate unigram probabilities

Probability Distribution

Represent probability distribution on a bit sequence.

A B Pr(AB)

0 0 .06

0 1 .24

1 0 .14

1 1 .56

Conditional Probs.

Pr(A|~B) = .14/(.14+.06) = .7
Pr(A|B) = .56/(.56+.24) = .7
Pr(B|~A) = .24/(.24+.06) = .8
Pr(B|A) = .56/(.56+.14) = .8

So, Pr(AB)=Pr(A)Pr(B)

Graphical Model

Pick a value for A.
Pick a value for B.
Independent influence: kind of and/or-ish.

Probability Distribution

A	B	Pr(AB)
0	0	.08
0	1	.42
1	0	.32
1	1	.18

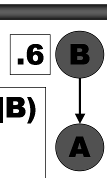
Dependent influence: kind of xor-ish.

Conditional Probs.

Pr(A|~B) = .32/(.32+.08) = .8
Pr(A|B) = .18/(.18+.42) = .3
Pr(B|~A) = .42/(.42+.08) = .84
Pr(B|A) = .18/(.18+.32) = .36

So, a bit more complex.

Graphical Model



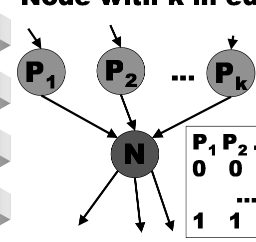
B	Pr(A|B)
0	.8
1	.3

CPT: Conditional Probability Table

Pick a value for B.
Pick a value for A, based on B.

General Form

Acyclic graph; each node a var.
Node with k in edges; size 2^k CPT.



P_1	P_2 ...	P_k	$Pr(N|P_1\ P_2 \ldots P_k)$
0	0	0	$p_{00\ldots0}$
	...		
1	1	1	$p_{11\ldots1}$

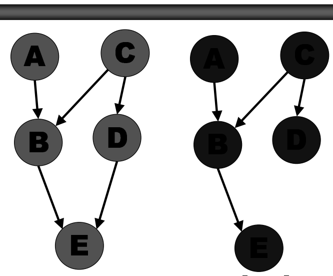
Belief Network

- Bayesian network, Bayes net, etc.
- Represents a prob. distribution over 2^n values with $O(2^k)$ entries, where k is the largest indegree
- Can be applied to variables with values beyond just {0, 1}. Kind of like a CSP.

What Can You Do?

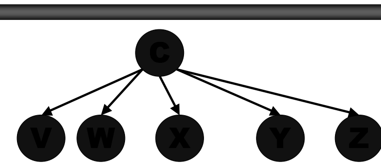
- Belief net inference: $Pr(N|E_1,\sim E_2,E_3, \ldots)$.
- Polytime algorithms exist if undirected version of DAG is acyclic (singly connected)
- NP-hard if multiply connected.

Example BNs



multiply singly

Popular BN

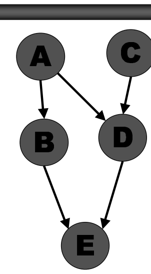


Recognize this?

BN Applications

- Diagnosing diseases
- Decoding noisy messages from deep space probes
- Reasoning about genetics
- Understanding consumer purchasing patterns
- Annoying users of Windows

Parameter Learning

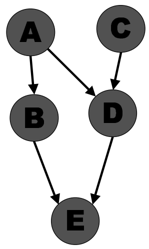


A	B	C	D	E
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	0	1	1	0
0	0	1	1	1

Pr(B|~A)?

1/5

Hidden Variable



A	B	C	D	E
	0		0	1
	0		1	1
	1		0	1
	1		0	1
	0		0	1
	0		1	0
	0		1	1

Pr(B|~A)?

What to Learn

- Segmentation problem
- Algorithm for finding the most likely segmentation
- How EM might be used for parameter learning
- Belief network representation
- How EM might be used for parameter learning