CS290D – Advanced Data Mining

Instructor: Xifeng Yan
Computer Science
University of California at Santa Barbara

Entity & Relation Extraction

Instructor: Yang Li
Computer Science
University of California at Santa Barbara

with slides adapted from Chris Manning and Andrew McCallum

What will be covered in this class

- Named Entity Recognition
- Entity Linking
- Relation Extraction

What will be covered in this class

- Named Entity Recognition
- Entity Linking
- Relation Extraction

What is Named Entity Recognition?

- Named Entity Recognition is the process of finding and classifying named entities in text
  - Input: unstructured text
  - Output: named entities (along with their types) in the text.

The decision by the independent MP Andrew Wilkie to withdraw his support for the minority Labor government sounded dramatic but it should not further threaten its stability. When, after the 2010 election, Wilkie, Rob Oakeshott, Tony Windsor and the Greens agreed to support Labor, they gave just two guarantees: confidence and supply.

What is Named Entity Recognition?

- Named Entity Recognition is the process of finding and classifying named entities in text
  - Input: unstructured text
  - Output: named entities (along with their types) in the text.

The decision by the independent MP Andrew Wilkie to withdraw his support for the minority Labor government sounded dramatic but it should not further threaten its stability. When, after the 2010 election, Wilkie, Rob Oakeshott, Tony Windsor and the Greens agreed to support Labor, they gave just two guarantees: confidence and supply.

What is Named Entity Recognition?

- Named Entity Recognition is the process of finding and classifying named entities in text
  - Input: unstructured text
  - Output: named entities (along with their types) in the text.

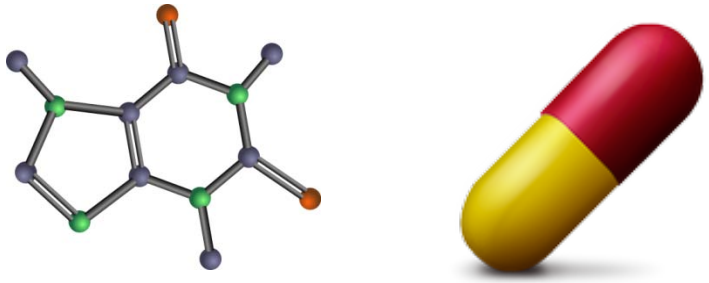
The decision by the independent MP Andrew Wilkie to withdraw his support for the minority Labor government sounded dramatic but it should not further threaten its stability. When, after the 2010 election, Wilkie, Rob Oakeshott, Tony Windsor and the Greens agreed to support Labor, they gave just two guarantees: confidence and supply.

Person Date Location Organization

Which Entities to Extract?

- General Entities:
  - Person
  - Location
  - Organization
  - Date
  - Car
  - Phone

- Domain-specific Entities:
  - Proteins
  - Virus
  - Drugs



How to Extract Entities?

- Hand written regular expressions
- ML-based sequence models
  - HMM
  - MEMM
  - CRF

Hand Written Regular Expressions

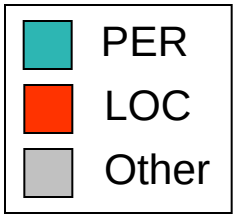
- For certain restricted, common types of entities in plain text, simple regex patterns usually work well.
  - Finding (US) phone numbers
  - (?:\(?[0-9]{3}\)?[-.])?[0-9]{3}[-.]?[0-9]{4}
  - Finding cars
  - match with the limited number of car brands/models
- For general entities like Person, Organization, it is hard to use regex patterns to extract

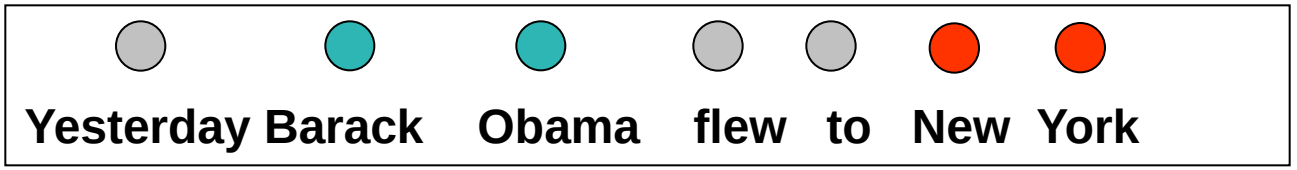
The ML sequence model approach to NER

- Training
  - Collect a set of representative training documents
  - Label each token for its entity class or other
  - Design feature extractors appropriate to the text and classes
  - Train a **sequence classifier** to predict the labels from the data
- Testing
  - Receive a set of testing documents
  - Run **sequence model** inference to label each token
  - Appropriately output the recognized entities

NER as Sequence Labeling

- Break the sentence into tokens, and classify each token with a label indicating what sort of entity it's part of:



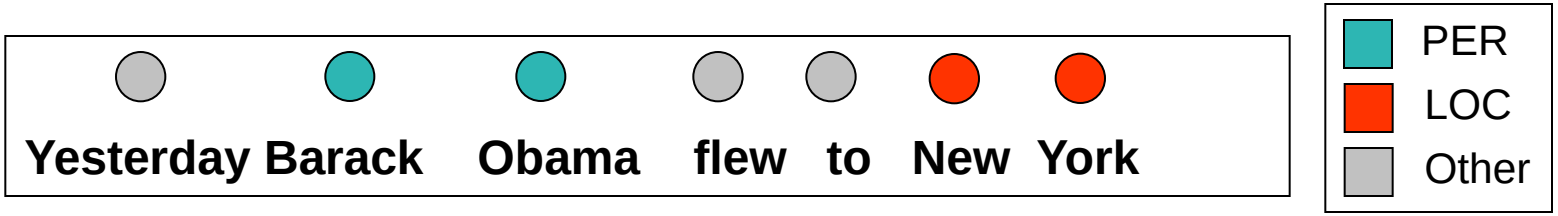


- Identify names based on the entity labels

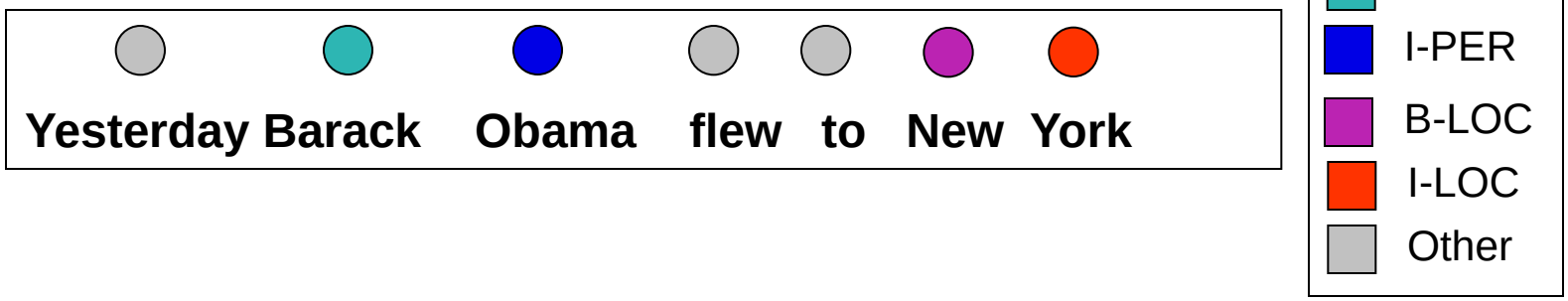
Person name: Barack Obama
Location name: New York

Encoding Scheme for Sequence Labeling

- IO Encoding:
  - Class Label + Other



- IOB Encoding
  - Begin Class Label + In Class Label + Other



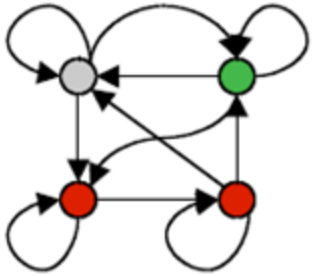
Encoding Scheme for Sequence Labeling

	IO encoding	IOB encoding
Fred	PER	B-PER
showed	O	O
Sue	PER	B-PER
Mengqiu	PER	B-PER
Huang	PER	I-PER
‘s	O	O
new	O	O
painting	O	O

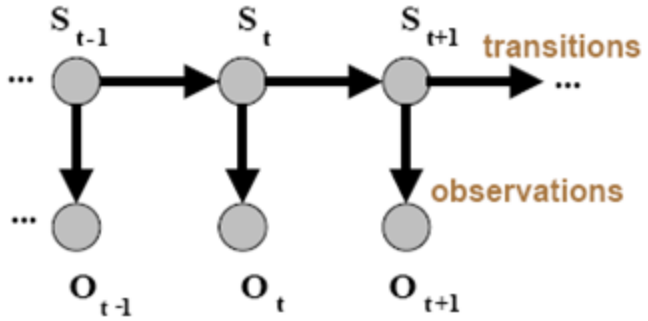
Hidden Markov Model (HMM)

HMMs are the standard sequence modeling tool in genomics, music, speech, NLP, …

Finite state model

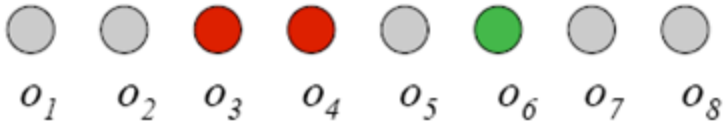


Graphical model



Generates:

State sequence
Observation sequence



$$P(\bar{s},\bar{o}) \propto \prod_{t=1}^{|\bar{o}|} P(s_t \mid s_{t-1}) P(o_t \mid s_t)$$

Parameters: for all states $S=\{s_1, s_2, \ldots\}$

- **Start state probabilities: $P(s_t)$**
- **Transition probabilities: $P(s_t|s_{t-1})$**
- **Observation (emission) probabilities: $P(o_t|s_t)$**

Usually a multinomial over atomic, fixed alphabet

Training:

- **Maximize probability of training observations**

Hidden Markov Model (HMM)

- Estimate state transition probabilities based on tag bigram and unigram statistics in the labeled data.

$$a_{ij} = \frac{C(q_t = s_i, \mathrm{q}_{\mathrm{t+1}} = s_j)}{C(q_t = s_i)}$$

- Estimate the observation probabilities based on tag/word co-occurrence statistics in the labeled data.

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

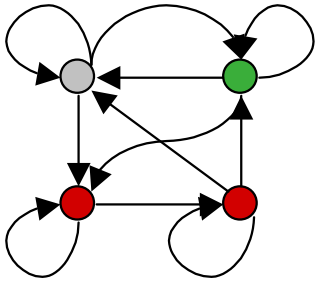
- Use appropriate smoothing if training data is sparse.

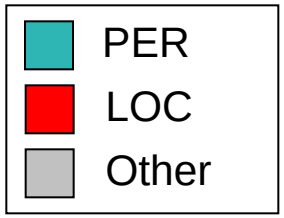
NER with Hidden Markov Model

Given a sequence of observations:

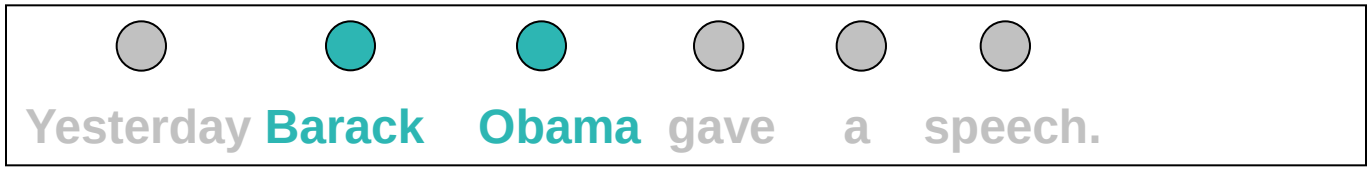
Yesterday Barack Obama gave a speech.

and a trained HMM:





