Information Extraction: Coreference and Relation Extraction

Lecture #22

Introduction to Natural Language Processing
CMPSCI 585, Spring 2004
University of Massachusetts Amherst

Andrew McCallum

1

What is "Information Extraction"

As a family of techniques:

Information Extraction = segmentation + classification + association + clustering

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

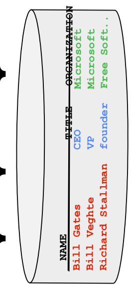
For years, Microsoft Corporation CEO Bill Gates railed against the economic philosophy of open-source software with Orwellian fervor, denouncing its communal licensing as a "cancer" that stifled technological innovation.

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."

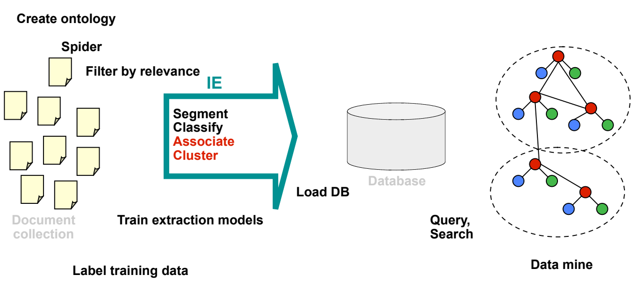
Richard Stallman, founder of the Free Software Foundation, countered saying...

* Microsoft Corporation
CEO
Bill Gates
* Microsoft
Gates
* Bill Veghte
Microsoft
VP
* Richard Stallman
founder
Free Software Foundation



2

IE in Context



3

Main Points

Co-reference
- How to cast as classification [Cardie]
- Measures of string similarity [Cohen]
- Scaling up [McCallum et al]

Relation extraction
- With augmented grammar [Miller et al 2000]
- With joint inference [Roth & Yih]
- Semi-supervised [Brin]

4

Coreference Resolution

AKA "record linkage", "database record deduplication", "citation matching", "object correspondence", "identity uncertainty"

Input

News article, with named-entity "mentions" tagged

Today Secretary of State Colin Powell met with . . . . he . . . . Condoleezza Rice . . . . Mr Powell . . . . she . . . . Powell . . . . President Bush . . . . Rice . . . . Bush . . . .

Output

Number of entities, *N = 3*

#1
Secretary of State Colin Powell
he
Mr. Powell
Powell

#2
Condoleezza Rice
she
Rice

#3
President Bush
Bush

5

Inside the Traditional Solution

Pair-wise Affinity Metric

Mention (3): . . . Mr Powell . . . — Y/N? — Mention (4): . . . Powell . . .

N	Two words in common	29
Y	One word in common	13
Y	"Normalized" mentions are string identical	39
Y	Capitalized word in common	17
Y	> 50% character tri-gram overlap	19
N	< 25% character tri-gram overlap	-34
Y	In same sentence	9
Y	Within two sentences	8
N	Further than 3 sentences apart	-1
Y	"Hobbs Distance" < 3	11
N	Number of entities in between two mentions = 0	12
N	Number of entities in between two mentions > 4	-3
Y	Font matches	1
Y	Default	-19
	OVERALL SCORE =	98 > threshold=0

6

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

> Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

> Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

> Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

> Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

IE Example: Input Text

SAN SALVADOR, 15 JAN 90 (ACAN-EFE) -- [TEXT] ARMANDO CALDERON SOL, PRESIDENT OF THE NATIONALIST REPUBLICAN ALLIANCE (ARENA), THE RULING SALVADORAN PARTY, TODAY CALLED FOR AN INVESTIGATION INTO ANY POSSIBLE CONNECTION BETWEEN THE MILITARY PERSONNEL IMPLICATED IN THE ASSASSINATION OF JESUIT PRIESTS.

"IT IS SOMETHING SO HORRENDOUS, SO MONSTROUS, THAT WE MUST INVESTIGATE THE POSSIBILITY THAT THE FMLN (FARABUNDO MARTI NATIONAL LIBERATION FRONT) STAGED THESE MURDERS TO DISCREDIT THE GOVERNMENT," CALDERON SOL SAID.

SALVADORAN PRESIDENT ALFREDO CRISTIANI IMPLICATED FOUR OFFICERS, INCLUDING ONE COLONEL, AND FIVE MEMBERS OF THE ARMED FORCES IN THE ASSASSINATION OF SIX JESUIT PRIESTS AND TWO WOMEN ON 16 NOVEMBER AT THE CENTRAL AMERICAN UNIVERSITY.

IE Example: Output Template

1. DATE	16 NOV 90
2. LOCATION	EL SALVADOR: CENTRAL AMERICAN UNIVERSITY
3. TYPE	MURDER
4. STAGE OF EXECUTION	ACCOMPLISHED
5. INCIDENT CATEGORY	TERRORIST ACT
6. PERP: INDIVIDUAL ID	"FOUR OFFICERS" "ONE COLONEL" "FIVE MEMBERS OF THE ARMED FORCES"
7. PERP: ORGANIZATION ID	"ARMED FORCES", "FMLN"
8. PERP: CONFIDENCE	REPORTED AS FACT; ACCUSED BY GOVT
9. HUM TGT: DESCRIPTION	"JESUITS" "WOMEN"
10. HUM TGT: TYPE	CIVILIAN: "JESUITS" CIVILIAN: "WOMEN"
11. HUM TGT: NUMBER	6: "JESUITS" 2: "WOMEN"
12. EFFECT OF INCIDENT	DEATH: "JESUITS" DEATH: "WOMEN"

IE Example: Coreference

SAN SALVADOR, 15 JAN 90 (ACAN-EFE) -- [TEXT] ARMANDO CALDERON SOL, PRESIDENT OF THE NATIONALIST REPUBLICAN ALLIANCE (ARENA), THE RULING SALVADORAN PARTY, TODAY CALLED FOR AN INVESTIGATION INTO ANY POSSIBLE CONNECTION BETWEEN THE MILITARY PERSONNEL IMPLICATED IN THE ASSASSINATION OF JESUIT PRIESTS.

"IT IS SOMETHING SO HORRENDOUS, SO MONSTROUS, THAT WE MUST INVESTIGATE THE POSSIBILITY THAT THE FMLN (FARABUNDO MARTI NATIONAL LIBERATION FRONT) STAGED THESE MURDERS TO DISCREDIT THE GOVERNMENT," CALDERON SOL SAID.

SALVADORAN PRESIDENT ALFREDO CRISTIANI IMPLICATED FOUR OFFICERS, INCLUDING ONE COLONEL, AND FIVE MEMBERS OF THE ARMED FORCES IN THE ASSASSINATION OF SIX JESUIT PRIESTS AND TWO WOMEN ON 16 NOVEMBER AT THE CENTRAL AMERICAN UNIVERSITY.

