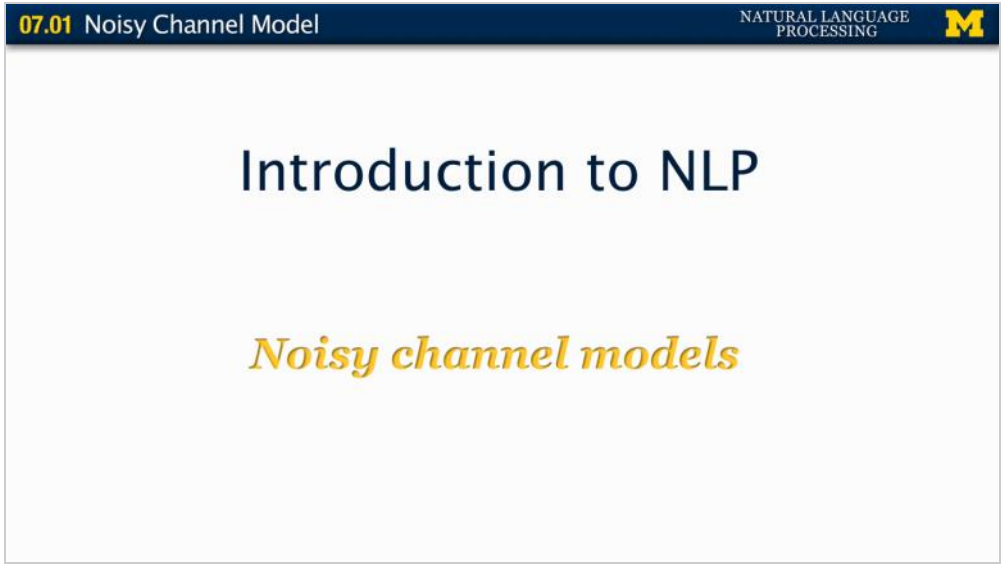
Week 7

Tuesday, July 26, 2016 17:26

7.1



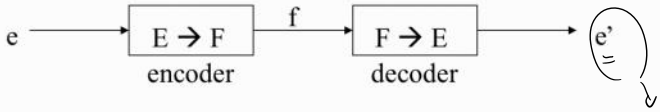
The Noisy Channel Model

- Example:
  - Input: Written English (X)
  - Encoder: garbles the input (X->Y)
  - Output: Spoken English (Y)
- More examples:
  - Grammatical English to English with mistakes
  - English to bitmaps (characters)
- P(X,Y) = P(X)P(Y|X)

Written → Spoken English

Encoding and Decoding

- Given f, guess e



$$e' = \arg\max_e P(e|f) = \arg\max_e P(f|e)\,P(e)$$

translation model language model

foreign language → guess English

what prob. of English sentence given foreign sentence

Example

- Translate "la maison blanche"

	P(f|e)	P(e)
cat plays piano	-	-
house white the	+	-
the house white	+	-
the red house	-	+
the small cat	-	+
the white house	+	+

valid English sequence → doesn't seem a good translation

the good translation

Uses of the Noisy Channel Model

- Handwriting recognition
- Text generation
- Text summarization
- Machine translation
- Spelling correction
  - See separate lecture on text similarity and edit distance

Spelling Correction

w	c	w | c	P(w | c)	P(c)	10^9 P(w | c) P(c)
thew	the	ew | e	.000007	.02	144.
thew	thew		.95	.00000009	90.
thew	thaw	e | a	.001	.0000007	0.7
thew	threw	h | hr	.000008	.000004	0.03
thew	thwe	ew | we	.000003	.00000004	0.0001

From Peter Norvig: http://**norvig**.com/ngrams/ch14.pdf

← n gram
↓ spelling correction

NLP

7.2

NLP

Introduction to NLP

Part of speech tagging

The POS Task

- Example
  - Bahrainis vote in second round of parliamentary election
- Jabberwocky (by Lewis Carroll, 1872)

`Twas brillig, and the slithy toves
Did gyre and gimble in the wabe:
All mimsy were the borogoves,
And the mome raths outgrabe.

get mopholagical info - even not real words

mimsy: adj. borogoves: n. follow "the"

Parts of Speech

- Open class:
  - nouns, non-modal verbs, adjectives, adverbs
- Closed class:
  - prepositions, modal verbs, conjunctions, particles, determiners, pronouns

Add new words any time

hardly adding new words

Penn Treebank tagset (1/2)

Penn Treebank

Tag	Description	Example
CC	coordinating conjunction	and
CD	cardinal number	1
DT	determiner	the
EX	existential there	*there* is
FW	foreign word	d'oeuvre
IN	preposition/subordinating conjunction	in, of, like
JJ	adjective	green
JJR	adjective, comparative	greener
JJS	adjective, superlative	greenest
LS	list marker	1)
MD	modal	could, will
NN	noun, singular or mass	table
NNS	noun plural	tables
NNP	proper noun, singular	John
NNPS	proper noun, plural	Vikings
PDT	predeterminer	*both* the boys
POS	possessive ending	friend*'s*

Penn Treebank tagset (2/2)

Tag	Description	Example
PRP	personal pronoun	I, he, it
PRP$	possessive pronoun	my, his
RB	adverb	however, usually, naturally, here, good
RBR	adverb, comparative	better
RBS	adverb, superlative	best
RP	particle	give *up*
TO	to	*to* go, *to* him
UH	interjection	uhhuhhuhh
VB	verb, base form	take
VBD	verb, past tense	took
VBG	verb, gerund/present participle	taking
VBN	verb, past participle	taken
VBP	verb, sing. present, non-3d	take
VBZ	verb, 3rd person sing. present	takes
WDT	wh-determiner	which
WP	wh-pronoun	who, what
WP$	possessive wh-pronoun	whose
WRB	wh-abverb	where, when

Some Observations

- Ambiguity
  - count (noun) vs. count (verb)
  - 11% of all types but 40% of all tokens in the Brown corpus are ambiguous.
  - Examples
    - *like* can be tagged as ADP VERB ADJ ADV NOUN
    - *present* can be tagged as ADJ NOUN VERB ADV

types / token — ambiguous types are frequent.