Find the most likely state sequence: $\arg\max_{\vec{s}} P(\vec{s}, \vec{o})$



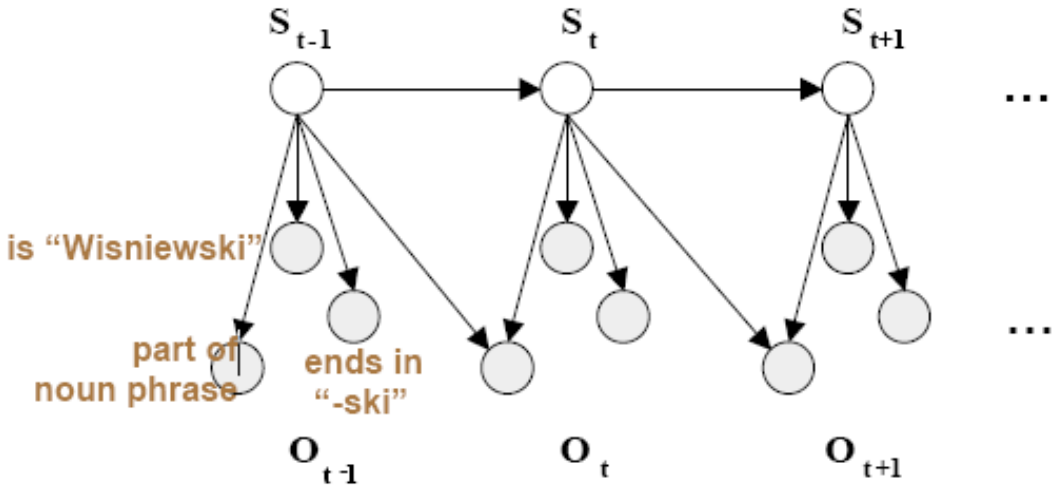
Any words said to be generated by the designated "person name" state extract as a person name:

Person name: Barack Obama

Atomic View of Words in HMM is Not Enough

- Would like to have richer representation of text: many arbitrary, overlapping features of the words.

identity of word
ends in "-ski"
is capitalized
is part of a noun phrase
is in a list of city names
is under node X in WordNet
is in bold font
is indented
is in hyperlink anchor
last person name was female
next two words are "and Associates"



Atomic View of Words in HMM is Not Enough

- These arbitrary features are not independent.
  - Multiple levels of granularity (chars, words, phrases)
  - Multiple dependent modalities (words, formatting, layout)
- Possible solutions:
  - ✗ Model dependencies (we may lose tractability!)
  - ✗ Ignore dependencies (but they are not independent!)
  - ✓ Conditional Sequence Models

Conditional Sequence Model

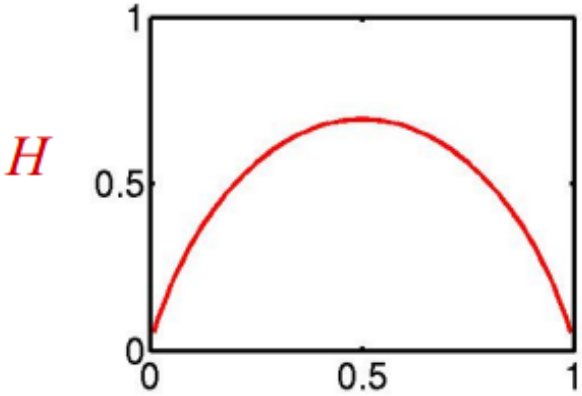
- Conditional model p(y|x).
  - Do not waste effort modeling p(x), since x is given anyway.
  - Allows more complicated input features, since we do not need to model dependencies between them.

- Example feature functions f(x,y):
  - f1(x,y) = { word is Boston & y=Location }
  - f2(x,y) = { first letter capitalized & y=Name }
  - f3(x,y) = { x is an HTML link & y=Location}

Entropy

- Entropy: the uncertainty of a distribution.
- Quantifying uncertainty ("surprise"):
  - Event x
  - Probability p_x
  - "Surprise" $\log(1/p_x)$
- Entropy: expected surprise (over p):

$$\mathrm{H}(p) = E_p\left[\log_2 \frac{1}{p_x}\right] = -\sum_x p_x \log_2 p_x$$



A coin-flip is most uncertain for a fair coin.

Maximum Entropy

- What do we want from a distribution?
  - Minimize commitment = maximize entropy.
  - Resemble some reference distribution (data).
- Solution: maximize entropy H, subject to feature-based constraints
- Adding constraints (features):
  - Lowers maximum entropy
  - Raises maximum likelihood of data
  - Brings the distribution further from uniform
  - Brings the distribution closer to data

Maximum Entropy Classifier

- Same as multinomial logistic regression (thus an exponential model for n-way classification)
- Features are constraints on the model.

$$p(y|x) = \frac{1}{Z(x)} exp\left(\sum_{i=1}^{n} \lambda_i f_i(x,y)\right)$$

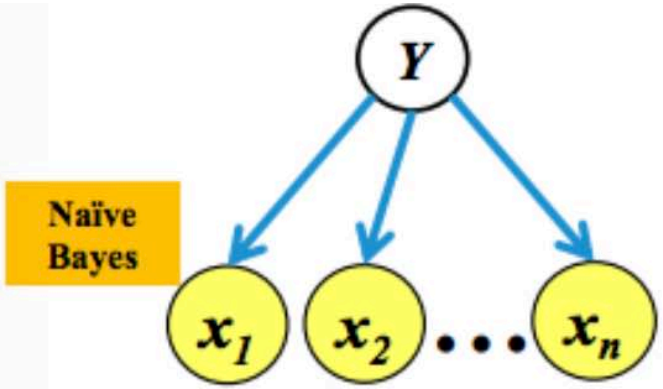
- Of all possible models, we choose that which maximizes the entropy of all models that satisfy these constraints.

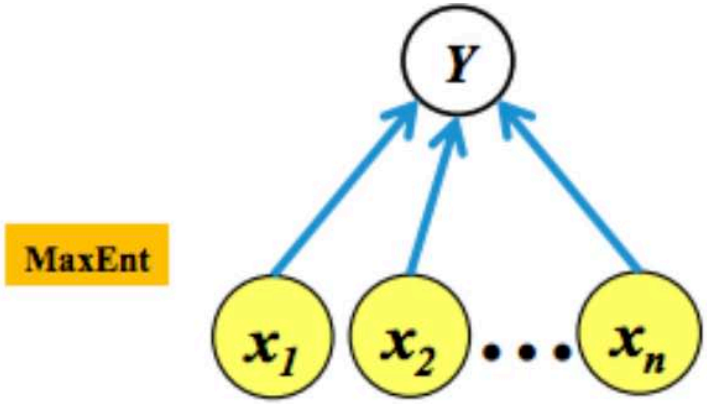
Maximum Entropy Classifier

- To identify the most probable label *y* for a particular context *x*, we calculate

$$
\begin{aligned}
\hat{y} &= argmax_y\ p(y|x) \\
&= argmax_y\ log\ p(y|x) \\
&= argmax_y\ log(\frac{1}{Z(x)} exp\left(\sum_{i=1}^{n} \lambda_i f_i(x,y)\right)) \\
&= argmax_y\ log(\frac{1}{Z(x)}) + log(exp\left(\sum_{i=1}^{n} \lambda_i f_i(x,y)\right)) \\
&= argmax_y \sum_{i=1}^{n} \lambda_i f_i(x,y)
\end{aligned}
$$

Maximum Entropy vs. Naïve Bayes





Naïve Bayes Classifier	Maximum Entropy Classifier
"Generative" models ➔ **p(input | output)** ➔ For instance, for text categorization, P(words | category) ➔ Waste energy on generating input (which we don't need to generate during test)	"Discriminative" models ➔ **p(output | input)** ➔ For instance, for text categorization, P(category | words) ➔ Focusing only on predicting the output
➔ **Independent assumption among input variables:** Given the category, each word is generated independently from other words **(too strong assumption in reality!)** ➔ Cannot incorporate arbitrary/redundant/overlapping features	➔ By conditioning on the entire input, we don't need to worry about the independent assumption among input variables ➔ **Can incorporate arbitrary features** ➔ **Can handle redundant and overlapping features**

Maximum Entropy Markov Model (MEMM)

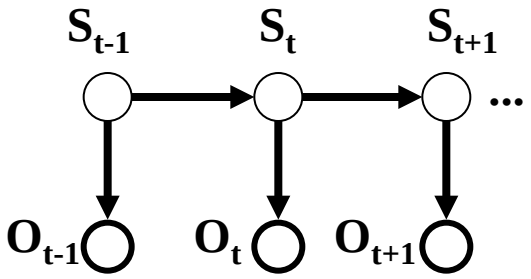
- Combines the advantages of maximum entropy and sequence model

- Use a series of maximum entropy classifiers that know the previous label.

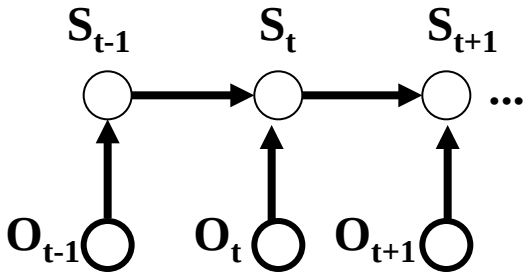
$$\Pr(s \mid o) = \prod_i \Pr(s_i \mid s_{i-1}, o_i)$$

HMM *vs.* MEMM

$$\Pr(s,o) = \prod_i \Pr(s_i \mid s_{i-1}) \Pr(o_i \mid s_i)$$



$$\Pr(s \mid o) = \prod_i \Pr(s_i \mid s_{i-1}, o_i)$$



replace generative model in HMM with a MaxEnt model, where state depends on observations and previous state

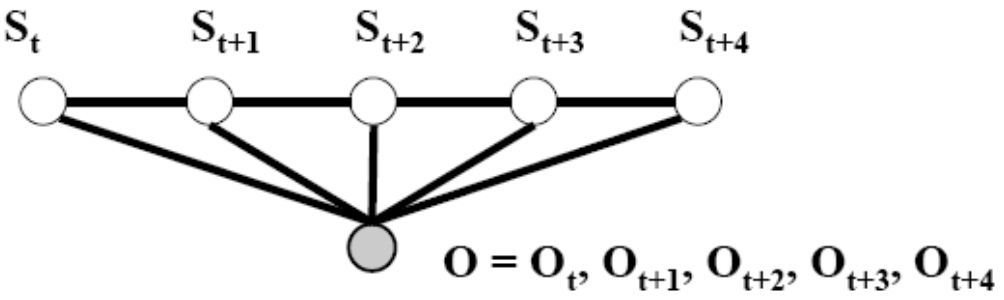
HMM *vs.* MEMM

HMM	MEMM
"Generative" models → joint probability **p(words, tags)** →"generate" input (in addition to tags) → but we need to predict tags, not words!	"Discriminative" or "Conditional" models → conditional probability **p(tags | words)** →"condition" on input → Focusing only on predicting tags
Probability of each slice = emission * transition = p(word_i | tag_i) * p(tag_i | tag_i-1) = → Cannot incorporate long distance features	Probability of each slice = p(tag_i | tag_i-1, word_i) or p(tag_i | tag_i-1, all words) → Can incorporate long distance features

Conditional Random Field (CRF)

- A conditionally-trained, undirected graphical model

- In CRF, the graph structure is often a linear chain, where the state sequence (S) connects S_{t-1} and S_t, and each state S_i is dependent on the whole observation sequence (O).



Markov on *s*, conditional dependency on *o*.

$$P(\vec{s} \mid \vec{o}) \propto \frac{1}{Z_{\vec{o}}} \prod_{t=1}^{|\vec{o}|} \exp\left(\sum_j \lambda_j f_j(s_t, s_{t-1}, \vec{o}, t) \right)$$

MEMM vs. CRF

MEMM	CRF
Directed graphical model	Undirected graphical model
"Discriminative" or "Conditional" models ➔ conditional probability **p(tags | words)**	
Probability is defined for each slice = P (tag_i | tag_i-1, word_i) or p (tag_i | tag_i-1, all words)	Instead of probability, ***potential (energy function)*** is defined for each slide = ϕ (tag_i, tag_i-1) * ϕ (tag_i, word_i) or ϕ(tag_i, tag_i-1, all words) * ϕ (tag_i, all words)
➔ Can incorporate long distance features	
locally normalized	globally normalized

Popular Features Used in NER

Feature	Explanation
Lexical items	The token to be labeled
Stemmed lexical items	Stemmed version of the target token
Shape	The orthographic pattern of the target word
Character affixes	Character-level affixes of the target and surrounding words
Part of speech	Part of speech of the word
Syntactic chunk labels	Base-phrase chunk label
Gazetteer or name list	Presence of the word in one or more named entity lists
Predictive token(s)	Presence of predictive words in surrounding text
Bag of words/Bag of N-grams	Words and/or *N*-grams occurring in the surrounding context

Shape	Example
Lower	cummings
Capitalized	Washington
All caps	IRA
Mixed case	eBay
Capitalized character with period	H.
Ends in digit	A9
Contains hyphen	H-P

State-of-the-art Performance for NER

- Performance depends on the entity types and text genres
- On Person, Location, Organization: Over 0.9 F1 Score
- Popular NER Systems
  - Stanford NER
  - UIUC NER
  - OpenNLP NER