Why It's Hard

Many sources of information play a role

- head noun matches
  - IBM *executives* = the *executives*
- syntactic constraints
  - John helped himself to...
  - John helped him to…
- number and gender agreement
- discourse focus, recency, syntactic parallelism, semantic class, world knowledge, …

Why It's Hard

- No single source is a completely reliable indicator
  - number agreement
    - the assassination = these murders
- Identifying each of these features automatically, accurately, and in context, is hard
- Coreference resolution subsumes the problem of pronoun resolution…

A Machine Learning Approach

- Classification
  - given a description of two noun phrases, NP_i and NP_j, classify the pair as *coreferent* or *not coreferent*

coref ? *coref ?*

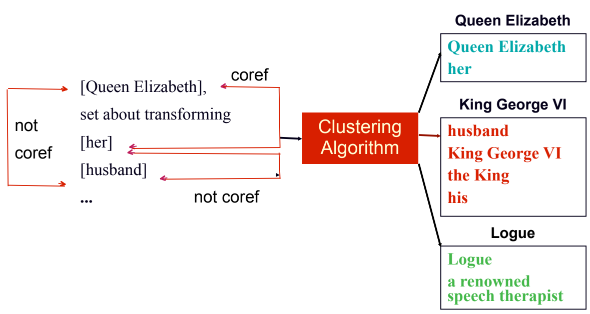
[Queen Elizabeth] set about transforming [her] [husband], ...

not coref ?

Aone & Bennett [1995]; Connolly et al. [1994]; McCarthy & Lehnert [1995]; Soon et al. [2001]; Ng & Cardie [2002]; …

A Machine Learning Approach

- Clustering
  - coordinates pairwise coreference decisions



Machine Learning Issues

- Training data creation
- Instance representation
- Learning algorithm
- Clustering algorithm

Supervised Inductive Learning

Examples of NP pairs (features + class)

↓

ML Algorithm

↓

(novel) pair of NPs (features) ⟶ Concept description (program) ⟶ label

19

Training Data Creation

- Creating training instances
  - texts annotated with coreference information
  - one instance *inst*(NP_i, NP_j) for each pair of NPs
    - assumption: NP_i precedes NP_j
    - feature vector: describes the two NPs and context
    - class value:

coref	pairs on the same coreference chain
not coref	otherwise

20

Instance Representation

- 25 features per instance
  - lexical (3)
    - string matching for pronouns, proper names, common nouns
  - grammatical (18)
    - pronoun, demonstrative (the, this), indefinite (it is raining), …
    - number, gender, animacy
    - appositive (george, the king), predicate nominative (a horse is a mammal)
    - binding constraints, simple contra-indexing constraints, …
    - span, maximalnp, …
  - semantic (2)
    - same WordNet class
    - alias
  - positional (1)
    - distance between the NPs in terms of # of sentences
  - knowledge-based (1)
    - naïve pronoun resolution algorithm

21

Learning Algorithm

- RIPPER (Cohen, 1995)
  C4.5 (Quinlan, 1994)
  - rule learners
    - input: set of training instances
    - output: coreference classifier
- Learned classifier
  - input: test instance (represents pair of NPs)
  - output: classification
    confidence of classification

22

Clustering Algorithm

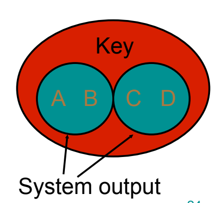
- Best-first single-link clustering
  - Mark each NP_j as belonging to its own class:
    $NP_j \in c_j$
  - Proceed through the NPs in left-to-right order.
    - For each NP, NP_j, create test instances, *inst*(NP_i, NP_j), for all of its preceding NPs, NP_i.
    - Select as the antecedent for NP_j the highest-confidence coreferent NP, NP_i, according to the coreference classifier (or none if all have below .5 confidence);

      Merge c_j and c_j .

23

Evaluation

- MUC-6 and MUC-7 coreference data sets
- documents annotated w.r.t. coreference
- 30 + 30 training texts (dry run)
- 30 + 20 test texts (formal evaluation)
- scoring program
  - recall
  - precision
  - F-measure: 2PR/(P+R)



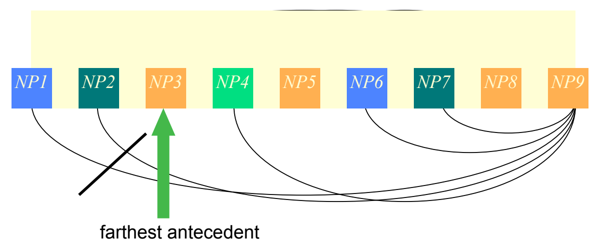
24

Baseline Results

	MUC-6			MUC-7		
	R	P	F	R	P	F
Baseline	40.7	73.5	52.4	27.2	86.3	41.3
Worst MUC System	36	44	40	52.5	21.4	30.4
Best MUC System	59	72	65	56.1	68.8	61.8

Problem 1

- Coreference is a rare relation
  - skewed class distributions (2% positive instances)
  - *remove some negative instances*



Problem 2

- Coreference is a discourse-level problem
  - different solutions for different types of NPs
    - proper names: string matching and aliasing
  - inclusion of "hard" positive training instances
  - *positive example selection*: selects easy positive training instances (cf. Harabagiu *et al.* (2001))

Queen Elizabeth set about transforming her husband,
King George VI, into a viable monarch. Logue,
the renowned speech therapist, was summoned to help
the King overcome his speech impediment...

Problem 3

- Coreference is an equivalence relation
  - loss of transitivity
  - need to tighten the connection between classification and clustering
  - *prune learned rules w.r.t. the clustering-level coreference scoring function*

coref ? *coref ?*

[Queen Elizabeth] set about transforming [her] [husband], ...

not coref ?

Results

	MUC-6			MUC-7		
	R	P	F	R	P	F
Baseline	40.7	73.5	52.4	27.2	86.3	41.3
NEG-SELECT	46.5	67.8	55.2	37.4	59.7	46.0
POS-SELECT	53.1	80.8	64.1	41.1	78.0	53.8
NEG-SELECT + POS-SELECT	63.4	76.3	69.3	59.5	55.1	57.2
NEG-SELECT + POS-SELECT + RULE-SELECT	63.3	76.9	69.5	54.2	76.3	63.4

- Ultimately: large increase in F-measure, due to gains in recall

Comparison with Best MUC Systems

	MUC-6			MUC-7		
	R	P	F	R	P	F
NEG-SELECT + POS-SELECT + RULE-SELECT	63.3	76.9	69.5	54.2	76.3	63.4
Best MUC System	59	72	65	56.1	68.8	61.8

Supervised ML for NP Coreference

- Good performance compared to other systems, but…**lots** of room for improvement
  - Common nouns < pronouns < proper nouns
  - Tighter connection between classification and clustering is possible
    - Rich Caruana's ensemble methods
    - Statistical methods for learning probabilistic relational models (Getoor *et al.*, 2001; Lafferty et al., 2001; Taskar *et al.*, 2003; McCallum and Wellner, 2003).
  - Need additional data sets
    - New release of ACE data from Penn's LDC
    - General problem: reliance on manually annotated data…

Main Points

Co-reference
- How to cast as classification [Cardie]
- **Measures of string similarity [Cohen]**
- Scaling up [McCallum et al]