Some Observations

- More examples:
  - transport, object, discount, address
  - content
- French pronunciation:
  - est président, fils
- Three main techniques:
  - rule-based
  - machine learning (e.g., conditional random fields, maximum entropy Markov models)
  - transformation-based
- Useful for parsing, translation, text to speech, word sense disambiguation, etc

V.n V.n

adj. n.

Words pronounced differently based on PoS.

CRF

MaxEnt Markov model

★ Machine Learning transformation-based) easier to use in a larger scale because they train automatically

Example

- Bethlehem/NNP Steel/NNP Corp./NNP ,/, hammered/VBN by/IN higher/JJR **costs/NNS**
- Bethlehem/NNP Steel/NNP Corp./NNP ,/, hammered/VBN by/IN higher/JJR **costs/VBZ**

How to choose? Uni-gram – prob. the word is tagged in certain way

Sources of Information

Sources of Information

- Bethlehem/NNP Steel/NNP Corp./NNP ,/, hammered/VBN by/IN higher/JJR **costs/NNS**
- Bethlehem/NNP Steel/NNP Corp./NNP ,/, hammered/VBN by/IN higher/JJR **costs/VBZ**
- Knowledge about individual words
  - lexical information
  - spelling (-or)
  - capitalization (IBM)
- Knowledge about neighboring words

Evaluation

- Baseline
  - tag each word with its most likely tag
  - tag each OOV word as a noun.
  - around 90%
- Current accuracy
  - around 97% for English
  - compared to 98% human performance

Rule-based POS tagging

- Use dictionary or finite-state transducers to find all possible parts of speech
- Use disambiguation rules
  - e.g., ART+V
- Hundreds of constraints can be designed manually

Example in French

Ambiguous

can be pronoun or article.

<S>	^	beginning of sentence
La	rf b nms u	article
teneur	nfs nms	noun feminine singular
moyenne	jfs nfs v1s v2s v3s	adjective feminine singular
en	p a b	preposition
uranium	nms	noun masculine singular
des	p r	preposition
rivières	nfp	noun feminine plural
,	x	punctuation
bien_que	cs	subordinating conjunction
délicate	jfs	adjective feminine singular
à	p	preposition
calculer	v	verb

Sample Rules

- **BS3 BI1**
  - A BS3 (3rd person subject personal pronoun) cannot be followed by a BI1 (1st person indirect personal pronoun).
  - In the example: "il nous faut" (= "we need") – "il" has the tag BS3MS and "nous" has the tags [BD1P BI1P BJ1P BR1P BS1P].
  - The negative constraint "BS3 BI1" rules out "BI1P", and thus leaves only 4 alternatives for the word "nous".
- **N K**
  - The tag N (noun) cannot be followed by a tag K (interrogative pronoun); an example in the test corpus would be: "... fleuve qui ..." (...river, that...).
  - Since "qui" can be tagged both as an ``E" (relative pronoun) and a "K" (interrogative pronoun), the "E" will be chosen by the tagger since an interrogative pronoun cannot follow a noun ("N").
- **R V**
  - A word tagged with R (article) cannot be followed by a word tagged with V (verb): for example "l' appelle" (calls him/her).
  - The word "appelle" can only be a verb, but "l'"" can be either an article or a personal pronoun.
  - Thus, the rule will eliminate the article tag, giving preference to the pronoun.

But the rule-based method is not very popular now. HMM, transformation-based methods are popular.

NLP

NLP

Introduction to NLP

Hidden Markov Models

Markov Models

Markov model (visible)

- Sequence of random variables that aren't independent
- Examples
  - weather reports
  - text

a sequence, whose elements are not independent

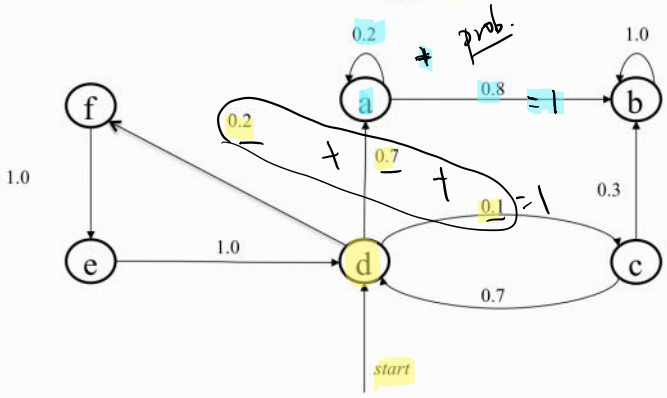
Properties

- Limited horizon:
  $P(X_{t+1} = s_k|X_1,\ldots,X_t) = P(X_{t+1} = s_k|X_t)$
- Time invariant (stationary)
  $= P(X_2=s_k|X_1)$
- Definition: in terms of a transition matrix A and initial state probabilities Π.

depend on the neighbour

← prob. of seeing a state should not depend on time

Example



prob.

$\sum Prob_{out} = 1$ node

Visible MM

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

general formula for joint distribution.

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

apply the Markov Rule

use a bigram model.