What will be covered in this class

- Named Entity Recognition
- Entity Linking
- Relation Extraction

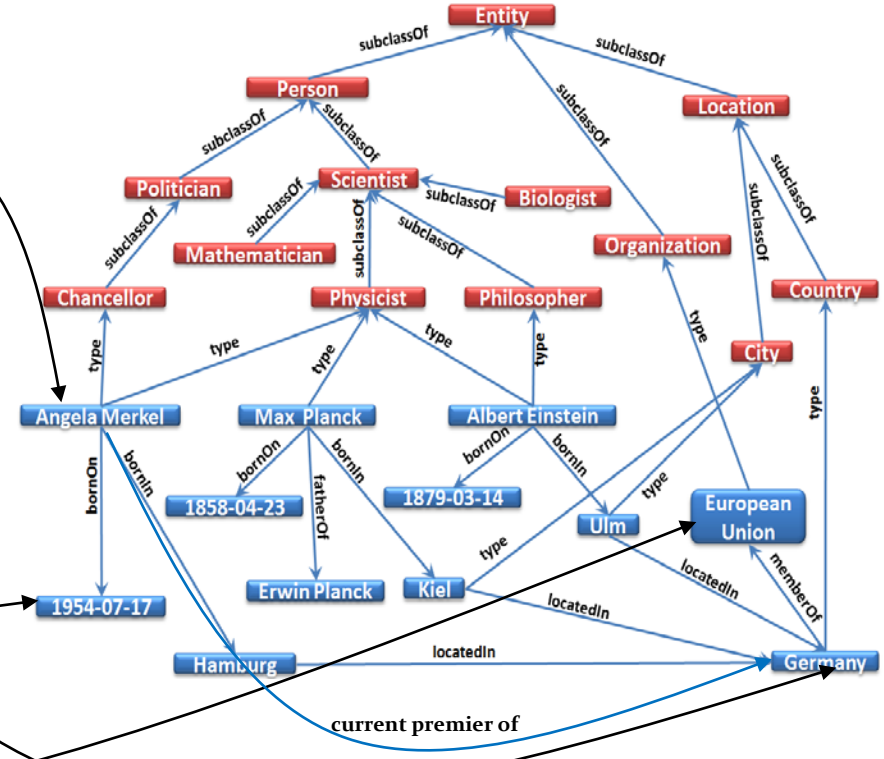
What is Entity Linking?

- Entity Linking is the process of associating an entity mentioned in a text to an entry, representing that entity, in a knowledge base (*e.g.* Wikipedia)
  - Input: a textual named entity mention *m*, along with the unstructured text in which it appears.
  - Output: the corresponding entity *e* in the knowledge base

- If the matching entity *e* for entity mention *m* does not exist in the knowledge base, return *NULL* for *m*

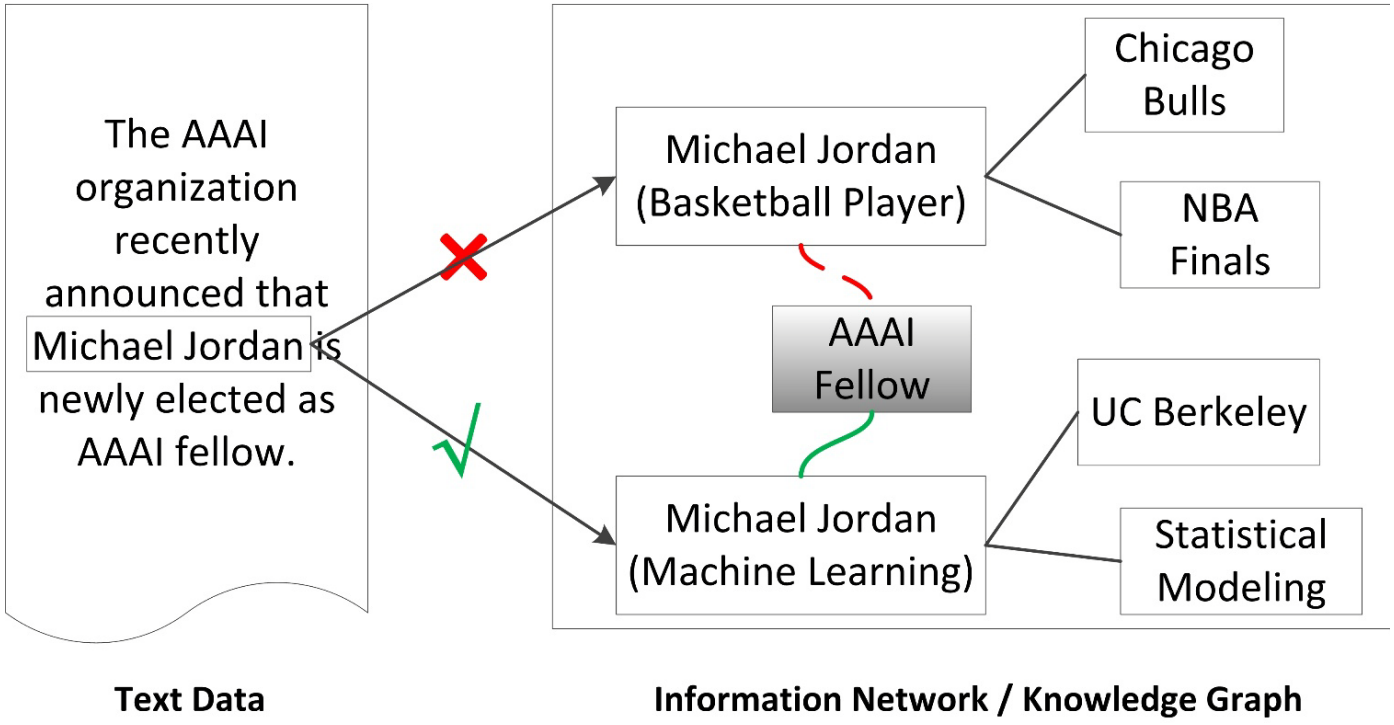
What is Entity Linking?

Mrs. Merkel, born on 1954-07-17, is the current premier of Germany, which is an important member of EU.



Why we need Entity Linking?

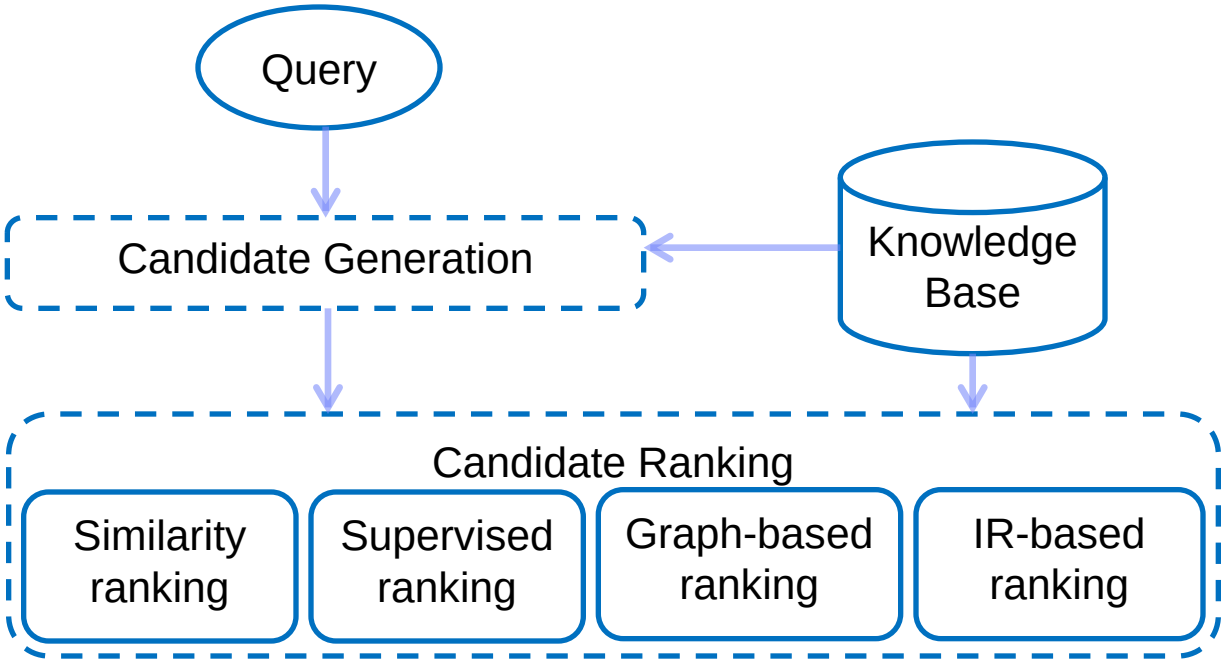
- Entity Linking (EL) is an important component in constructing and enriching high-quality knowledge graph from unstructured text



Challenges of Entity Linking

- Name Variations:
  - Same entity can have different name variations.
- Name Ambiguity:
  - Same name mention can refer to different entities.
- Entity Linking should handle both Name Variations and Name Ambiguity

General Entity Linking Framework



Candidates Generation

- Build a dictionary that contains vast amount of information about the surface forms of entities
  - including name variations, abbreviations, confusable names, spelling variations, nicknames, *etc.*
- Leverage the four structures of Wikipedia
  - Entity pages
  - Redirect pages
  - Disambiguation pages
  - Hyperlinks in Wikipedia articles

Candidates Generation

- Leverage the four structures of Wikipedia
  - Entity pages

Michael Jordan

From Wikipedia, the free encyclopedia

For other people named Michael Jordan, see Michael Jordan (disambiguation).

"Air" Jordan" redirects here. For the shoe and athletic wear company, see Air Jordan.

Michael Jeffrey Jordan (born February 17, 1963), also known by his initials, **MJ**,[2] is an
He played 15 seasons in the National Basketball Association (NBA) for the Chicago Bulls ar
basketball player of all time."[3] Jordan was one of the most effectively marketed athletes of

After a three-season career at the University of North Carolina at Chapel Hill, where he was
quickly emerged as a league star, entertaining crowds with his prolific scoring. His leaping a
Jordan" and "**His Airness**". He also gained a reputation for being one of the best defensiv
titles in 1992 and 1993, securing a "three-peat". Although Jordan abruptly retired from basl
led them to three additional championships in 1996, 1997, and 1998, as well as an NBA-rec
more NBA seasons from 2001 to 2003 as a member of the Wizards.

Michael I. Jordan

From Wikipedia, the free encyclopedia

For other people named Michael Jordan, see Michael Jordan (disambig

Michael Irwin Jordan (born 1956) is an American scientist, Professor at t

Contents [hide]

1 Biography
2 Work
3 References
4 External links

Biography [edit]

Jordan was born in Ponchatoula, Louisiana,[5] to a working class family, and
State University and his PhD in Cognitive Science in 1985 from the Univers
in the 1980s.

Jordan is currently a full professor at the University of California, Berkeley v

Candidates Generation

- Leverage the four structures of Wikipedia
  - Redirect pages

Lyndon B. Johnson

From Wikipedia, the free encyclopedia
(Redirected from LBJ)

"LBJ" redirects here. For other uses, see LBJ (disambiguation).

Lyndon Baines Johnson (/ˈlɪndən ˈbeɪnz ˈdʒɒnsən/) (1908–1973), often r
(1961–1963). Johnson, a Democrat from Texas, served as a United States F
Senate Minority Leader and two as Senate Majority Whip. He campaigned ur
election. After their election, Johnson succeeded Kennedy following the latte
by a large margin over Barry Goldwater. He is one of four people[1] who serv

Johnson was strongly supported by the Democratic Party, and as President

Candidates Generation

- Leverage the four structures of Wikipedia
  - Disambiguation pages

Michael Jordan (disambiguation)

From Wikipedia, the free encyclopedia

Michael Jordan (born 1963) is an American basketball player.

Michael Jordan may also refer to:

- Michael Jordan (mycologist), English mycologist
- Michael Jordan (footballer) (born 1986), English goalkeeper (Arsenal, Chesterfield, Lewes)
- Michael Jordan (insolvency baron) (born 1931), English businessman
- Mike Jordan (born 1958), English racing driver
- Mike Jordan (baseball) (1863–1940), baseball player
- Michael Jordan (Irish politician), Irish Farmers' Party TD from Wexford, 1927–1932
- Michael B. Jordan (born 1987), American actor
- Michael I. Jordan (born 1957), American researcher in machine learning and artificial intelligence
- Michael H. Jordan (1936–2010), American executive for CBS, PepsiCo, Westinghouse
- Michael-Hakim Jordan (born 1977), American professional basketball player
- Michal Jordan (born 1990), Czech ice hockey player
- "Michael Jordan", a song by Kendrick Lamar featuring ScHoolboy Q on the album Overly Dedicated'

Candidates Generation

- Leverage the four structures of Wikipedia
  - Hyperlinks

In the summer of 1984, the Bulls had the third pick of the 1984 NBA Draft, after Houston and Portland. The Rockets selected Hakeem Olajuwon, the Blazers picked Sam Bowie and the Bulls chose shooting guard M. Jordan. The team, with new management in owner Jerry Reinsdorf and general manager Jerry Krause, decided to rebuild around MJ...