Relation extraction
- With augmented grammar [Miller et al 2000]
- With joint inference [Roth & Yih]
- Semi-supervised [Brin]

Record linkage: definition

- *Record linkage:* determine if pairs of *data records* describe the same entity
  - I.e., find record pairs that are *co-referent*
  - Entities: usually people (or organizations or…)
  - Data records: names, addresses, job titles, birth dates, …
- Main applications:
  - *Joining* two heterogeneous relations
  - *Removing duplicates* from a single relation

Record linkage: terminology

- The term "*record linkage"* is possibly co-referent with:
  - For DB people: data matching, merge/purge, duplicate detection, data cleansing, ETL (extraction, transfer, and loading), de-duping
  - For AI/ML people: reference matching, database hardening, object consolidation,
  - In NLP: co-reference/anaphora resolution
  - Statistical matching, clustering, language modeling, …

Finding a technical paper *c.* 1995

- Start with citation:

> " Experience With a Learning Personal Assistant", T.M. Mitchell, R. Caruana, D. Freitag, J. McDermott, and D. Zabowski, *Communications of the ACM*, Vol. 37, No. 7, pp. 81-91, July 1994.

- Find author's institution (w/ INSPEC)
- Find web host (w/ NETFIND)
- Find author's home page and (hopefully) the paper by browsing

The data integration problem

internet host	*institution*
cs.ucsd.edu	computer science department, university of california, san diego
cs.stanford.edu	computer science department, stanford university, palo alto, california
(INSPEC)	Dept. of Comput. Sci., California Univ., San Diego, La Jolla, CA, USA.
(INSPEC)	Dept. of Comput. Sci. Stanford Univ., CA, USA.

String distance metrics: overview

- Term-based (e.g. TF/IDF as in WHIRL)
  - Distance depends on **set of words** contained in both *s* and *t*.
- Edit-distance metrics
  - Distance is **shortest sequence of edit commands** that transform *s* to *t*.
- Pair HMM based metrics
  - Probabilistic extension of edit distance
- Other metrics

String distance metrics: term-based

- Term-based (e.g. TFIDF as in WHIRL)
  - Distance between *s* and *t* based on **set of words** appearing in both *s* and *t*.
  - Order of words is **not** relevant
    - E.g, “Cohen, William” = “William Cohen” and “James Joyce = Joyce James”
  - Words are usually **weighted** so common words count less
    - E.g. “Brown” counts less than “Zubinsky”
    - Analogous to Felligi-Sunter’s Method 1

Jaccard Distance

S		William	Cohen	CM	Univ		Pgh
T	Dr.	William	Cohen	CM		University	
$\lvert S \cup T \rvert$	Dr.	William	Cohen	CM	Univ	University	Pgh
$\lvert S \cap T \rvert$		William	Cohen	CM			

$$\text{Jaccard Score} = \frac{|S \cap T|}{|S \cup T|} = \frac{3}{7}$$

String distance metrics: term-based

- Advantages:
  - Exploits **frequency** information
  - Efficiency: Finding { *t : sim(t,s)>k* } is sublinear!
  - Alternative word orderings ignored (William Cohen *vs* Cohen, William)
- Disadvantages:
  - Sensitive to spelling errors (Willliam Cohon)
  - Sensitive to abbreviations (Univ. vs University)
  - Alternative word orderings ignored (James Joyce *vs* Joyce James, City National Bank *vs* National City Bank)

String distance metrics: Levenshtein

- Edit-distance metrics
  - Distance is **shortest sequence of edit commands** that transform *s* to *t*.
  - Simplest set of operations:
    - Copy character from *s* over to *t*
    - Delete a character in *s* (cost 1)
    - Insert a character in *t* (cost 1)
    - Substitute one character for another (cost 1)
  - This is “Levenshtein distance”

Levenshtein distance - example

- distance(“William Cohen”, “Willliam Cohon”)

s	W	I	L	L	gap	I	A	M	_	C	O	H	E	N
t	W	I	L	L	L	I	A	M	_	C	O	H	O	N
op	C	C	C	C	**I**	C	C	C	C	C	C	C	**S**	C
cost	0	0	0	0	1	1	1	1	1	1	1	1	2	2

alignment

Computing Levenshtein distance - 1

D(i,j) = score of **best** alignment from *s1..si* to *t1..tj*

$$
= \min \begin{cases} D(i-1,j-1), \text{ if } si=tj & \textit{//copy} \\ D(i-1,j-1)+1, \text{ if } si!=tj & \textit{//substitute} \\ D(i-1,j)+1 & \textit{//insert} \\ D(i,j-1)+1 & \textit{//delete} \end{cases}
$$

Computing Levenshtein distance - 2

D(i,j) = score of **best** alignment from *s1..si* to *t1..tj*

$$
= \min \begin{cases} D(i-1,j-1) + d(si,tj) & \textit{//subst/copy} \\ D(i-1,j)+1 & \textit{//insert} \\ D(i,j-1)+1 & \textit{//delete} \end{cases}
$$

(simplify by letting d(c,d)=0 if c=d, 1 else)

also let D(i,0)=i *(for i inserts)* and D(0,j)=j

Computing Levenshtein distance - 3

$$
D(i,j) = \min \begin{cases} D(i-1,j-1) + d(si,tj) & \textit{//subst/copy} \\ D(i-1,j)+1 & \textit{//insert} \\ D(i,j-1)+1 & \textit{//delete} \end{cases}
$$

	C	O	H	E	N
M	1	2	3	4	5
C	1	2	3	4	5
C	2	2	3	4	5
O	3	2	3	4	5
H	4	3	2	3	4
N	5	4	3	3	3

= D(*s*,*t*)

Computing Levenshtein distance – 4

$$
D(i,j) = \min \begin{cases} D(i-1,j-1) + d(si,tj) & \textit{//subst/copy} \\ D(i-1,j)+1 & \textit{//insert} \\ D(i,j-1)+1 & \textit{//delete} \end{cases}
$$

A *trace* indicates where the min value came from, and can be used to find edit operations and/or a best *alignment* (may be more than 1)

	C	O	H	E	N
M	**1**	2	3	4	5
C	**1**	2	3	4	5
C	**2**	3	3	4	5
O	3	**2**	3	4	5
H	4	3	**2**	**3**	4
N	5	4	3	3	**3**

Needleman-Wunch distance

$$
D(i,j) = \min \begin{cases} D(i-1,j-1) + d(si,tj) & \textit{//subst/copy} \\ D(i-1,j) + \boldsymbol{G} & \textit{//insert} \\ D(i,j-1) + \boldsymbol{G} & \textit{//delete} \end{cases}
$$

G = "gap cost"

d(c,d) is an arbitrary distance function on characters (e.g. related to typo frequencies, amino acid substitutibility, etc)

William Cohen

Wukkuan Cigeb

Smith-Waterman distance

- Instead of looking at each sequence in its entirety, this compares segments of all possible lengths and chooses whichever maximise the similarity measure. (Thus it is a generalization of "longest common subsequence.")
- For every cell the algorithm calculates all possible paths leading to it. These paths can be of any length and can contain insertions and deletions.