$= \pi_{X_1} \prod_{t=1}^{T-1} a_{X_t X_{t+1}}$

first state

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

2nd state

3rd state

$= 1.0 \times 0.7 \times 0.8$

$= 0.56$

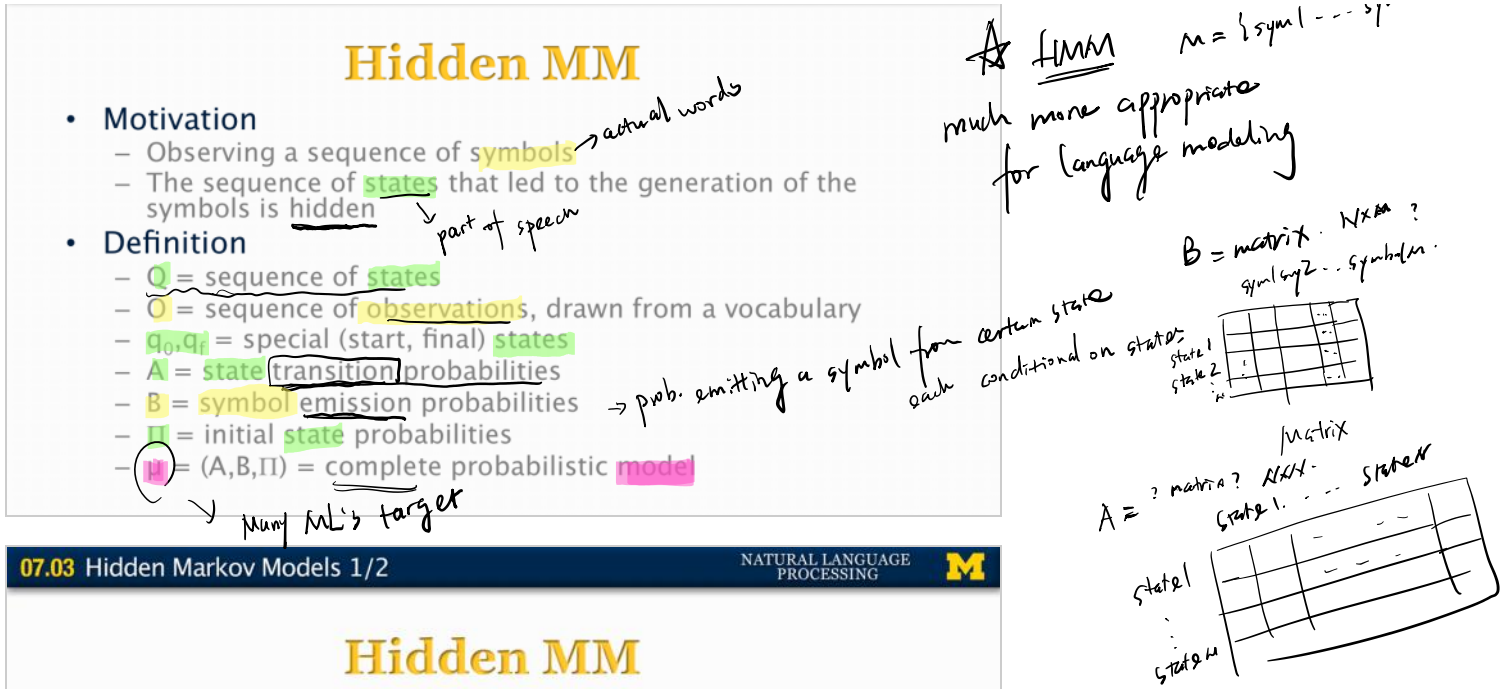
$Pr(d \to a \to b)$

Hidden MM

HMM

Hidden MM

- Motivation
  - Observing a sequence of symbols
  - The sequence of states that led to the generation of the symbols is hidden
- Definition
  - Q = sequence of states
  - O = sequence of 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

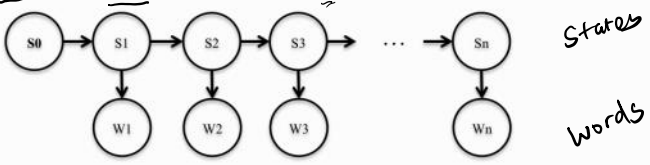


Hidden MM

- Uses
  - part of speech tagging
  - speech recognition
  - gene sequencing

Hidden Markov Model (HMM)

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

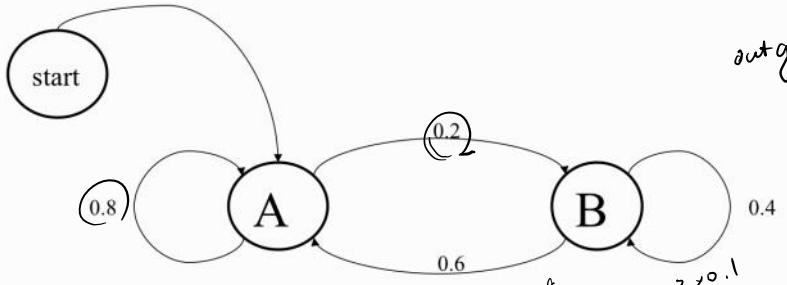


Generative Algorithm

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

↘ matrix of emission probs

State Transition Probabilities



outgoing prob add up to 1.

$P(yz|AA) = \frac{P(yz, AA)}{P(AA)} = \frac{0.8 \times 0.8 \times 0.2 \times 0.1}{0.8}$

$P(yz, AB)$

$P(yz|AB) =$

Emission Probabilities

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

symbol

state	x	y	z
A	0.7	0.2	0.1
B	0.3	0.5	0.2

in state A, more likely to produce x.

in state B, more likely to produce y

All Parameters of the Model

All Parameters of the Model

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

Observation Sequence "yz"

- Starting in state A, P(yz) = ?
- Possible sequences of states:
  - AA
  - AB
  - BA
  - BB
- P(yz) = P(yz|AA) + P(yz|AB) + P(yz|BA) + P(yz|BB) =
  = .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

07.03 Hidden Markov Models 1/2 NATURAL LANGUAGE PROCESSING

States and Transitions

- 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 transition probabilities

Emissions

- Estimating the emission probabilities
  - Harder than transition probabilities
  - 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)

smoothing 消灭 prob. =0

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

$P(S|\mu)$ → model μ, sequence

Prob. the sequence is generated by the model

Tasks With HMM

- Tasks
  - Given $\mu = (A,B,\Pi)$, find $P(O|\mu)$ → observation, model
  - Given O, μ, what is $(X_1,\ldots X_{T+1})$ → sequence of states
  - Given O and a space of all possible μ, find model that best describes the observations
- Decoding – most likely sequence
  - tag each token with a label
- Observation likelihood
  - classify sequences
- Learning
  - train models to fit empirical data

tag → classify, train → learn

Inference

Inference

- Find the most likely sequence of tags, given the sequence of words
  - $t^* = \text{argmax}_t P(t|w)$ (t: tag)
- 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
- Possible solution:
  - Use beam search (partial hypotheses) — beam search; just partial hypotheses, e.g. top 10
  - At each state, only keep the k best hypotheses so far — 一路扔掉 low hypotheses
  - May not work → may show low prob. in the first stage but then turn out to be the best towards the end of sentence