Michael Jordan

From Wikipedia, the free encyclopedia

For other people named Michael Jordan, see Michael Jordan (disambiguation).

"Air" Jordan" redirects here. For the shoe and athletic wear company, see Air Jordan.

Michael Jeffrey Jordan (born February 17, 1963), also known by his initials, **MJ**,[2] is an
He played 15 seasons in the National Basketball Association (NBA) for the Chicago Bulls ar
basketball player of all time."[3] Jordan was one of the most effectively marketed athletes of

After a three-season career at the University of North Carolina at Chapel Hill, where he was
quickly emerged as a league star, entertaining crowds with his prolific scoring. His leaping a
Jordan" and "**His Airness**". He also gained a reputation for being one of the best defensiv
titles in 1992 and 1993, securing a "three-peat". Although Jordan abruptly retired from bask
led them to three additional championships in 1996, 1997, and 1998, as well as an NBA-rec
more NBA seasons from 2001 to 2003 as a member of the Wizards.

Candidates Generation

- For each mention *m*
  - Search it in the field of surface forms
  - If a hit is found, add all target entities of that surface form m to the set of candidate entities E_m

Surface form	Target entity	Count
Microsoft Corporation	Microsoft	16
Michael Jordan	Michael Jordan	65
	Michael I. Jordan	10
	Michael Jordan (mycologist)	7
	Michael Jordan (footballer)	3
	...	...
New York	New York City	121
	New York (magazine)	12
	New York (film)	7
	"New York" (Eskimo Joe song)	5
	...	...

Candidates Ranking - Features

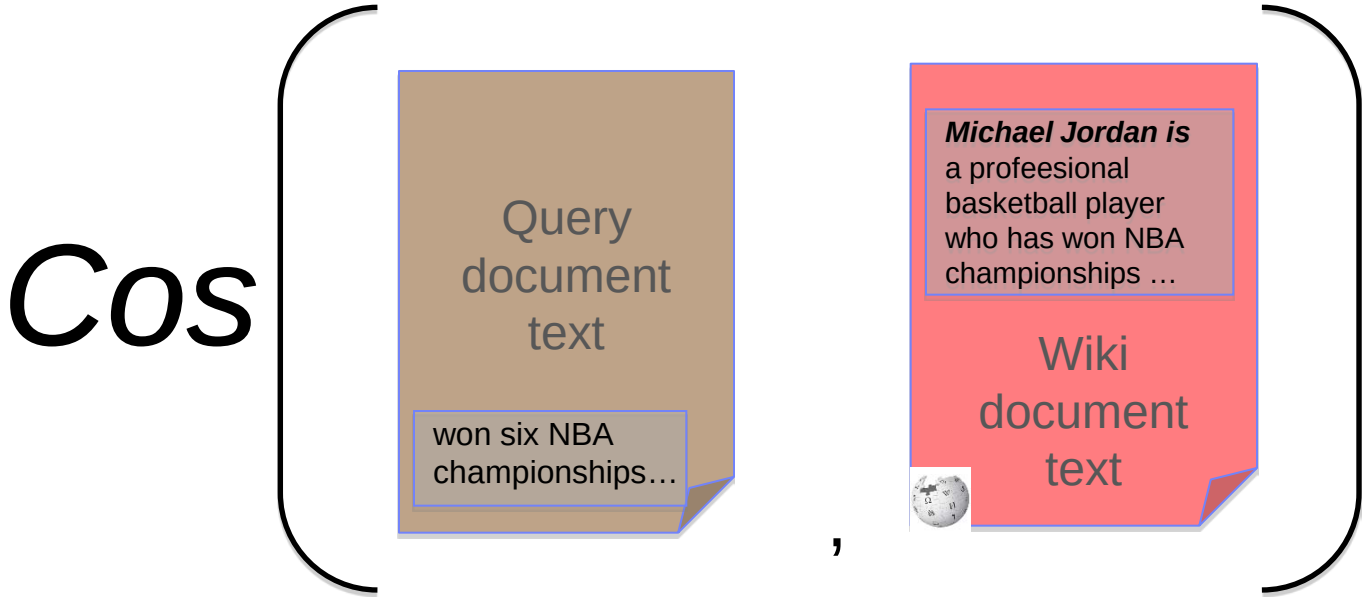
- Prior Popularity
  - Assume the most prominent entity for a given mention is the most probable underlying entity for that mention.

$$Popularity(m \Rightarrow t) = \frac{count(m \rightarrow t)}{\sum_{t' \in W} count(m \rightarrow t')}$$

t	P(t|"Michael Jordan")
Michael J. Jordan	.76
Michael I. Jordan	.121
Michael B. Jordan	.092
Michael H. Jordan	.00186

Candidates Ranking - Features

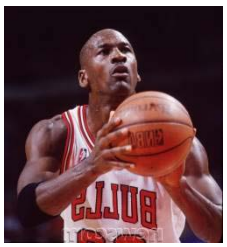
- Context Similarity
  - Define a similarity measure between the text around the entity mention and the document describing the referent entity in the knowledge base.



Candidates Ranking - Features

- Topical Coherence
  - Define a topical/semantic coherence measure between the mention's referent entity and other entities within the same context

Michael Jordan co-invented the *Latent Dirichlet Allocation* model with *David Blei* and *Andrew Ng*

TC   >> TC  

Candidates Ranking – Wikifier (L. Ratinov *et al.* ACL'11)

- A collective Entity Linking method
  - Linking all the mentions in the query simultaneously
  - Making use of global semantic coherence in the query
- Global optimization $\Gamma^* \approx \arg\max_{\Gamma} \sum_{i=1}^{N} [\phi(m_i, t_i) + \sum_{t_i \in \Gamma, t_j \in \Gamma'} \psi(t_i, t_j)]$

- Γ is a solution to the problem
  - A set of pairs (m,t)
- m: a mention in the query
- t: a candidate entity

- Invent a surrogate solution Γ'
  - Γ' is the local optimum for each mention t_j.
- Evaluate the coherence based on pair-wise coherence scores $\Psi(t_i,t_j)$

Candidates Ranking – Wikifier (L. Ratinov *et al.* ACL'11)

- Local features $\varphi(m_i,t_i)$
  - Prior Popularity: $P(t_i|m_i)$
  - Context Similarity: TF-IDF weighted cosine similarity
- Global features $\Psi(t_i,t_j)$
  - Normalized Google Distance

$$NGD(L_1, L_2) = \frac{Log(Max(|L_1|, |L_2|)) - Log(|L_1 \cap L_2|)}{Log(|W|) - Log(Min(|L_1|, |L_2|))}$$

  - Pointwise Mutual Information

$$PMI(L_1, L_2) = \frac{|L_1 \cap L_2|/|W|}{|L_1|/|W||L_2|/|W|}$$

Candidates Ranking – Wikifier (L. Ratinov *et al.* ACL'11)

Cognitive Computation Group ▸ Demos ▸ Wikifier

Wikifier Demo

fewer concepts — more concepts

wikify! clear

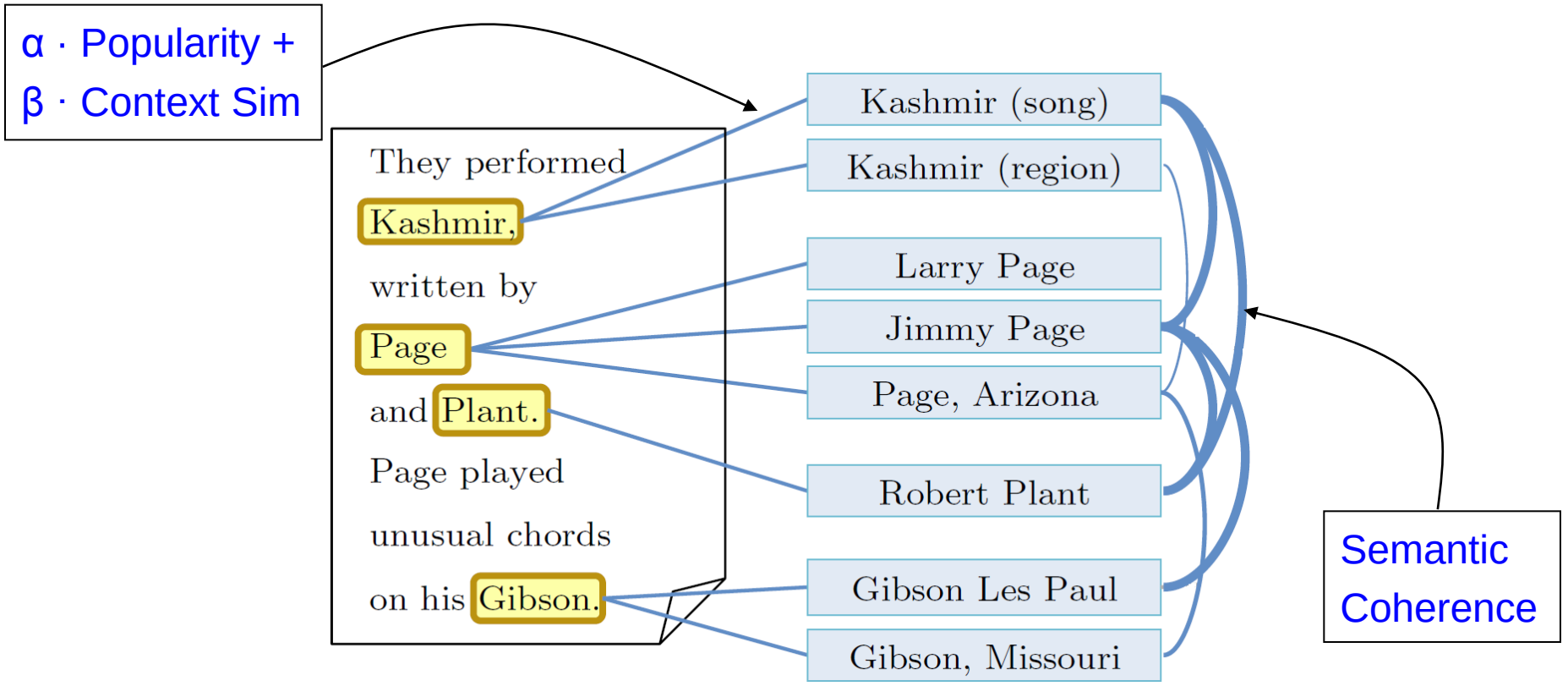
* If you wish to cite this work, please cite the following publications: (1) Retinov et. al. and (2) Cheng and Roth.

The Chiefs didn't trade for Alex Smith this offseason solely because they wanted a smart game manager who wouldn't kill their offense with turnovers. They acquired him because they needed a quarterback who knows how to win. Sometimes that requires him to do what he's done for most of this season: throw the safe pass, make the key play when necessary and use his feet to keep the chains moving when his arm can't get the job done. These days it means Smith has to show people more of what he revealed in Sunday's 41-38 loss to San Diego -- that he can elevate his game when his team is in dire straits.

http://cogcomp.cs.illinois.edu/demo/wikify/?id=25

Candidates Ranking – AIDA (J. Hoffart *et al.* EMNLP'11)

- Also a collective Entity Linking method
  - Based on Mention-Entity Graph

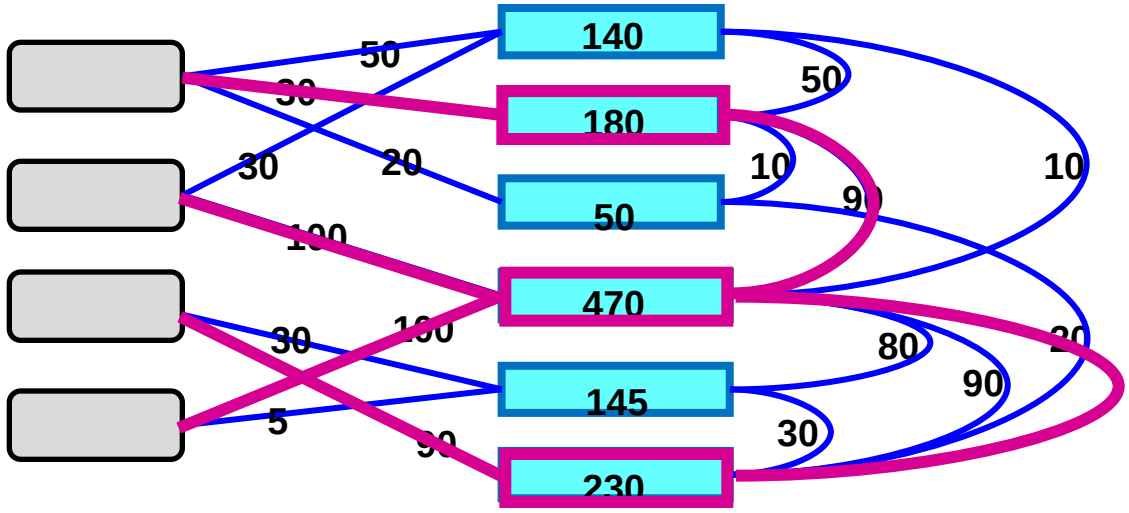


Candidates Ranking – AIDA (J. Hoffart *et al.* EMNLP'11)