Smith-Waterman distance

$$D(i,j) = \max \begin{cases} 0 & \textbf{//start over} \\ D(i-1,j-1) - d(si,tj) & \textit{//subst/copy} \\ D(i-1,j) - G & \textit{//insert} \\ D(i,j-1) - G & \textit{//delete} \end{cases}$$

$G = 1$

$d(c,c) = -2$

$d(c,d) = +1$

	C	**O**	**H**	**E**	**N**
M	0	0	0	0	0
C	**+2**	0	0	0	0
C	**+2**	0	0	0	0
O	0	**+4**	+3	0	0
H	0	+3	**+6**	**+5**	+3
N	0	+2	+5	+5	**+7**

Smith-Waterman distance: Monge & Elkan's WEBFIND (1996)

internet host	*institution*
cs.ucsd.edu	computer science department, university of california, san diego
cs.stanford.edu	computer science department, stanford university, palo alto, california
(INSPEC)	Dept. of Comput. Sci., California Univ., San Diego, La Jolla, CA, USA.
(INSPEC)	Dept. of Comput. Sci. Stanford Univ., CA, USA.

Table 1: Example of NETFIND and INSPEC fields.

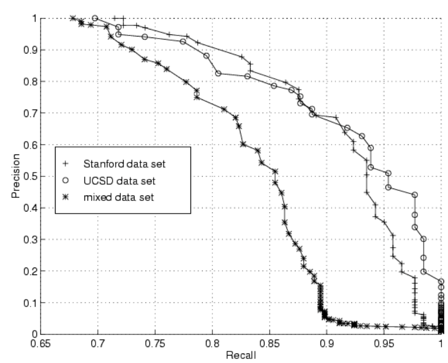
Smith-Waterman distance in Monge & Elkan's WEBFIND (1996)

Used a **standard version** of Smith-Waterman with hand-tuned weights for inserts and character substitutions.

Split large text fields by separators like commas, etc, and found minimal cost over **all possible pairings** of the subfields (since S-W assigns a large cost to large transpositions)

Result **competitive** with plausible competitors.

Results: S-W from Monge & Elkan



Affine gap distances

- Smith-Waterman fails on some pairs that seem quite similar:

William W. Cohen

William W. 'Don't call me Dubya' Cohen

Intuitively, single long insertions are "cheaper" than a lot of short insertions

Affine gap distances - 2

- Idea:
  - Current cost of a "gap" of *n* characters: *nG*
  - Make this cost: *A + (n-1)B,* where *A* is cost of "opening" a gap, and *B* is cost of "continuing" a gap.

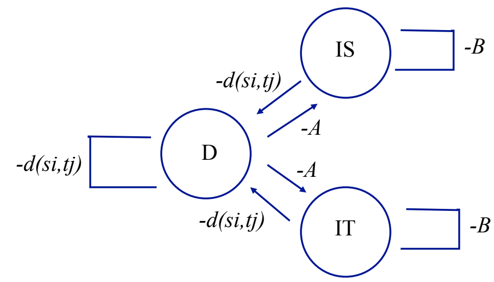
Affine gap distances - 3

$$D(i,j) = \max \begin{cases} D(i\text{-}1,j\text{-}1) + d(si,tj) \\ IS(I\text{-}1,j\text{-}1) + d(si,tj) \\ IT(I\text{-}1,j\text{-}1) + d(si,tj) \end{cases}$$

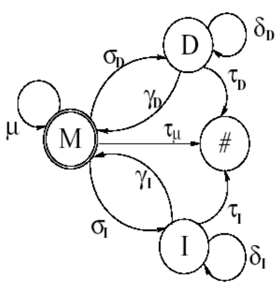
$$IS(i,j) = \max \begin{cases} D(i\text{-}1,j) - A \\ IS(i\text{-}1,j) - B \end{cases}$$ *Best score in which si is aligned with a 'gap'*

$$IT(i,j) = \max \begin{cases} D(i,j\text{-}1) - A \\ IT(i,j\text{-}1) - B \end{cases}$$ *Best score in which tj is aligned with a 'gap'*

Affine gap distances as automata



Generative version of affine gap automata (Bilenko&Mooney, TechReport 02)



HMM emits **pairs:** *(c,d)* in state *M*, pairs *(c,-)* in state *D*, and pairs *(-,d)* in state *I.*

For each state there is a **multinomial** distribution on pairs.

The HMM can trained with EM from a sample of pairs of **matched** strings *(s,t)*

E-step is forward-backward; M-step uses some *ad hoc* smoothing

Affine gap edit-distance learning: experiments results (Bilenko & Mooney)

Table 2: Sample duplicate records from the RESTAURANT database

name	address	city	phone	cuisine
Second Avenue Deli	156 2nd Ave. at 10th St.	New York	212/677-0606	Delicatessen
Second Avenue Deli	156 Second Ave.	New York City	212-677-0606	Delis

Table 3: Sample duplicate records from the MAILING database

first	last	street address	city
Tsy C	Dodgson	18 Lilammal Ave 3k1	Christina MT 59423
Tessy	Dodgeson	PO Box 3879	Christina MT 59428

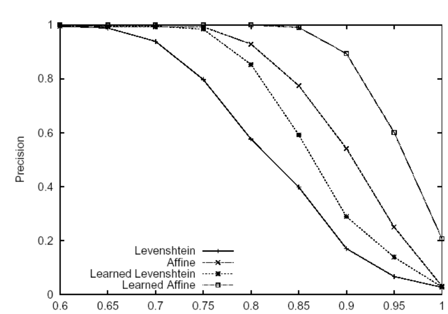
Experimental method: parse records into fields; append a few key fields together; sort by similarity; pick a threshold *T* and call all pairs with distance$(s,t) < T$ "duplicates"; picking *T* to maximize F-measure.

Affine gap edit-distance learning: experiments results (Bilenko & Mooney)

Distance metric	CORA title	RESTAURANT name
Levenshtein	0.870	0.843
Learned Levenshtein	0.902	**0.886**
Affine	0.917	0.883
Learned Affine	**0.971**	**0.967**

Distance met	RESTAURANT address	MAILING name	MAILING address
Levenshtein	0.950	0.867	0.878
Learned Lev	0.975	**0.899**	0.897
Affine	0.870	0.923	0.886
Learned Affi	**0.929**	**0.959**	0.892

Affine gap edit-distance learning: experiments results (Bilenko & Mooney)



Precision/recall for MAILING dataset duplicate detection

Affine gap distances – experiments (from McCallum,Nigam,Ungar KDD2000)

- Goal is to match data like this:

Fahlmann, Scott & Lebiere, Christian (1989). The cascade-correlation learning architecture. In Touretzky, D., editor, Advances in Neural Information Processing Systems (volume 2), (pp. 524-532), San Mateo, CA. Morgan Kaufmann.

Fahlman, S.E. and Lebiere, C., "The Cascade Correlation Learning Architecture," NIPS, Vol. 2, pp. 524-532, Morgan Kaufmann, 1990.

Fahlmann, S. E. and Lebiere, C. (1989). The cascade-correlation learning architecture. In Advances in Neural Information Processing Systems 2 (NIPS-2), Denver, Colorado.