07.03 Hidden Markov Models 1/2 NATURAL LANGUAGE PROCESSING

Viterbi Algorithm

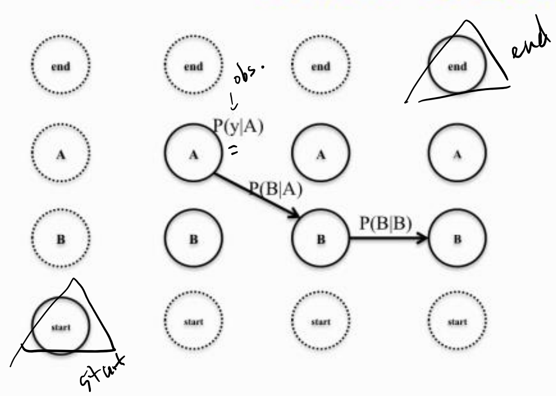
one of the most fundamental algorithm

- Find the best path up to observation i and state s — best up to some point
- Characteristics
  - Uses dynamic programming
  - Memoization → store prob. of some prob. up to some point
  - Backpointers

Also used in CKY.

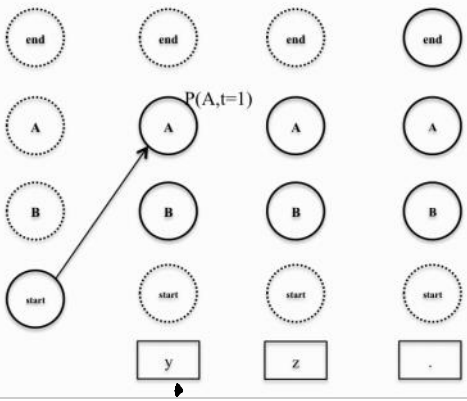
07.03 Hidden Markov Models 1/2 NATURAL LANGUAGE PROCESSING

HMM Trellis



Row: different states of POS

HMM Trellis

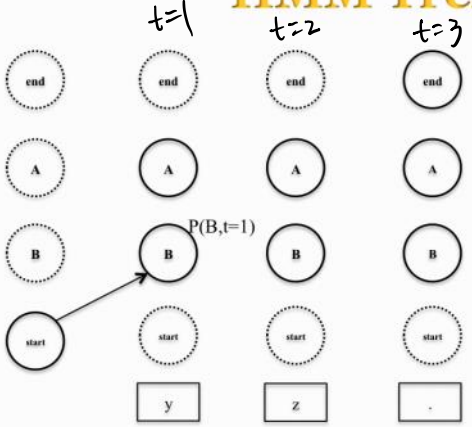


$P(A,t=1) =$
$P(start) \times P(A|start) \times P(y|A)$

omitting y symbol.

state A

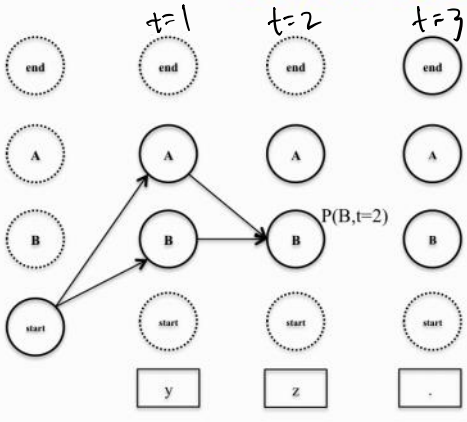
HMM Trellis



time = 1

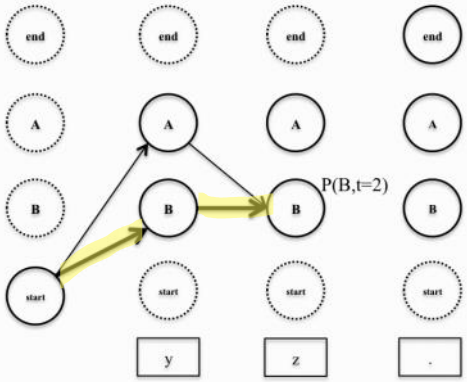
$P(B,t=1) =$
$P(start) \times P(B|start) \times P(y|B)$

HMM Trellis

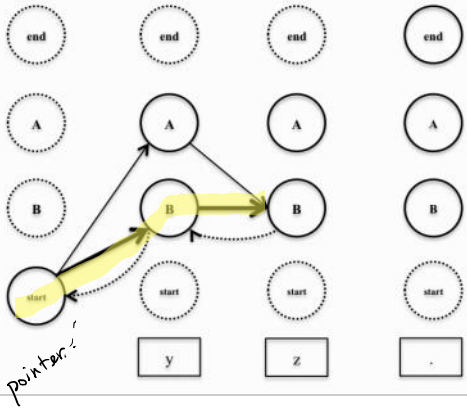


$P(B,t=2) =$
$\max(P(A,t=1) \times P(B|A) \times P(z|b),$ → Path from A
$P(B,t=1) \times P(B|B) \times P(z|b))$ → Path from B