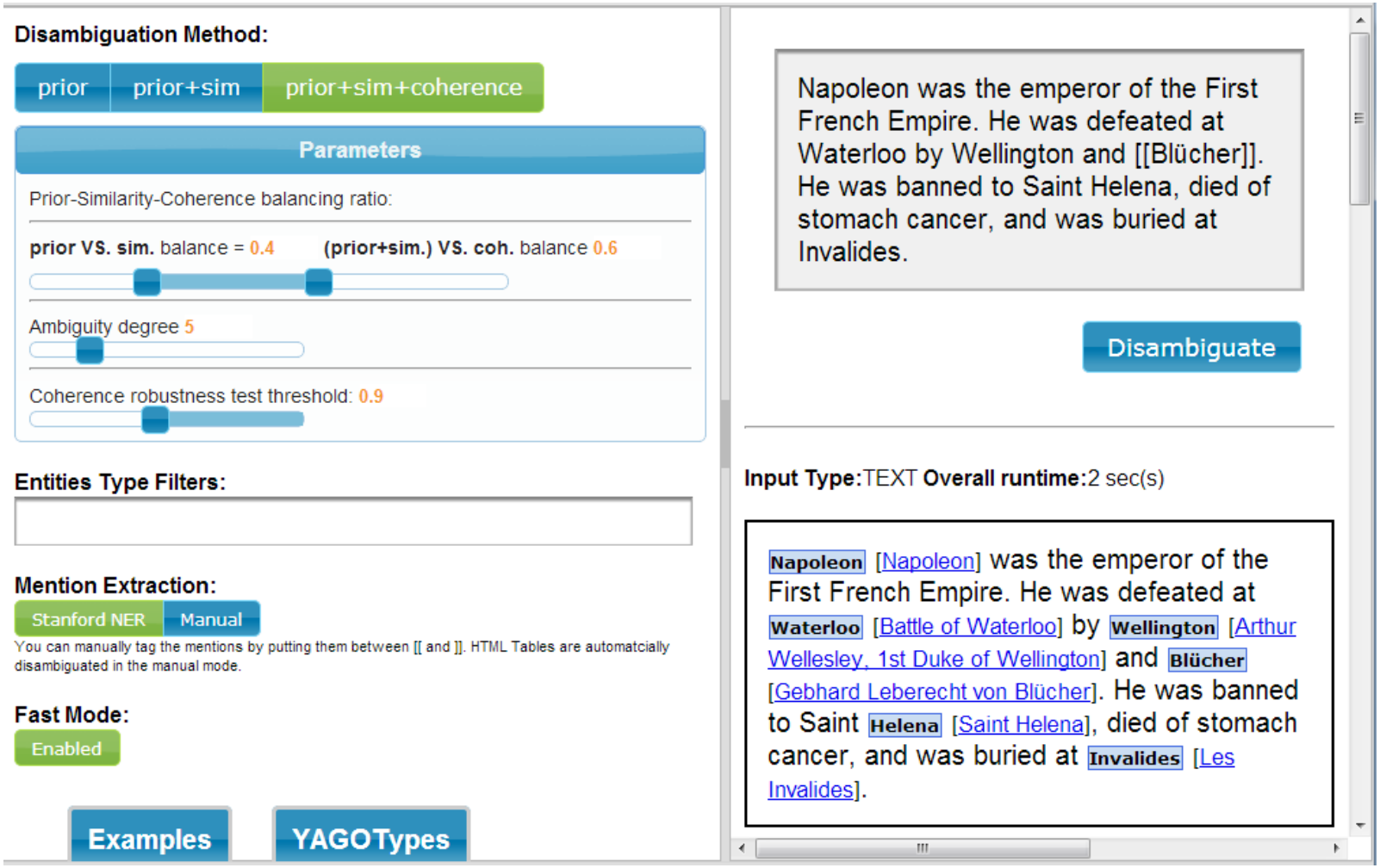
- Goal: compute dense subgraph to maximize weighted degree among entity nodes such that **each m is connected to exactly one e**
  - weighted degree: total weight of a node's incident edges
- NP-hard → need approximation algorithm
- Greedy approximation:
  - iteratively remove weakest entity and its edges

Candidates Ranking – AIDA (J. Hoffart *et al.* EMNLP'11)

- Greedy approximation:
  - iteratively remove weakest entity and its edges



Candidates Ranking – AIDA (J. Hoffart et al. EMNLP'11)



https://gate.d5.mpi-inf.mpg.de/webaida/

Other Available Entity Linking Systems

- P. Ferragina, U. Scaella: CIKM 2010
  - http://tagme.di.unipi.it/
- R. Isele, C. Bizer: VLDB 2012
  - http://spotlight.dbpedia.org/demo/index.html
- Reuters Open Calais
  - http://viewer.opencalais.com/
- Alchemy API
  - http://www.alchemyapi.com/api/demo.html
- S. Kulkarni, A. Singh, G. Ramakrishnan, S. Chakrabarti: KDD 2009
  - http://www.cse.iitb.ac.in/soumen/doc/CSAW/
- D. Milne, I. Witten: CIKM 2008
  - http://wikipedia-miner.cms.waikato.ac.nz/demos/annotate/

What will be covered in this class

- Named Entity Recognition
- Entity Linking
- Relation Extraction

What is Relation Extraction?

- Relation extraction is the process of extracting semantic relations among concepts in text
  - Input: A set of recognized concepts C, along with the unstructured text in which they appears.
  - Output: The semantic relations R among those concepts.

What is Relation Extraction?

CHICAGO (AP) — Citing high fuel prices, United Airlines said Friday it has increased fares by $6 per round trip on flights to some cities also served by lower-cost carriers. American Airlines, a unit of AMR, immediately matched the move, spokesman Tim Wagner said. United, a unit of UAL, said the increase took effect Thursday night and applies to most routes where it competes against discount carriers, such as Chicago to Dallas and Atlanta and Denver to San Francisco, Los Angeles and New York.

Subject	Relation	Object
American Airlines	subsidiary	AMR
Tim Wagner	employee	American Airlines
United Airlines	subsidiary	UAL

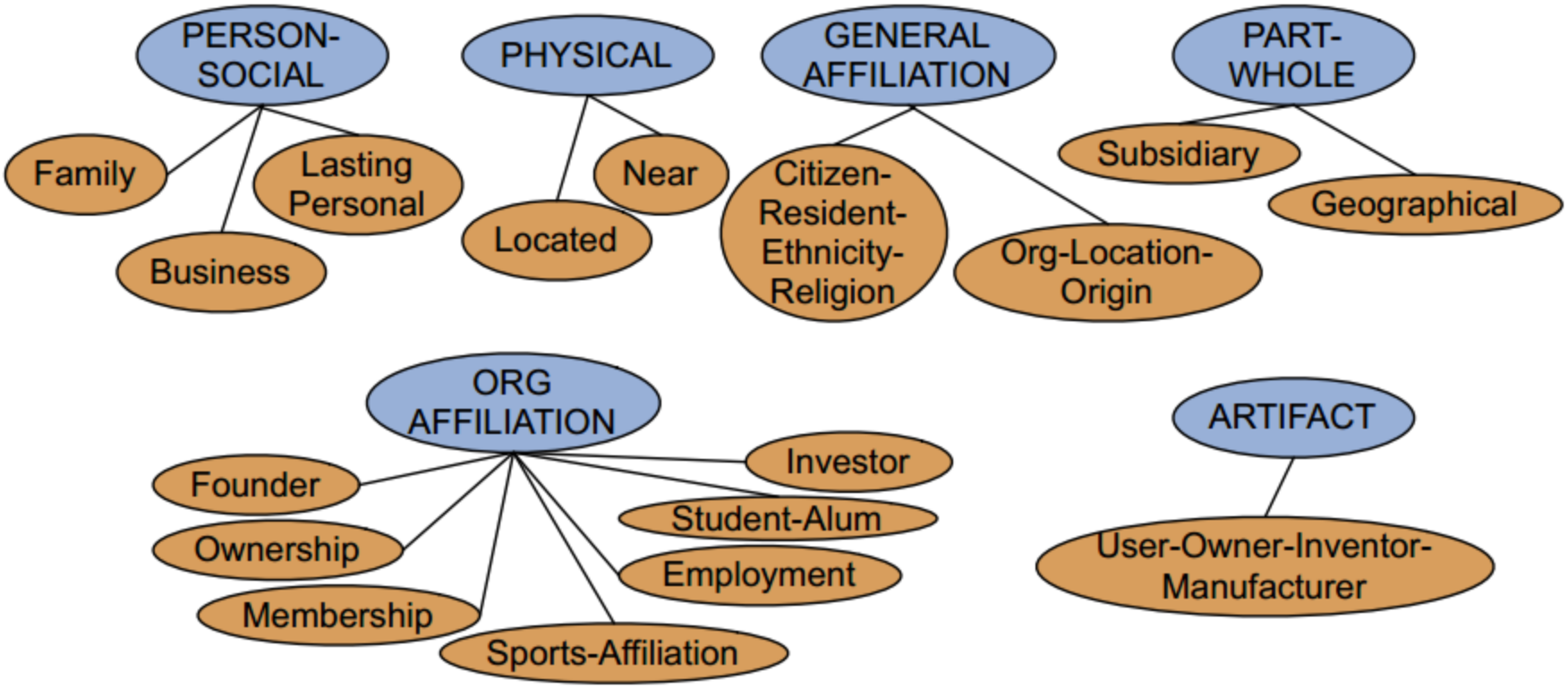
Which Relations to Extract?

- For generic news texts

Relations		Examples	Types
Affiliations			
	Personal	*married to, mother of*	PER → PER
	Organizational	*spokesman for, president of*	PER → ORG
	Artifactual	*owns, invented, produces*	(PER | ORG) → ART
Geospatial			
	Proximity	*near, on outskirts*	LOC → LOC
	Directional	*southeast of*	LOC → LOC
Part-Of			
	Organizational	*a unit of, parent of*	ORG → ORG
	Political	*annexed, acquired*	GPE → GPE

Which Relations to Extract?

- Relation types from ACE (Automated Content Extraction)



Which Relations to Extract?

- Ontological relations

IS-A (hypernym): subsumption between classes

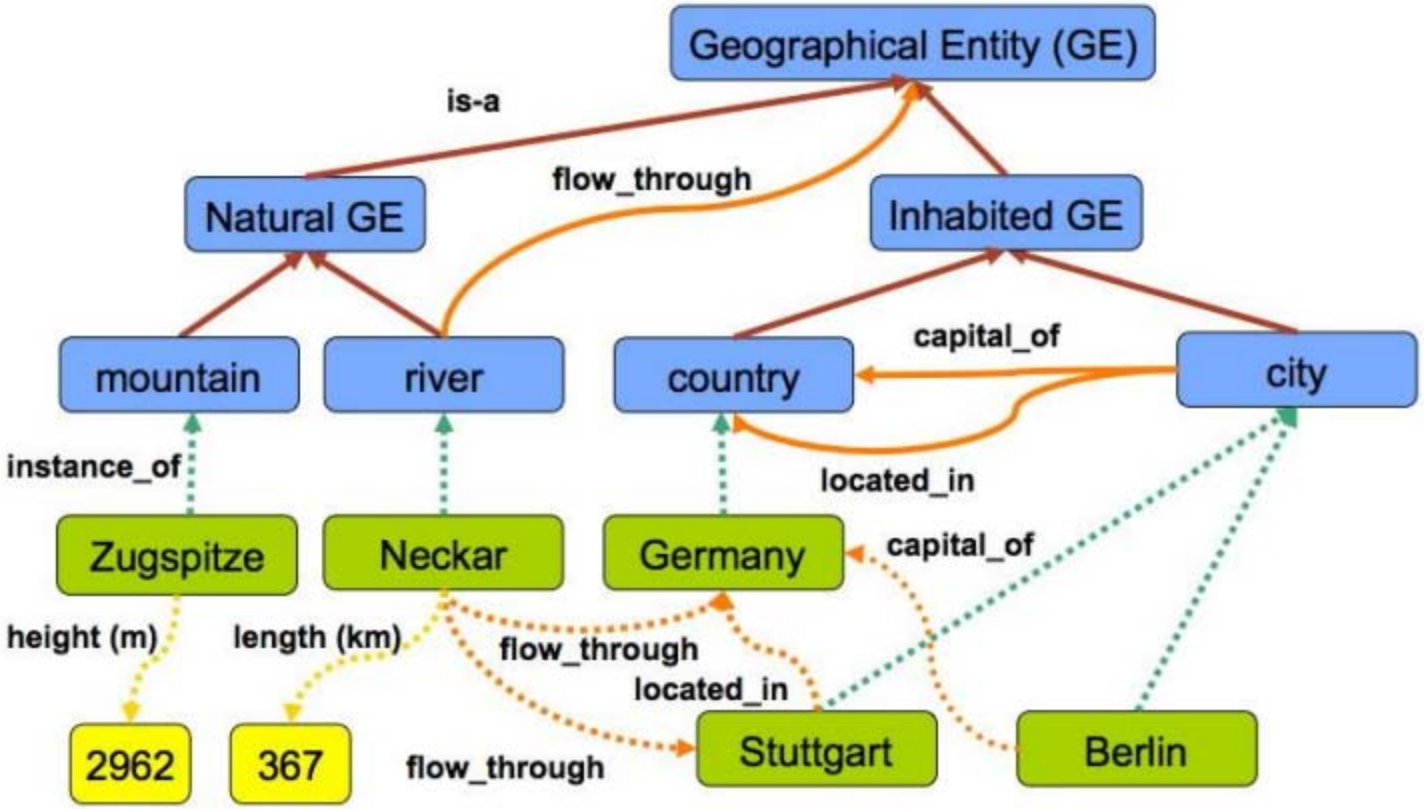
- `Giraffe` IS-A `ruminant` IS-A `ungulate` IS-A `mammal` IS-A `vertebrate` IS-A `animal`…