Figure 2: Three sample citations to the same paper.

Affine gap distances – experiments (from McCallum,Nigam,Ungar KDD2000)

- Hand-tuned edit distance
- Lower costs for affine gaps
- Even lower cost for affine gaps near a "."
- HMM-based **normalization** to group title, author, booktitle, etc into **fields**

Affine gap distances – experiments

	TFIDF	Edit Distance	Adaptive
Cora	0.751	0.839	**0.945**
	0.721		**0.964**
OrgName1	**0.925**	0.633	0.923
	0.366	**0.950**	0.776
Orgname2	**0.958**	0.571	**0.958**
	0.778	0.912	**0.984**
Restaurant	0.981	0.827	**1.000**
	0.967	0.867	0.950
Parks	0.976	0.967	**0.984**
	0.967	0.967	**0.967**

String distance metrics: outline

- Term-based (e.g. TF/IDF as in WHIRL)
  - Distance depends on **set of words** contained in both *s* and *t*.
- Edit-distance metrics
  - Distance is **shortest sequence of edit commands** that transform *s* to *t*.
- Pair HMM based metrics
  - Probabilistic extension of edit distance
- **Other metrics**

Jaro metric

- Jaro metric is (apparently) tuned for personal names:
  - Given *(s,t)* define *c* to be *common in s,t* if it *si=c, tj=c,* and |*i-j*|<*min(*|*s*|*,*|*t*|*)/2.*
  - Define *c,d* to be a *transposition* if *c,d* are common and *c,d* appear in different orders in *s* and *t*.
  - Jaro(*s,t)* = average of *#common*/|*s*|, *#common*/|*t*|, and 0.5*#transpositions/#common*
  - Variant: weight errors early in string more heavily
- Easy to compute – note edit distance is O(|*s*||*t*|)

NB. This is my interpretation of Winkler's description

Jaro metric

	W	I	L	L	I	A	M
W	$\boxed{1}$	0	0	0	0	0	0
I	0	$\boxed{1}$	0	0	1	0	0
L	0	0	$\boxed{1}$	1	0	0	0
L	0	0	1	$\boxed{1}$	0	0	0
L	0	0	1	1	$\boxed{0}$	0	0
A	0	0	0	0	0	$\boxed{1}$	0
I	0	1	0	0	1	0	$\boxed{0}$
M	0	0	0	0	0	0	1

Illustration of the Jaro metric. Boxed entries are on the main diagonal, and every character in a row (column) which contains a boldfaces one is considered to be "in common" with the string "WILLIAM" ("WILLLAIM").

$$Jaro(s,t) = \frac{1}{3} \cdot \left(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{|s'|} \right)$$

$|s'| = |t'| =$ no. of characters common to s and t.

$T_{s',t'} =$ no. of transpositions for s' and t'

Soundex metric

- Soundex is a coarse **phonetic** indexing scheme, widely used in genealogy.
- Every Soundex code consists of a letter and three numbers between 0 and 6, e.g. B-536 for "Bender". The letter is always the first letter of the surname. The numbers hash together the rest of the name.
  - Vowels are generally ignored: e.g. Lee, Lu => L-000. Later later consonants in a name are ignored.
  - Similar-sounding letters (e.g. B, P, F, V) are not differentiated, nor are doubled letters.
  - There are lots of Soundex variants….

N-gram metric

– Idea: split every string *s* into a set of *all* character n-grams that appear in *s,* for *n<=k.* Then, use term-based approaches.

– e.g. "COHEN" => {C,O,H,E,N,CO,OH,HE,EN,COH,OHE,HEN}

– For n=4 or 5, this is competitive on retrieval tasks. It doesn't seem to be competitive with small values of n on matching tasks (but it's useful as a fast approximate matching scheme)

Main Points

Co-reference

- How to cast as classification [Cardie]
- Measures of string similarity [Cohen]
- **Scaling up [McCallum et al]**

Relation extraction

- With augmented grammar [Miller et al 2000]
- With joint inference [Roth]
- Semi-supervised [Brin]

Reference Matching

- Fahlman, Scott & Lebiere, Christian (1989). The cascade-correlation learning architecture. In Touretzky, D., editor, Advances in Neural Information Processing Systems (volume 2), (pp. 524-532), San Mateo, CA. Morgan Kaufmann.
- Fahlman, S.E. and Lebiere, C., "The Cascade Correlation Learning Architecture," NIPS, Vol. 2, pp. 524-532, Morgan Kaufmann, 1990.
- Fahlman, S. E. (1991) The recurrent cascade-correlation learning architecture. In Lippman, R.P. Moody, J.E., and Touretzky, D.S., editors, NIPS 3, 190-205.

The Citation Clustering Data

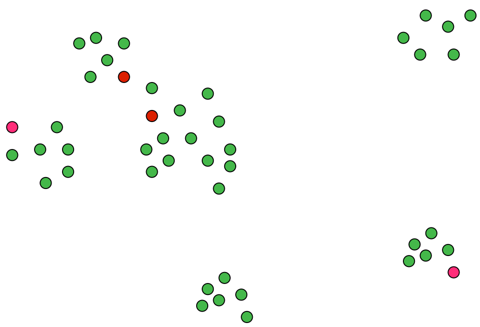
- Over 1,000,000 citations
- About 100,000 unique papers
- About 100,000 unique vocabulary words

- Over 1 trillion distance calculations

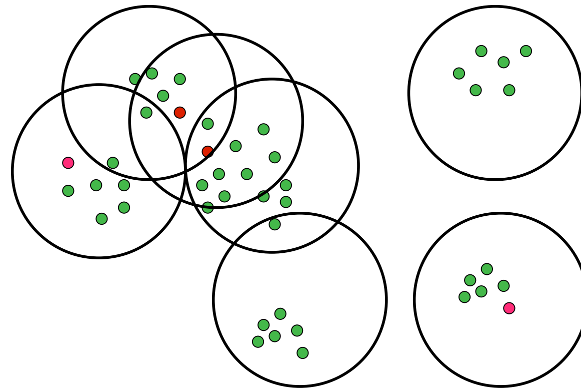
The Canopies Approach

- Two distance metrics: cheap & expensive
- First Pass
  – very inexpensive distance metric
  – create overlapping canopies
- Second Pass
  – expensive, accurate distance metric
  – canopies determine which distances calculated

Illustrating Canopies

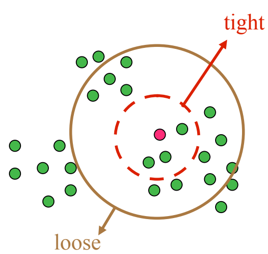


Overlapping Canopies



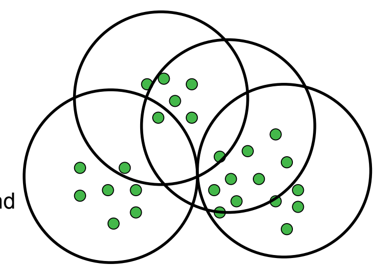
Creating canopies with two thresholds

- Put all points in D
- Loop:
  - Pick a point X from D
  - Put points within K_{loose} of X in canopy
  - Remove points within K_{tight} of X from D