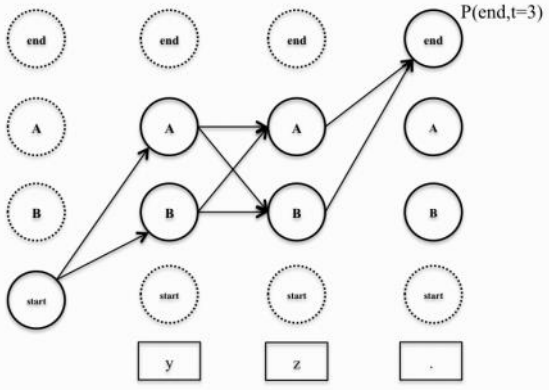
HMM Trellis



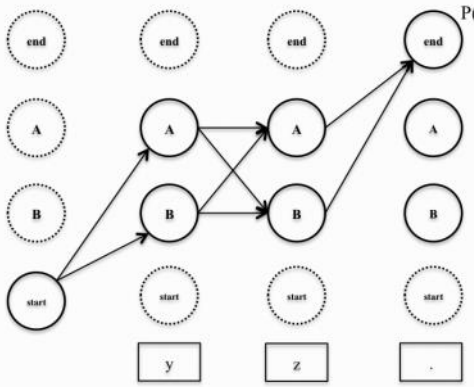
HMM Trellis



HMM Trellis



HMM Trellis



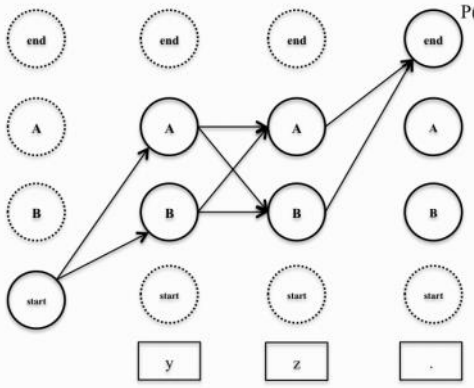
P(end,t=3)

P(end,t=3) =
max (P(A,t=2) x P(end|A),
P(B,t=2) x P(end|B))

→ A to end state

↳ B to end state

HMM Trellis



P(end,t=3)

P(end,t=3) =
max (P(A,t=2) x P(end|A),
P(B,t=2) x P(end|B))

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

Use the backpointers to find the sequence of states.

Best path encoded

NLP

7.4

NLP

07.04 Hidden Markov Models 2/2 NATURAL LANGUAGE PROCESSING

Introduction to NLP

Hidden Markov Models (cont'd)

07.04 Hidden Markov Models 2/2 NATURAL LANGUAGE PROCESSING

Observation Likelihood

- Given multiple HMMs
  - e.g., for different languages
  - Which one is the most likely to have generated the observation sequence

→ compare the likelihoods

- Naïve solution
  - try all possible state sequences

Forward Algorithm

- Dynamic programming method
  - Computing a forward trellis that encodes all possible state paths.
  - Based on the Markov assumption that the probability of being in any state at a given time point only depends on the probabilities of being in all states at the previous time point

previous time

HMM Learning

- Supervised
  - Training sequences are labeled
- Unsupervised
  - Training sequences are unlabeled
  - Known number of states

know the # of states

- Semi-supervised
  - Some training sequences are labeled

A few labeled several hundred a large number not labeled

Supervised HMM Learning

- Estimate the static transition probabilities using MLE

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

$Pr(s_j|s_i)$

- 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|μ)

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|μ) increases
  - it can be stopped at any point and give a partial solution
  - it converges to a local maximum

Expectation step

→ after thousands, the set of param diverges, can stop

NLP

7.5

Combine the models into Statistical POS tagging

NLP

07.05 Statistical POS Tagging NATURAL LANGUAGE PROCESSING

Introduction to NLP

Statistical POS Tagging

07.05 Statistical POS Tagging NATURAL LANGUAGE PROCESSING

Part of Speech Tagging Methods

- Rule-based
- Stochastic
  - HMM (generative)
  - Maximum Entropy MM (discriminative)
- Transformation-based

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 right hand side of the equation
  - P(W) can be ignored
  - P(T) is called the prior, P(W|T) is called the likelihood.

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
  - $T = \text{argmax } P(T|W) = \text{argmax } \prod P(w_i|t_i)\ P(t_i|t_{i-1})$

Maximum Likelihood Estimates

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

Prob. noun following adjectives

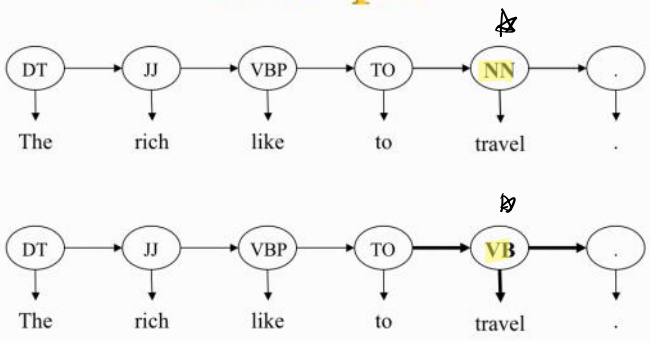
- P(this|DT) = C(DT,this)/C(dT)= 7037/103687 = .068

count of determiner.

Example

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

Example



Evaluating Taggers

- Data set
  - Training set
  - Development set
  - Test set
- Tagging accuracy
  - how many tags right
- Results
  - Accuracy around 97% on PTB trained on 800,000 words
  - (50-85% on unknown words; 50% for trigrams)
  - Upper bound 98% - noise (e.g., errors and inconsistencies in the data, e.g., NN vs JJ)

Transformation-Based Learning

- [Brill 1995]
- Example
  - P(NN|sleep) = .9
  - P(VB|sleep) = .1
  - Change NN to VB when the previous tag is TO
- Types of rules:
  - The preceding (following) word is tagged z
  - The word two before (after) is tagged z
  - One of the two preceding (following) words is tagged z
  - One of the three preceding (following) words is tagged z
  - The preceding word is tagged z and the following word is tagged w

07.05 Statistical POS Tagging
NATURAL LANGUAGE PROCESSING

Transformation Based Tagger

#	Change Tag From	To	Condition
1	NN	VB	Previous tag is *TO*
2	VBP	VB	One of the previous three tags is *MD*
3	NN	VB	One of the previous two tags is *MD*
4	VB	NN	One of the previous two tags is *DT*
5	VBD	VBN	One of the previous three tags is *VBZ*
6	VBN	VBD	Previous tag is *PRP*
7	VBN	VBD	Previous tag is *NNP*
8	VBD	VBN	Previous tag is *VBD*
9	VBP	VB	Previous tag is *TO*
10	POS	VBZ	Previous tag is *PRP*
11	VB	VBP	Previous tag is *NNS*
12	VBD	VBN	One of previous three tags is *VBP*
13	IN	WDT	One of next two tags is *VB*
14	VBD	VBN	One of previous two tags is *VB*
15	VB	VBP	Previous tag is *PRP*
16	IN	WDT	Next tag is *VBZ*
17	IN	DT	Next tag is *NN*
18	JJ	NNP	Next tag is *NNP*
19	IN	WDT	Next tag is *VBD*
20	JJR	RBR	Next tag is *JJ*

Figure 4
The first 20 nonlexicalized transformations.