Instance-of: relation between individual and class

- `San Francisco` instance-of `city`

Which Relations to Extract?

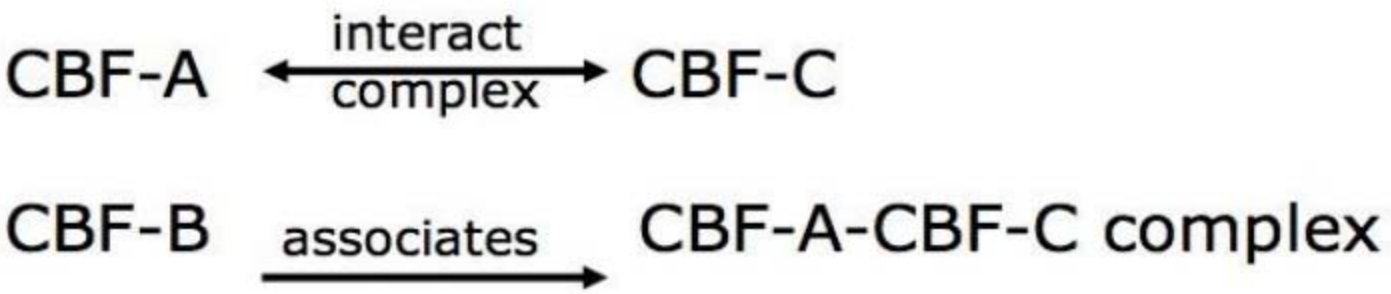
- Domain-centric relation types: geographical



Which Relations to Extract?

- Domain-centric relation types: biological

„We show that CBF-A and CBF-C interact with each other to form a CBF-A-CBF-C complex and that CBF-B does not interact with CBF-A or CBF-C individually but that it associates with the CBF-A-CBF-C complex."



Which Relations to Extract?

- Domain-centric relation types: medical

Injury	disrupts	Physiological Function
Bodily Location	location-of	Biologic Function
Anatomical Structure	part-of	Organism
Pharmacologic Substance	causes	Pathological Function
Pharmacologic Substance	treats	Pathologic Function

Which Relations to Extract?

- Open domain relations
  - No prior information of relation types.
  - Extract a large number of relations with high coverage.

Which Relations to Extract? – Summary

- Ontological relations
  - IsA (hypernym): subsumption between classes.
  - InstanceOf: relation between individual and class
- Domain-centric relations
  - Limited set of pre-defined relation types
  - E.g. UMLS(Unified Medical Language System): 54 relations
- Open-domain relations
  - No prior information of relation types.
  - Extract a large number of relations with high coverage.

How to Extract Relations?

- Hand written patterns/rules
- Supervised machine learning
- Semi-supervised bootstrapping
- Distant supervised methods
- Open-domain (unsupervised) extraction

Hearst's Patterns

- Hearst's Patterns for extracting IsA relations

Hearst pattern	Example occurrences
X and other Y	...temples, treasuries, and other important civic buildings.
X or other Y	bruises, wounds, broken bones or other injuries...
Y such as X	The bow lute, such as the Bambara ndang...
such Y as X	...such authors as Herrick, Goldsmith, and Shakespeare.
Y including X	...common-law countries, including Canada and England...
Y, especially X	European countries, especially France, England, and Spain...

Rules + NER

- Intuition: relations often hold between specific types
  - located-in (ORGANIZATION, LOCATION)
  - founded (PERSON, ORGANIZATION)
  - cures (DRUG, DISEASE)

- Extracting Richer Relations Using Rules and NER
  - PERSON (is born in | has the birthplace of | etc.) LOCATION
  - PERSON [be]? (named | appointed | etc.) Prep? POSITION

A Hand-built Extraction Rule

```
;;; For <company> appoints <person> <position>

(defpattern appoint
    "np-sem(C-company)?  rn?  sa?  vg(C-appoint) np-sem(C-person) ´,´?
     to-be?  np(C-position) to-succeed?:
     company-at=1.attributes, sa=3.span, lv=4.span, person-at=5.attributes
     position-at=8.attributes |
...
```

```
(defun when-appoint (phrase-type)
     (let ((person-at (binding ´person-at))
           (company-entity (entity-bound ´company-at))
           (person-entity (essential-entity-bound ´person-at ´C-person))
           (position-entity (entity-bound ´position-at))
           (predecessor-entity (entity-bound ´predecessor-at))
           new-event)
      (not-an-antecedent position-entity)
      ;; if no company is specified for position, use agent
...
```

NYU Proteus system (1997)

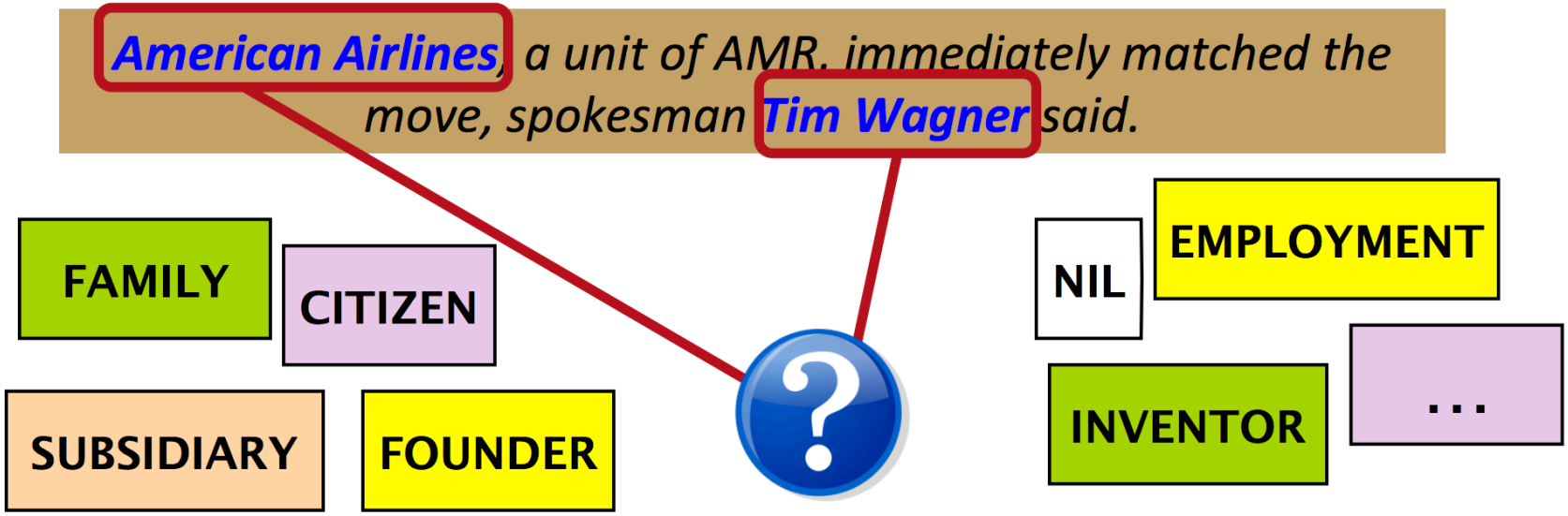
Hand Written Patterns/Rules - Summary

- Pros
  - high precision
  - can be tailored to specific domains
- Cons
  - hard to write/maintain
  - low recall
  - huge labor work
  - bad generalization

Supervised Machine Learning

- Choose a set of target relations
- Find and label data
  - Choose a representative corpus
  - Select sentences containing concepts
  - Hand label the relations between these concepts
  - Break into training, development, and test
- Train a classifier on the training set

Supervised Machine Learning - Labeling



Supervised Machine Learning - Features

- Light-weight features – require little preprocessing
  - bag of words/n-grams
  - types of entities
  - distance between entities
- Medium-weight features — require phrase chunking
  - phrase chunking paths
- Heavy-weight features – require syntactic parsing
  - parsing tree
  - dependency path

Word Features

American Airlines, *a unit of AMR, immediately matched the move, spokesman* ***Tim Wagner*** *said*

Mention 1 Mention 2

- Headwords of M1 and M2, and combination

  Airlines Wagner Airlines-Wagner

- Bag of words and bigrams in M1 and M2

  {American, Airlines, Tim, Wagner, American Airlines, Tim Wagner}

- Words or bigrams in particular positions left and right of M1/M2

  M2: -1 spokesman

  M2: +1 said

- Bag of words or bigrams between the two entities

  {a, AMR, of, immediately, matched, move, spokesman, the, unit}

Word features yield good precision (69%), but poor recall (24%)

Overlap Features

***American Airlines**, a unit of AMR, immediately matched the move, spokesman **Tim Wagner** said*
Mention 1 Mention 2

- **Number of mentions in between**
  1
- **Number of words in between**
  9
- **Whether one mention is included in the other**
  false

These features hurt precision a lot (-10%), but also help recall a lot (+8%)

Named Entity Type and Mention Level Features

American Airlines, *a unit of AMR, immediately matched the move, spokesman* ***Tim Wagner*** *said*
Mention 1 Mention 2

- Named-entity types
  - M1: ORG
  - M2: PERSON
- Concatenation of the two named-entity types
  - ORG-PERSON
- Entity Level of M1 and M2 (NAME, NOMINAL, PRONOUN)
  - M1: NAME [it or he would be PRONOUN]
  - M2: NAME [the company would be NOMINAL]

Named entity type features help recall a lot (+8%)
Mention level features have little impact

Phrase Chunking Features

***American Airlines**, a unit of AMR, immediately matched the move, spokesman **Tim Wagner** said*

Mention 1 Mention 2

[$_{NP}$ American Airlines], [$_{NP}$ a unit] [$_{PP}$ of] [$_{NP}$ AMR], [$_{ADVP}$ immediately] [$_{VP}$ matched] [$_{NP}$ the move], [$_{NP}$ spokesman Tim Wagner] [$_{VP}$ said].