Using canopies with Greedy Agglomerative Clustering

- Calculate expensive distances between points in the same canopy
- All other distances default to infinity
- Sort finite distances and iteratively merge closest



Computational Savings

- inexpensive metric << expensive metric
- # canopies per data point: f (small, but > 1)
- number of canopies: c (large)
- complexity reduction:

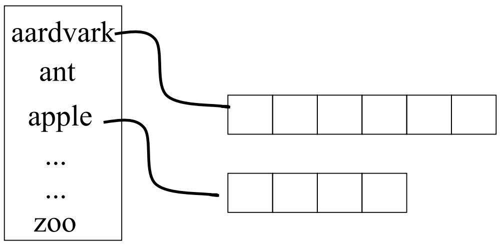
$$O\left(\frac{f^2}{c}\right)$$

The Experimental Dataset

- All citations for authors:
  - Michael Kearns
  - Robert Schapire
  - Yoav Freund
- 1916 citations
- 121 unique papers
- Similar dataset used for parameter tuning

Inexpensive Distance Metric for Text

- Word-level matching (TFIDF)
- Inexpensive using an inverted index



Expensive Distance Metric for Text

- String edit distance
- Compute with Dynamic Programming
- Costs for character:
  - insertion
  - deletion
  - substitution
  - ...

		S	e	c	a	t
	0.0	0.7	1.4	2.1	2.8	3.5
S	0.7	0.0	0.7	1.1	1.4	1.8
c	1.4	0.7	1.0	0.7	1.4	1.8
o	2.1	1.1	1.7	1.4	1.7	2.4
t	2.8	1.4	2.1	1.8	2.4	1.7
t	3.5	1.8	2.4	2.1	2.8	2.4

do Fahlman vs Falman

Extracting Fields using HMMs

Fahlman, S.E. and Lebiere, C., "The Cascade Correlation Learning Architecture," NIPS, Vol. 2, pp. 524-532, Morgan Kaufmann, 1990.

Author: Fahlman, S.E. and Lebiere, C.

Title: The Cascade Correlation Learning Architecture

Venue: NIPS

Year: 1990

Experimental Results

	F1	Minutes
Canopies GAC	0.838	7.65
Complete GAC	0.835	134.09
Existing Cora	0.784	0.03
Author/Year	0.697	0.03

Add precision, recall along side F1

Main Points

Co-reference
- How to cast as classification [Cardie]
- Measures of string similarity [Cohen]
- Scaling up [McCallum et al]

Relation extraction
- **With augmented grammar [Miller et al 2000]**
- With joint inference [Roth & Yih]
- Semi-supervised [Brin]

Information Extraction

Named Entity Recognition

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

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

Relationships between Entities

INPUT: Boeing is located in Seattle. Alan Mulally is the CEO.

OUTPUT:

{Relationship = Company-Location
Company = Boeing
Location = Seattle}

{Relationship = Employer-Employee
Employer = Boeing Co.
Employee = Alan Mulally}

Extraction From Entire Documents

Hi [PERSON Ted] and [PERSON Hill],
Just a reminder that the game move will need to be entered [TIME tonight]. We will need data on operations, rawmaterials ordering, and details of the bond to be sold.
[PERSON Hill]: I will be in the [LOCATION lobby] after the class at [TIME 9 pm]. how about we meet in the [LOCATION lobby] around that time (i.e when both our classes are over).
[PERSON Ted]: Let me know how you are going to provide the bond related input information. We can either meet in the [LOCATION lobby] around [TIME 5.30 pm] or you can e-mail me the info.
Thanks, [PERSON Ajay]

⇓

TIME	9 pm, 18th September
LOCATION	Lobby, Building NE43
PERSON	David Hill, Ajay Sinclair
TOPIC	data on operations...

TIME	5.30 pm, 18th September
LOCATION	Lobby, Building NE43
PERSON	Ted Jones, Ajay Sinclair
TOPIC	bond related input information

10TH DEGREE is a full service advertising agency specializing in direct and interactive marketing. Located in Irvine CA, 10TH DEGREE is looking for an Assistant Interactive Account Manager to help manage and coordinate interactive marketing initiatives for a marquee automotive account. Experience in online marketing, automotive and/or the advertising agency field is a plus.
Assistant Account Manager Responsibilities
Ensures smooth implementation of programs and initiatives Helps manage the delivery of projects and key client deliverables ...
Compensation: $50,000 – $80,000 Hiring Organization: 10TH DEGREE
Principals only. Recruiters, please don't contact this job poster. Please, no phone calls about this job! Please do not contact job poster about other services, products or commercial interests. Reposting this message elsewhere is NOT OK. this is in or around Orange County - Irvine

⇓

INDUSTRY	Advertising
POSITION	Assistant Account Manager
LOCATION	Irvine, CA
COMPANY	10th Degree
SALARY	$50,000 – $80,000

Relationship Extraction

[Miller et. al, 2000]

An example:

Donald M. Goldstein, a historian at the University of Pittsburgh ...

- Entity information to be extracted:
  - Named entity boundaries:
    Organizations, people, and locations
  - Person descriptors: "a historian at the University of Pittsburgh" refers to "Donald M. Goldstein"
- Entity relationships to be extracted:
  - Employer/Employee relations
    (e.g., *Goldstein* is employed at *University of Pittsburgh*)
  - Company/product relations
  - Organization/headquarters-location relation

Relationship Extraction: Annotation

Another example:

Nance, who is a paid consultant to ABC News, said ...

- The following information was annotated:
  - *Nance* as a person; *ABC News* as an organization; *a paid consultant to ABC News* as a descriptor
  - A *coreference* link between *Nance* and *a paid consultant to ABC News*
  - An *employer-relation* link from *a paid consultant to ABC News* to *ABC News*

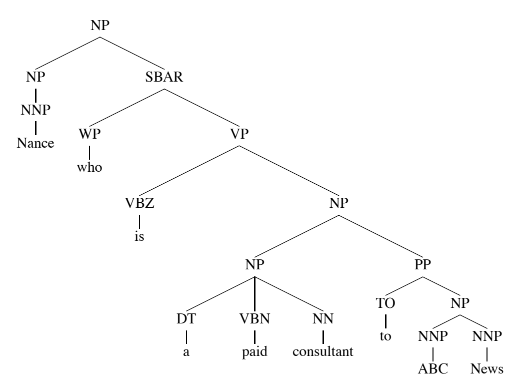
Next question: how can we build a model which recovers this information?