07.05 Statistical POS Tagging
NATURAL LANGUAGE PROCESSING

Transformation Based Tagger

Change tag **a** *to tag* **b** *when:*

1. The preceding (following) word is *w*.
2. The word two before (after) is *w*.
3. One of the two preceding (following) words is *w*.
4. The current word is *w* and the preceding (following) word is *x*.
5. The current word is *w* and the preceding (following) word is tagged *z*.
6. The current word is *w*.
7. The preceding (following) word is *w* and the preceding (following) tag is *t*.
8. The current word is *w*, the preceding (following) word is w_2 and the preceding (following) tag is *t*.

where *w* and *x* are variables over all words in the training corpus, and *z* and *t* are variables over all parts of speech.

Transformation Based Tagger

#	Change Tag From	To	Condition
1	NN	NNS	Has suffix **-s**
2	NN	CD	Has character **.**
3	NN	JJ	Has character **-**
4	NN	VBN	Has suffix **-ed**
5	NN	VBG	Has suffix **-ing**
6	??	RB	Has suffix **-ly**
7	??	JJ	Adding suffix **-ly** results in a word.
8	NN	CD	The word **$** can appear to the left.
9	NN	JJ	Has suffix **-al**
10	NN	VB	The word **would** can appear to the left.
11	NN	CD	Has character **0**
12	NN	JJ	The word **be** can appear to the left.
13	NNS	JJ	Has suffix **-us**
14	NNS	VBZ	The word **it** can appear to the left.
15	NN	JJ	Has suffix **-ble**
16	NN	JJ	Has suffix **-ic**
17	NN	CD	Has character **1**
18	NNS	NN	Has suffix **-ss**
19	??	JJ	Deleting the prefix **un-** results in a word
20	NN	JJ	Has suffix **-ive**

Figure 6
The first 20 transformations for unknown words.

07.05 Statistical POS Tagging NATURAL LANGUAGE PROCESSING

Thoughts About POS Taggers

- New domains → hard to tag article of new domain → need to train again
  - Lower performance
- Distributional clustering
  - Combine statistics about semantically related words
  - Example: names of companies
  - Example: days of the week
  - Example: animals

07.05 Statistical POS Tagging NATURAL LANGUAGE PROCESSING

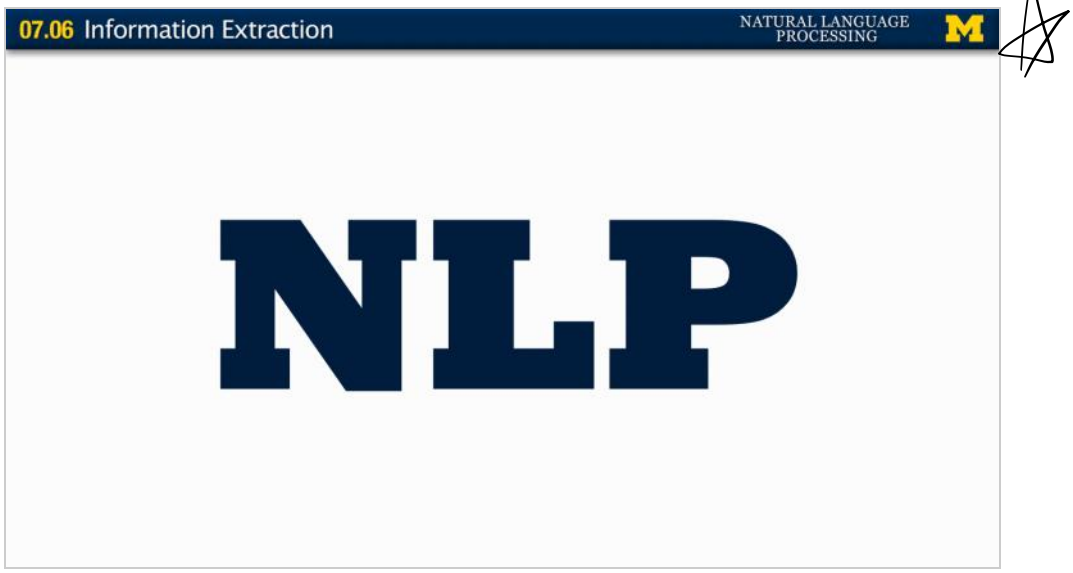
External Links

learning HMM

- 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

7.6

★ practically useful



Introduction to NLP

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

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.

Times and Events

- Times
  - Absolute expressions
  - Relative expressions (e.g., "last night")
- Events

Sequence Labeling

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 $w_1 w_2 w_3$
- Output
  - Labeled words
- Classification methods
  - Can use the categories of the previous tokens as features in classifying the next one
  - Direction matters

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.