- Base syntactic chunk sequence from one to the other

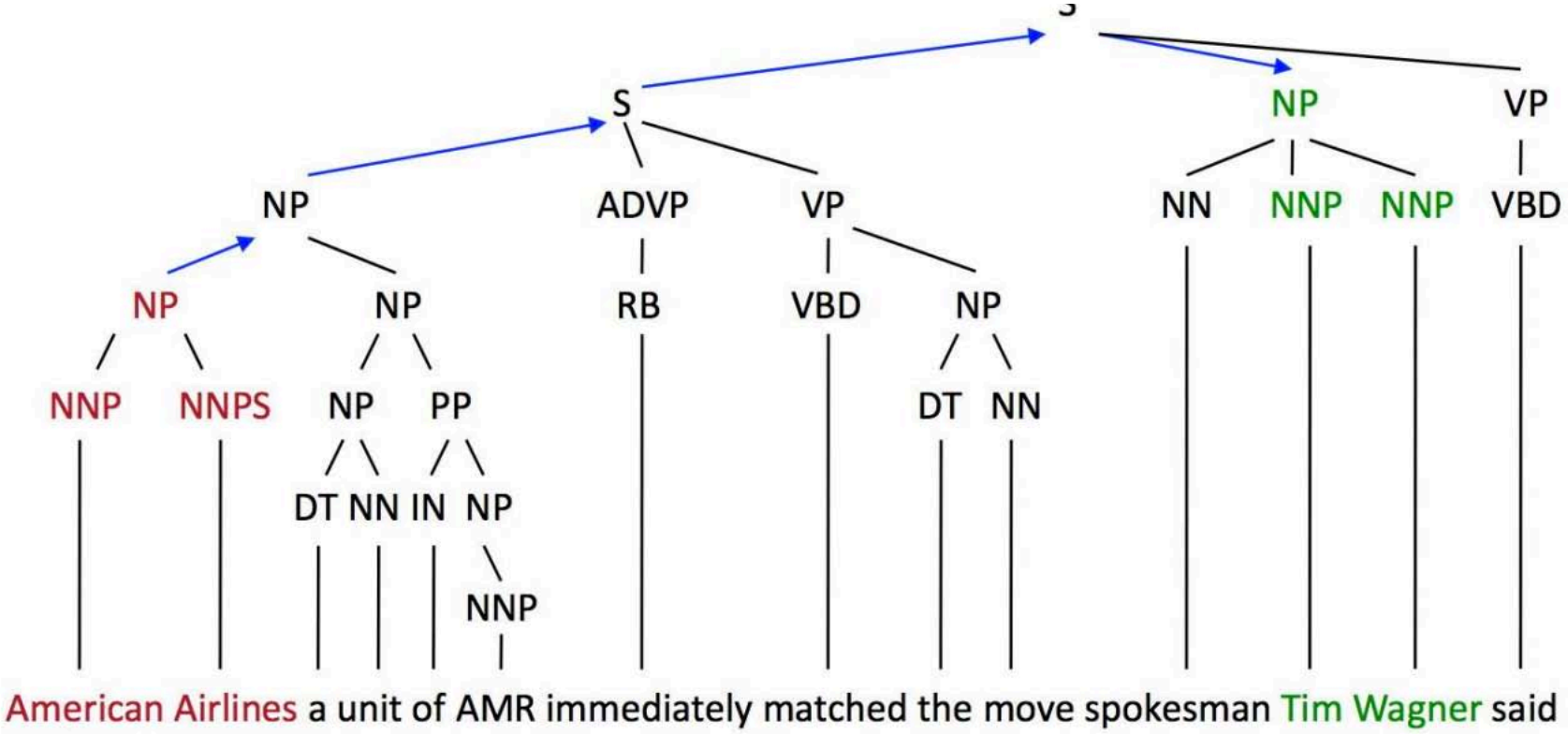
NP NP PP NP ADVP VP NP NP

These features increased both precision & recall by 4-6%

Parsing Features

***American Airlines**, a unit of AMR, immediately matched the move, spokesman **Tim Wagner** said*

Mention 1 Mention 2



These features had disappointingly little impact!

Semantic Resource Features

- Trigger list for family: kinship terms
  - parent, wife, husband, grandparent, etc. [from WordNet]
- Gazetteer:
  - Lists of useful geo or geopolitical words
    - Country name list
    - Other sub-entities

Feature Helpfulness – Summary

Features	P	R	F
Words	69.2	23.7	35.3
+Entity Type	67.1	32.1	43.4
+Mention Level	67.1	33.0	44.2
+Overlap	57.4	40.9	47.8
+Chunking	61.5	46.5	53.0
+Dependency Tree	62.1	47.2	53.6
+Parse Tree	62.3	47.6	54.0
+Semantic Resources	63.1	49.5	55.5

Table 2: Contribution of different features over 43 relation subtypes in the test data

Exploring various knowledge in relation extraction (ACL'05)

Supervised Machine Learning - Classifier

- Any classifier you like
  - SVM
  - Naïve Bayes
  - Max Entropy
  - Neural Nets
  - …

- Train it on the training set, tune on the dev set, test on the test set

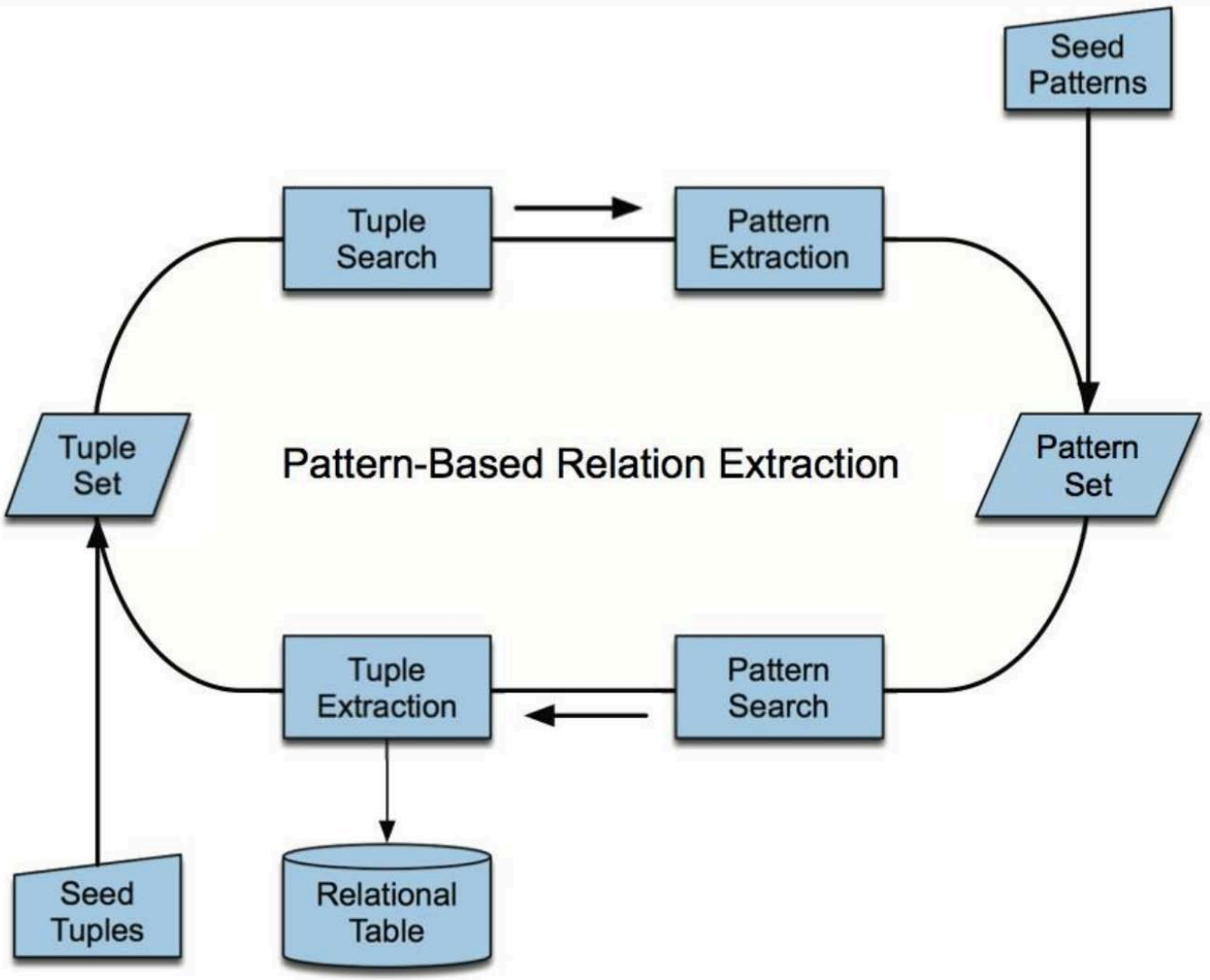
Supervised Machine Learning - Summary

- Pros
  - can get high accuracies with enough hand labeled training data, plus carefully selected features
- Cons
  - labeling a large training set is expensive
  - don't generalize to different relations

Semi-supervised Bootstrapping

- Gather a set of seed pairs that have relation R
- Iterate
  - Fetch sentences with these pairs
  - Analyze the context around the pair to generalize patterns
  - Use the patterns to gather more pairs satisfying R

Semi-supervised Bootstrapping



Semi-supervised Bootstrapping - Example

- Target Relation: burial place
- Seed tuple: [Mark Twain, Elmira]
- Fetch sentences containing “Mark Twain” and “Elmira”
  - “Mark Twain is buried in Elmira, NY.” → X is buried in Y
  - “The grave of Mark Twain is in Elmira” → The grave of X is in Y
  - “Elmira is Mark Twain’s final resting place” → Y is X’s final resting place
- Use those patterns to search for new tuples

DIRPE: Extract <author,book> pairs

Brin, Sergei. 1998. Extracting Patterns and Relations from the World Wide Web.

- Start with 5 seeds:

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

- Find Instances:

The Comedy of Errors, by William Shakespeare, was

The Comedy of Errors, by William Shakespeare, is

The Comedy of Errors, one of William Shakespeare's earliest attempts

The Comedy of Errors, one of William Shakespeare's most

- Extract patterns (group by middle, take longest common prefix/suffix)

```
?x , by ?y ,          ?x , one of ?y 's
```

- Now iterate, finding new seeds that match the pattern

Results: after three iterations of bootstrapping loop, extracted 15,000 author-book pairs with 95% accuracy

Semi-supervised Bootstrapping - Summary

- Pros
  - do not require manually labeling
- Cons
  - sensitive to original set of seeds
  - semantic drift at each iteration
  - hard to measure confidences of patterns/extractions

Distant Supervised Methods

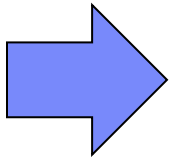
- Hypothesis: If two entities belong to a certain relation, any sentence containing those two entities is likely to express that relation

- Key idea: use a database of relations (e.g. Wikipedia InfoBox) to get lots of noisy training examples
  - instead of hand-creating seed tuples (bootstrapping)
  - instead of using hand-labeled corpus (supervised)

Distant Supervised Methods

Lawrence Livermore National Laboratory

Motto	"Science and Technology in the National Interest"
Established	1952 by the University of California; 62 years ago
Research type	Nuclear and basic science
Budget	$1.5 billion
Director	William H. Goldstein
Staff	5,800
Location	Livermore, California
Campus	1 square mile (2.6 km^2)
Operating agency	Lawrence Livermore National Security, LLC
Website	llnl.gov llnsllc.com



LLNL LOC-IN California
Livermore LOC-IN California
LLNL IS-A scientific research laboratory
LLNL FOUNDED-BY University of California
LLNL FOUNDED-IN 1952

Distant Supervised Methods

- Has advantages of supervised approach
  - leverage rich, reliable hand-created knowledge
  - can use rich features (e.g. syntactic features)
  - doesn't require iteratively expanding patterns
- Has advantages of unsupervised approach
  - leverage huge amounts of text data
  - do not require manually labeling

Distant Supervised Methods

1. For each relation
2. For each tuple in big database
3. Find sentences in large corpus with both entities
4. Extract frequent features (parse, words, etc)
5. Train supervised classifier using thousands of patterns

Born-In

<Edwin Hubble, Marshfield>
<Albert Einstein, Ulm>

Hubble was born in Marshfield
Einstein, born (1879), Ulm
Hubble's birthplace in Marshfield