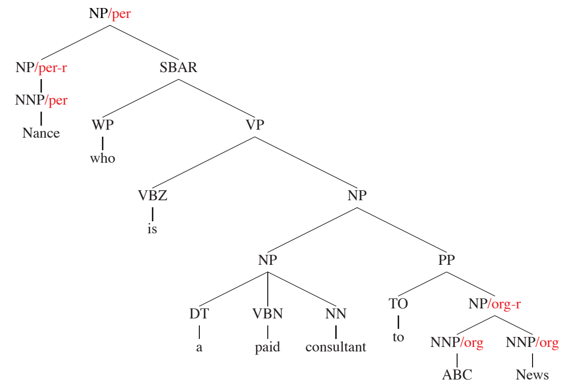
The Basic Approach

- Build a statistical parsing model which simultaneously recovers syntactic relation and the information extraction information

  To do this:

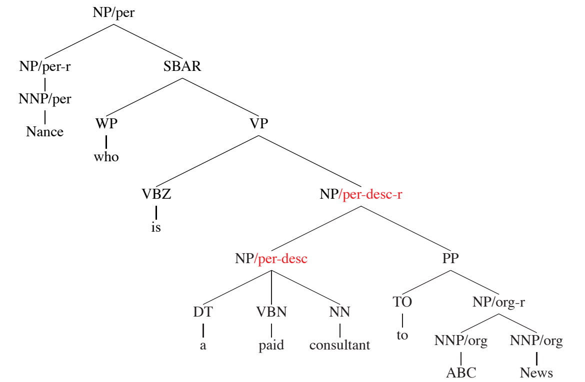
- Step 1: annotate training sentences for entities, descriptors, coreference links, and relation links
- Step 2: train a parser on the Penn treebank, and apply it to the new training sentences. **Force the parser to produce parses that are consistent with the entity/descriptor etc. boundaries**
- Step 3: enhance the parse trees to include the information extraction information (we'll come to this soon)
- Step 4: **re-train** the parser on the new training data, and with the new annotations





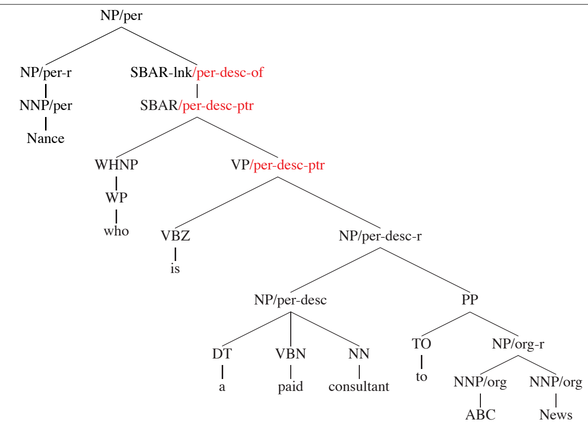
Add semantic tags showing named entities

org = organization, per = person, org-r = organization “reportable” (complete), per-r = person “reportable” (complete)



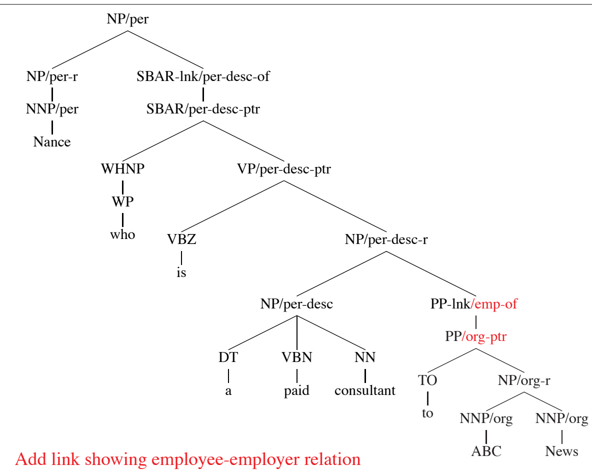
Add semantic tags showing descriptors

per-desc = person descriptor,
per-desc-r = person descriptor “reportable” (complete)



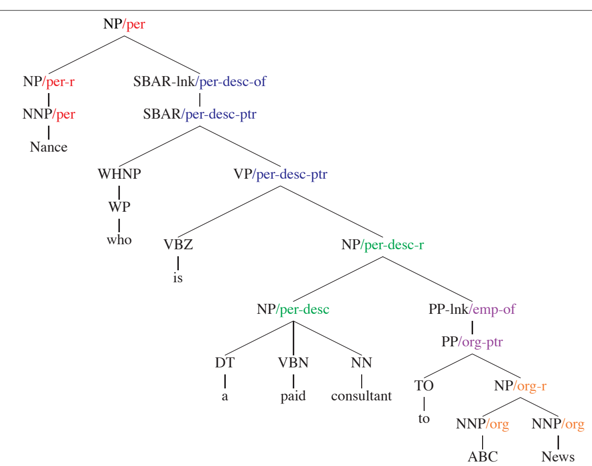
Add semantic tags showing link between “Nancy” and the descriptor

per-desc-of = person/descriptor link, per-desc-ptr = person/descriptor pointer



Add link showing employee-employer relation

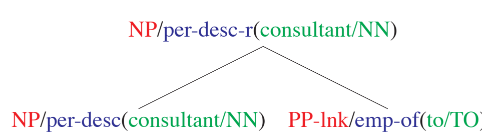
emp-of = employee-of link, emp-ptr = employee-of pointer



PERSON entity, PERSON descriptor link, DESCRIPTOR, EMPLOYER-OF relation, ORG entity

Building a Parser

- We now have context-free rules where each non-terminal in the grammar has
  - A syntactic category
  - A semantic label
  - A head-word/head-tag



- It’s possible to modify syntactic parsers to estimate rule probabilities in this case

Summary

- Goal: build a parser that recovers syntactic structure, named entities, descriptors, and relations
- Annotation: mark entity boundaries, descriptor boundaries, links between entities and descriptors
- Enriched parse trees: given annotation, and a parse tree, form a new **enriched** parse tree
- The statistical model: non-terminals now include syntactic category, semantic label, head word, head tag. Rule probabilities are estimated using similar methods to syntactic parsers
- Results: precision = 81%, recall = 64% in recovering relations (employer/employee, company/product, company/headquarters-location)

Main Points

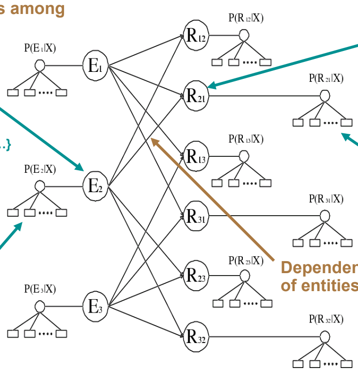
Co-reference
- How to cast as classification [Cardie]
- Measures of string similarity [Cohen]
- Scaling up [McCallum et al]

Relation extraction
- With augmented grammar [Miller et al 2000]
- **With joint inference [Roth & Yih]**
- Semi-supervised [Brin]

(1) Association with Graphical Models

[Roth & Yih 2002]

Capture arbitrary-distance dependencies among predictions.



Random variable over the class of entity #1, e.g. over {person, location,…}

Local language models contribute evidence to entity classification.

Random variable over the class of relation between entity #2 and #1, e.g. over {lives-in, is-boss-of,…}

Local language models contribute evidence to relation classification.