NER, Time, and Event Extraction

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

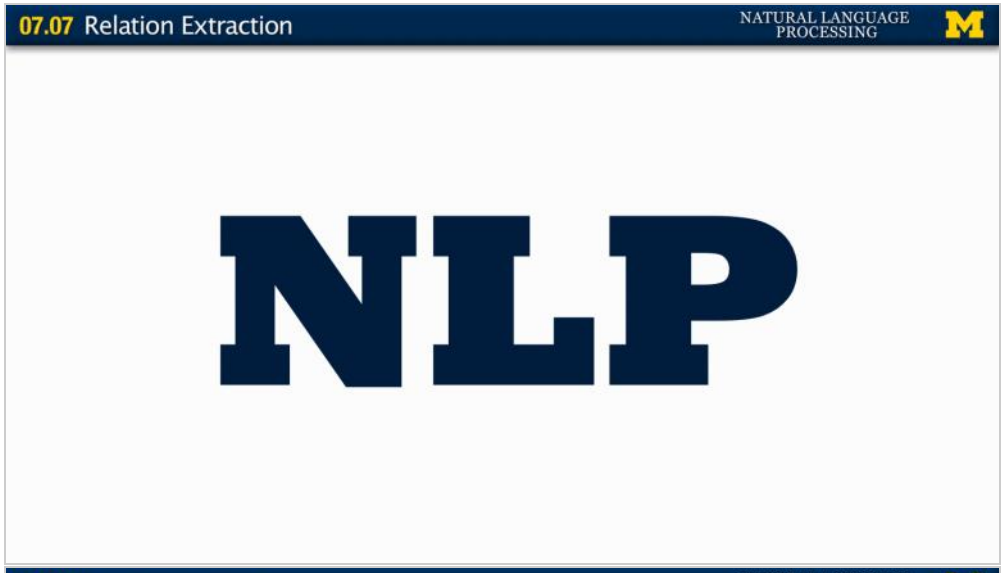
Biomedical Example

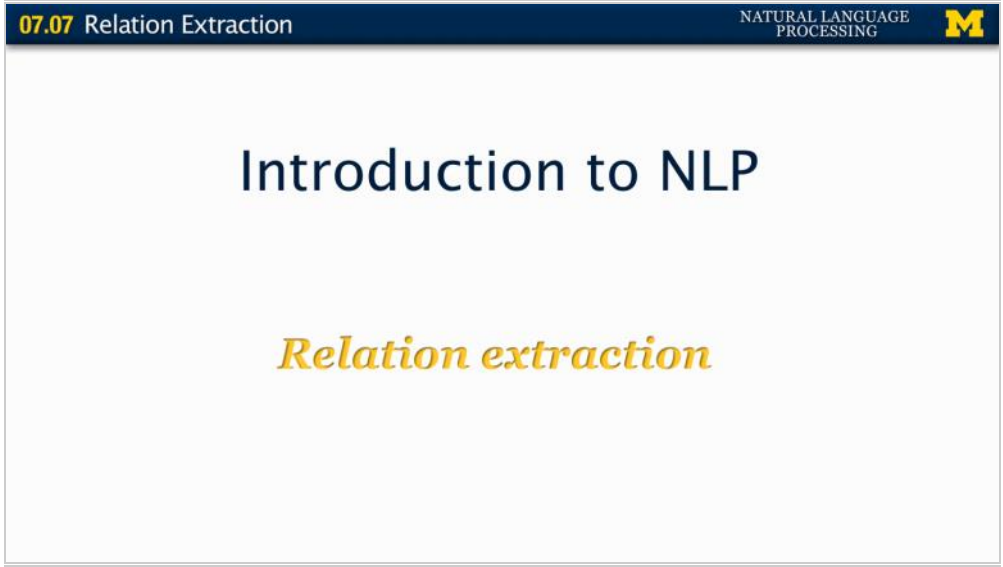
- Gene labeling
- Sentence:
  - [$_{GENE}$ BRCA1] and [$_{GENE}$ BRCA2] are human genes that produce tumor suppressor proteins

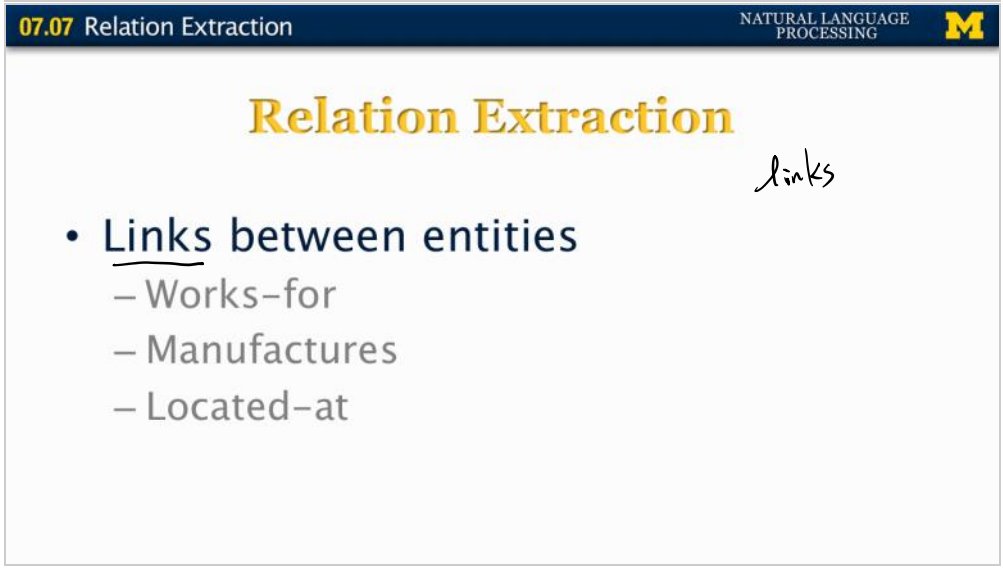
Gene labeling

NLP

7.7


07.07 Relation Extraction
NATURAL LANGUAGE PROCESSING
NLP


07.07 Relation Extraction
NATURAL LANGUAGE PROCESSING
Introduction to NLP
Relation extraction


07.07 Relation Extraction
NATURAL LANGUAGE PROCESSING
Relation Extraction
links
• Links between entities
– Works-for
– Manufactures
– Located-at

MUC

- Annual competition
  - DARPA, 1990s
- Events in news stories
  - Terrorist events
  - Joint ventures
  - Management changes
- Evaluation metrics
  - Precision
  - Recall
  - F-measure

MUC

extract events from news stories

MUC Example

<DOCNO> 0592 </DOCNO>
<DD> NOVEMBER 24, 1989, FRIDAY </DD>
<SO> Copyright (c) 1989 Jiji Press Ltd.; </SO>
<TXT>
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. THE MONTHLY OUTPUT WILL BE LATER RAISED TO 50,000 UNITS, BRIDGESTON SPORTS OFFICIALS SAID.
THE NEW COMPANY, BASED IN KAOHSIUNG, SOUTHERN TAIWAN, IS OWNED 75 PCT BY BRIDGESTONE SPORTS, 15 PCT BY UNION PRECISION CASTING CO. OF TAIWAN AND THE REMAINDER BY TAGA CO., A COMPANY ACTIVE IN TRADING WITH TAIWAN, THE OFFICIALS SAID.
BRIDGESTONE SPORTS HAS SO FAR BEEN ENTRUSTING PRODUCTION OF GOLF CLUB PARTS WITH UNION PRECISION CASTING AND OTHER TAIWAN COMPANIES.
WITH THE ESTABLISHMENT OF THE TAIWAN UNIT, THE JAPANESE SPORTS GOODS MAKER PLANS TO INCREASE PRODUCTION OF LUXURY CLUBS IN JAPAN.
</TXT>
</DOC>

Figure 2: A sample article from the MUC-5 English joint ventures task.

participants given material
→ identify companies
initiating the joint venture

Figure 3: A sample filled template from the MUC-5 English joint ventures task.

Example from
Grishman and Sundheim 1996

Other Examples

Application

- 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

Approaches

- View IE as a sequence labeling problem
  - Use HMM
- Use patterns
  - E.g., regular expressions
- Features
  - Capitalization (initial, allcaps), contains digits, spelling (e.g., suffixes), punctuation

Perl Regular Expressions

^	beginning of string; complement inside []
$	end of string
.	any character except newline
*	match 0 or more times
+	match 1 or more times
?	match 0 or 1 times
|	alternatives
( )	grouping and memory
[ ]	set of characters
{ }	repetition modifier
\	special symbol

Use RE for info extraction
Since are specific to certain Programming Language.

Perl Regular Expressions

*a**	zero or more
a+	one or more
a?	zero or one
a{m}	exactly *m*
a{m,}	at least *m*
a{m,n}	at least *m* but at most *n*
repetition?	shortest match

Perl Regular Expressions

\t	tab
\n	newline
\r	carriage return (CR)
*	asterisk
\?	question mark
\.	period
\x*hh*	hexadecimal character
\w	Matches one alphanumeric (or '_') character
\W	matches the complement of \w
\s	space, tab, newline
\S	complement of \s
\d	same as [0-9]
\D	complement of \d
\b	"word" boundary
\B	complement of \b
[x-y]	inclusive range from x to y

Sample Patterns

- Price (e.g., $14,000.00)
  - \$[0-9,]+(\.[0-9]{2})?
- Date (e.g., 2015-02-01)
  - ^(19|20)\d\d[- /.](0[1-9]|1[012])[- /.](0[1-9]|[12][0-9]|3[01])$
- Email
  - ^[_a-z0-9-]+(\.[_a-z0-9-]+)*@[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,4})$
- Person
- May include HTML code
- May include POS information
- May include Wordnet information

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	ob_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	.	.

Evaluating Template-based IE

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

Relation Extraction

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

ACE Evaluation

- 2002 newspaper data
- Entities:
  - Person, Organization, Facility, Location, Geopolitical Entity
- Relations:
  - Role, Part, Located, Near, Social

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
  - IsHeadquarteredIn (Mazda North American Operations, Irvine)

Relation Extraction

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

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

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

Example

- English
  - **Beethoven** *was born* in December **1770** in Bonn
  - *Born* in Bonn in **1770**, **Beethoven** …
  - After his *birth* on December 16, **1770**, **Beethoven** grew up in a musical family
  - **Ludwig van Beethoven** (**1770**–1827)
  - While this evidence supports the case for 16 December **1770** as **Beethoven**'s *date of birth*

Example (non-English)

- German
  - **Ludwig van Beethoven** *wurde* am 17. Dezember **1770** in Bonn *getauft*
  - **Ludwig van Beethoven** *wurde* in Bonn, 15. Dezember **1770**, eine Familie ursprünglich aus Brabant in Belgien *geboren*
  - Der *Geburtstag* von **Ludwig van Beethoven** wurde im Winter **1770** in Bonn nicht genau dokumentiert
- Spanish
  - **Ludwig van Beethoven** *nació* en Bonn el 17 de diciembre de **1770**
  - *Nacido* en Bonn **1770**, **Beethoven** …
  - **Ludwig van Beethoven**, *nace* en diciembre de **1770**

Semi-supervised Relation Extraction

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

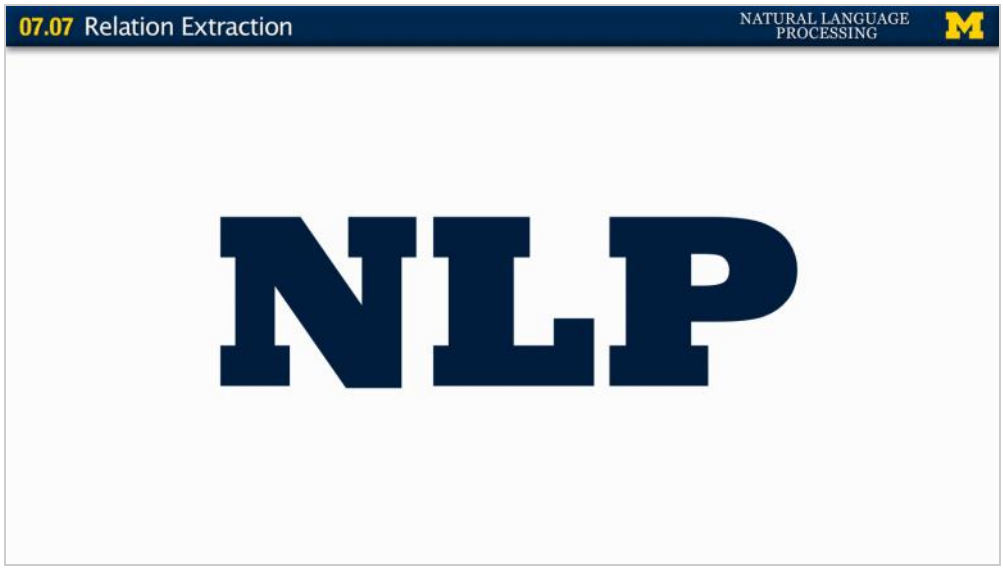
Evaluating Relation Extraction

- 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

Conclusion

- Probabilistic NLP
- Part of Speech Tagging
- Hidden Markov Models
- Information Extraction

Conditional random field (CRF) — not covered


NLP