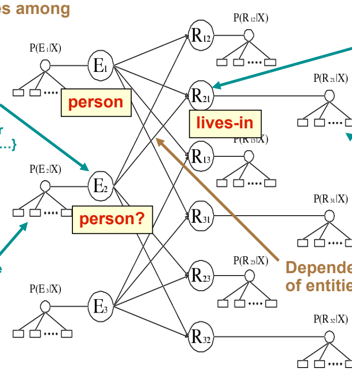
Dependencies between classes of entities and relations!

Inference with loopy belief propagation.

(1) Association with Graphical Models

[Roth & Yih 2002]

Also capture long-distance dependencies among predictions.



Random variable over the class of entity #1, e.g. over {person, location,…}

Local language models contribute evidence to entity classification.

Random variable over the class of relation between entity #2 and #1, e.g. over {lives-in, is-boss-of,…}

Local language models contribute evidence to relation classification.

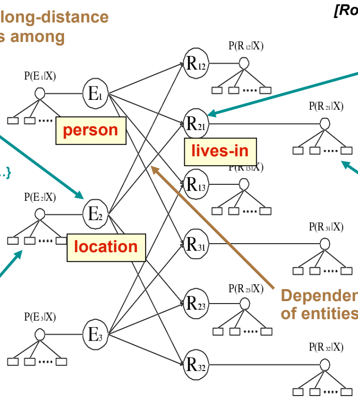
Dependencies between classes of entities and relations!

Inference with loopy belief propagation.

(1) Association with Graphical Models

[Roth & Yih 2002]

Also capture long-distance dependencies among predictions.



Random variable over the class of entity #1, e.g. over {person, location,…}

Local language models contribute evidence to entity classification.

Random variable over the class of relation between entity #2 and #1, e.g. over {lives-in, is-boss-of,…}

Local language models contribute evidence to relation classification.

Dependencies between classes of entities and relations!

Inference with loopy belief propagation.

Main Points

Co-reference
- How to cast as classification [Cardie]
- Measures of string similarity [Cohen]
- Scaling up [McCallum et al]

Relation extraction
- With augmented grammar [Miller et al 2000]
- With joint inference [Roth & Yih]
- **Semi-supervised [Brin]**

Partially Supervised Approaches to Relation Extraction

- Last lecture: introduced a partially supervised method for named entity classification
- Basic observation: "redundancy" in that either spelling or context of an entity is often sufficient to determine its type
- Lead to *cotraining* approaches, where two classifiers bootstrap each other from a small number of seed rules
- **Can we apply these kind of methods to relation extraction?**

From [Brin, 1998]

The World Wide Web provides a vast source of information of almost all types, ranging from DNA databases to resumes to lists of favorite restaurants. However, this information is often scattered among many web servers and hosts using many different formats. If these chunks of information could be extracted from the World Wide Web and integrated into a structure form, they would form an unprecedented source of information. It would include the largest international directory of people, the largest and most diverse databases of products, the greatest bibliography of academic works, and many other useful resources.

From [Brin, 1998]

For data we used a repository of 24 million web pages totalling 147 gigabytes. This data is part of the Stanford WebBase and is used for the Google search engine [BP], and other research projects. As a part of the search engine, we have built an inverted index of the entire repository.

The repository spans many disks and several machines. It takes a considerable amount of time to make just one pass over the data even without doing any substantial processing. Therefore, in these [sic] we only made passes over subsets of the repository on any given iteration.

[BP] Sergey Brin and Larry Page. Google search engine. `http://google.stanford.edu`

- From [Brin, 1998]:
  authors/book-titles, data = web data, seeds are

Isaac Asimov	The Robots of Dawn
David Brin	Startide Rising
James Gleik	Chaos: Making a New Science
Charles Dickens	Great Expectations
William Shakespeare	The Comedy of Errors

DIPRE [Brin, 1998]

A pattern is a 5 tuple:

- *Order*: author preceding title, or visa versa
- *URL-prefix*: a prefix of the URL of the page of the pattern
- *prefix*: up to 10 characters preceding the author/title pair
- *middle*: the characters between the author and title
- *suffix*: up to 10 characters following the author/title pair

DIPRE: Inducing Patterns from Data

- Find all instances of seeds on web pages.
  Basic question: how do we induce patterns from these examples?

- Answer = Following procedure:
  1. Group all occurrences together which have the same values for *order*, *middle*
  2. For any group: Set *url-prefix* to be longest common prefix of the group's URLs, *prefix* to be the longest common prefix of the group, *suffix* to be the longest common suffix
  3. For each group's pattern, calculate its specificity as

     $$spec(p) = n|middle||url\text{-}prefix||prefix||suffix|$$

     where n is the number of examples in the group, $|x|$ is the length of x in characters
  4. **If** specificity exceeds some threshold, include the pattern
  5. **Else If** all patterns occur on same webpage, reject the pattern
  6. **Else** create new sub-groups grouped by characters in the urls which is one past *url-prefix*, and repeat the procedure in step 2 for these new sub-groups.

The Overall Algorithm

1. Use the seed examples to label some data
2. Induce patterns from the labeled examples, using method described on the previous slide
3. Apply the patterns to data, to get a new set of author/title pairs
4. Return to step 2, and iterate

DIPRE: Inducing Patterns from Data

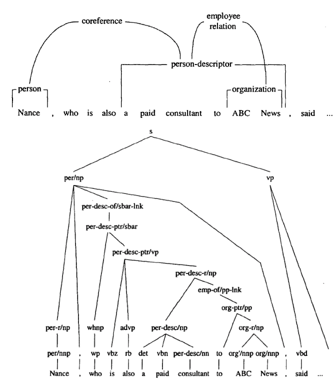
The patterns found in the first iteration:

`www.sff.net/locus/c.*`	<LI><B>*title*</B> by *author* (
`dns.city-net.com/lmann/awards/hugos/1984.html`	<i>*title*</i> by *author* (
`dolphin-upenn.edu/dcummins/texts/sf-award.htm`	*author* || *title* || (

- The 5 seeds produced 199 labeled instances, giving the 3 patterns above
- Applying the three patterns gave 4047 new book instances
- Searching 5 million web pages gave 3972 occurrences of these books
- This gave 105 patterns, 24 applied to more than one URL
- Applied to 2 million URLS produced 9369 unique (author,title) pairs
- Manual intervention: removed 242 "bogus" items where the author was "Conclusion"
- Final iteration: ran over 156,000 documents which contained the word "books"; induced 346 patterns, 15,257 (author,title) pairs

(1) Association using Parse Tree

Simultaneously POS tag, parse, extract & associate! ***[Miller et al 2000]***



Increase space of parse constitutes to include entity and relation tags

Notation	Description
c_h	head constituent category
c_m	modifier constituent category
X_p	X of parent node
t	POS tag
w	word

Parameters	e.g.
$P(c_h\|c_p)$	P(vp|s)
$P(c_m\|c_p,c_{hp},c_{m-1},w_p)$	P(per/np|s,vp,null,said)
$P(t_m\|c_m,t_h,w_h)$	P(per/nnp|per/np,vbd,
$P(w_m\|c_m,t_m,t_h,w_h)$	P(nance|per/np,per/nnp,vbd,said)

(This is also a great example of extraction using a tree model.)