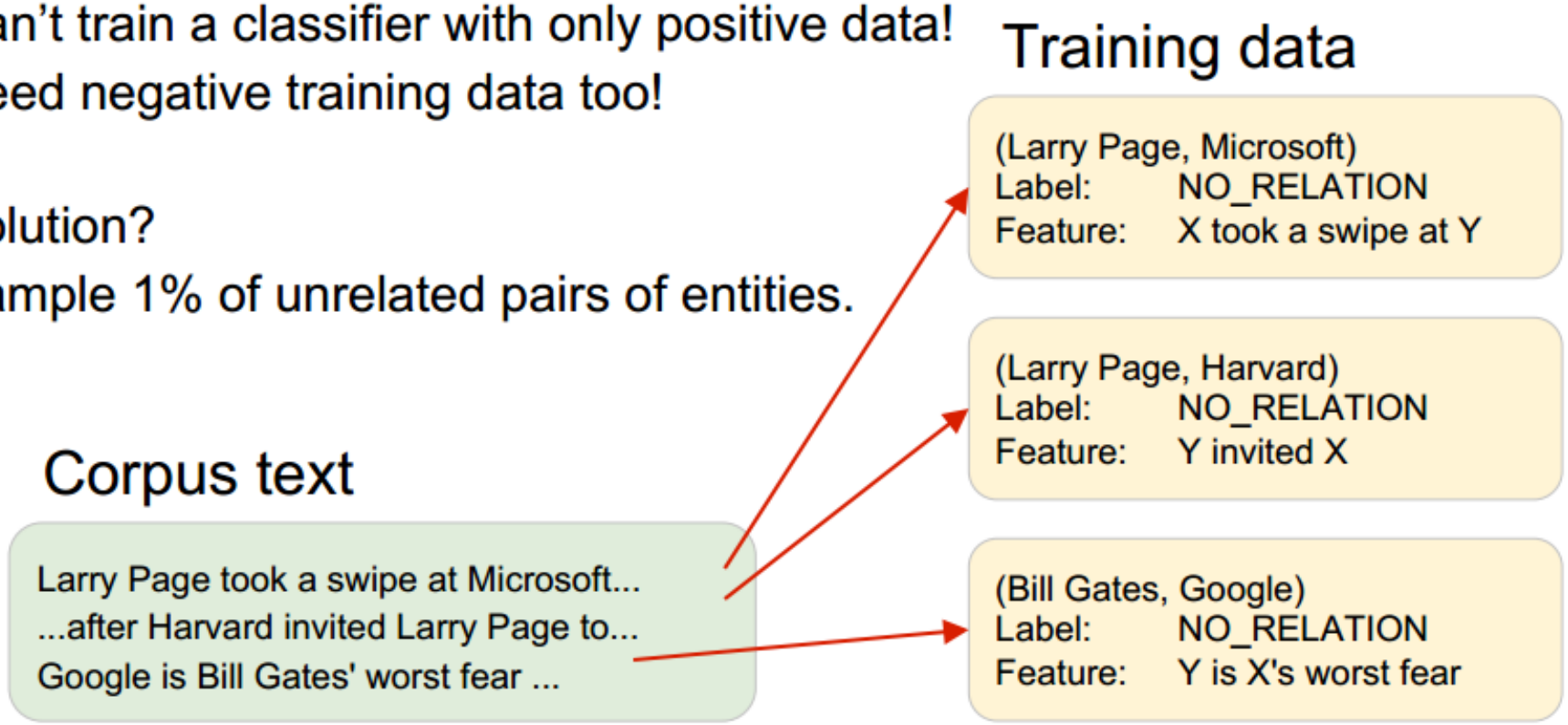
PER was born in LOC
PER, born (XXXX), LOC
PER's birthplace in LOC

$$P(\text{born-in} \mid f_1, f_2, f_3, \ldots, f_{70000})$$

How to Get Negative Training Data?

Can't train a classifier with only positive data!
Need negative training data too!

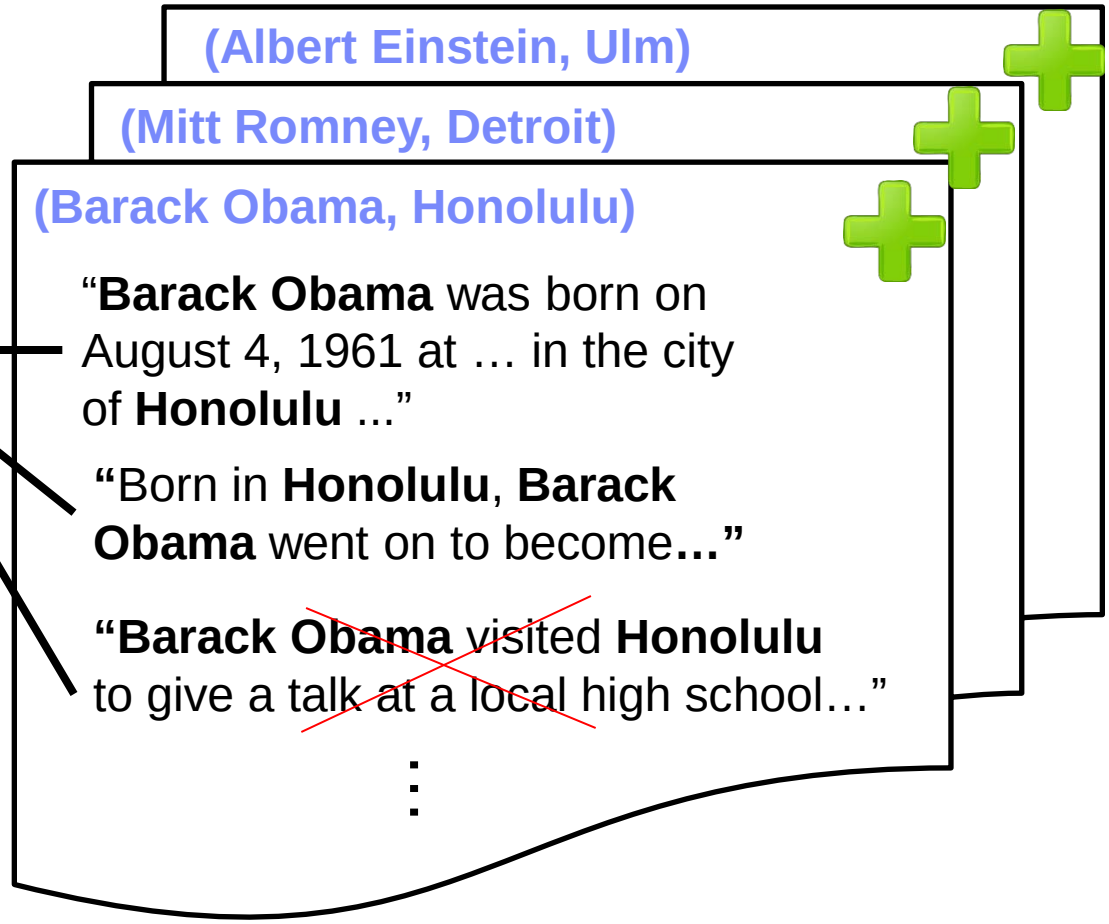
Solution?
Sample 1% of unrelated pairs of entities.



Any Problems with Distant Supervision?

Freebase

Person	Birth Location
Barack Obama	**Honolulu**
Mitt Romney	Detroit
Albert Einstein	Ulm
Nikola Tesla	Smiljan
...	...

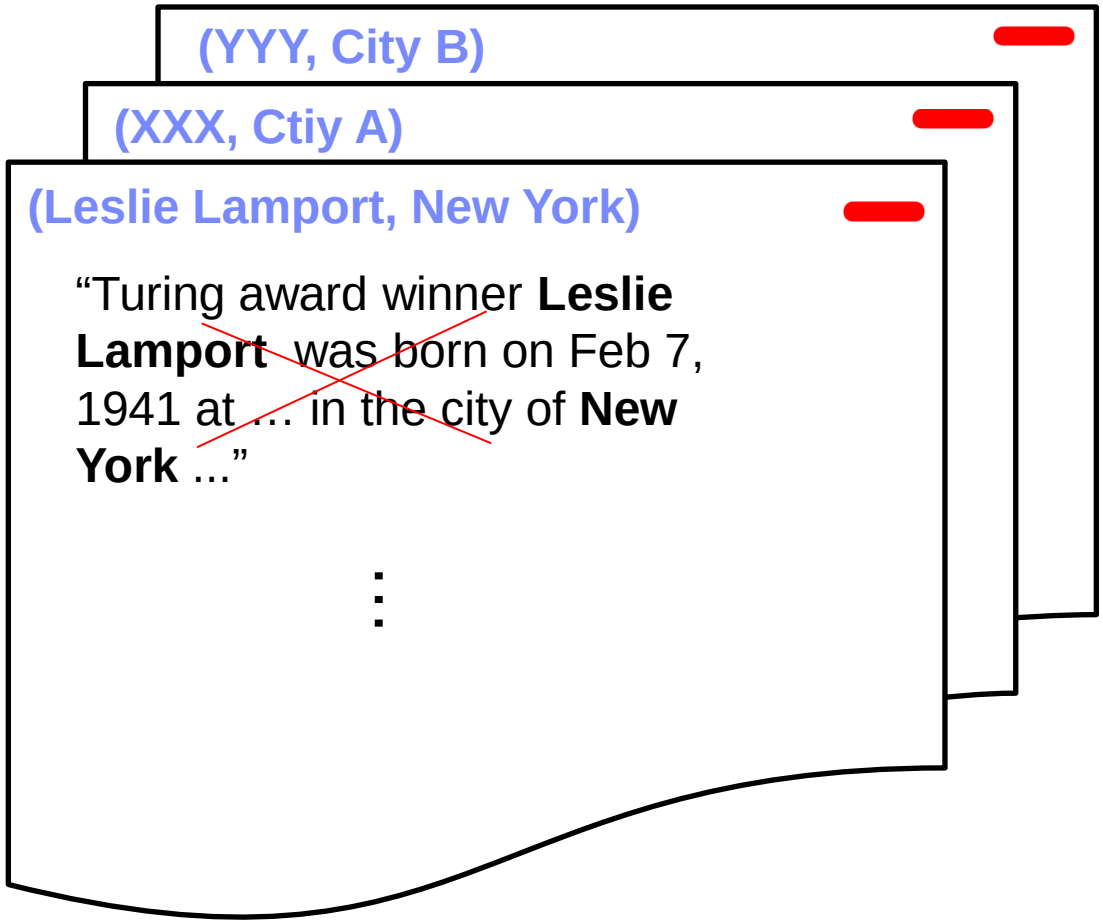


False Positive Training Data!

Any Problems with Distant Supervision?

Freebase

Person	Birth Location
Barack Obama	**Honolulu**
Mitt Romney	Detroit
Albert Einstein	Ulm
Nikola Tesla	Smiljan
...	...



False Negative Training Data!

Distant Supervised Methods - Summary

- Pros
  - high precision
  - do not require manually labeling
  - can utilize much more data than supervised methods
- Cons
  - false positive/negative training data (open research problem!)
  - poor coverage for tail relations

Open Relation Extraction

- Extract relations from the web with no training data, no list of relations: unsupervised understanding of text

ReVerb took .41 seconds.

Retrieved **49** results for Argument 1 containing **"Angela Merkel"**

Grouping results by argument 1. Group by: ***predicate*** | ***argument 2***

Angela Merkel (19 results)

Angela Merkel is chancellor of Germany (12)
Angela Merkel is Germany 's Chancellor (4), German chancellor (2), no Margaret Thatcher (5)
Angela Merkel became Germany 's first female chancellor (10), first woman chancellor of Germany (2)
Angela Merkel was elected Chancellor of Germany (5)
Angela Merkel was elected in 2005 (2)
Angela Merkel has the full confidence and support of the CD U and CSU (2), East German roots (2)
Angela Merkel was elected Chancellor in Germany (2)
Angela Merkel is head of the CDU. (2)
Angela Merkel is not a fawning head of state (2)
Angela Merkel was sworn in as Germany 's chancellor (3)
Angela Merkel apparently has a fear of dogs (2)
Angela Merkel was born in Hamburg (3)
Angela Merkel comes from East Germany (2)
Angela Merkel aimed this salvo (2)
Angela Merkel is the new chancellor in Germany (2)

Open Relation Extraction – TextRunner/Reverb

- Self-supervised learner: automatically labels +/- examples & learns a crude relation extractor
- Single-pass extractor: makes one pass over corpus, extracting candidate relations in each sentence

Step 1: Self-supervised Learner

Run a parser over 2000 sentences

- Parsing is relatively expensive, so can't run on whole web
- For each pair of base noun phrases NP_i and NP_j
- Extract all tuples t = (NP_i, $relation_{i,j}$, NP_j)

Label each tuple based on features of parse:

- Positive iff the dependency path between the NPs is short, and doesn't cross a clause boundary, and neither NP is a pronoun

Now train a Naïve Bayes classifier on the labeled tuples

- Using *lightweight* features like POS tags nearby, stop words, etc.

Step 2: Single-pass Extractor

Over a huge (web-sized) corpus:

- Run a dumb POS tagger
- Run a dumb Base Noun Phrase chunker
- Extract all text strings between base NPs
- Run heuristic rules to simplify text strings

  Scientists from many universities are intently studying stars

  → ⟨*scientists*, *are studying*, *stars*⟩

Pass candidate tuples to Naïve Bayes classifier

Save only those predicted to be "trustworthy"

Sample Extractions

Probability	Count	Arg1	Predicate	Arg2
0.98	59	Smith	invented	the margherita
0.97	49	Al Gore	invented	the Internet
0.97	44	manufacturing plant	first invented	the automatic revolver
0.97	41	Alexander Graham Bell	invented	the telephone
0.97	36	Thomas Edison	invented	light bulbs
0.97	29	Eli Whitney	invented	the cotton gin
0.96	23	C. Smith	invented	the margherita
0.96	19	the Digital Equipment Corporation manufacturing plant	first invented	the automatic revolver
0.96	18	Edison	invented	the phonograph

Evaluations

From corpus of 9M web pages, containing 133M sentences

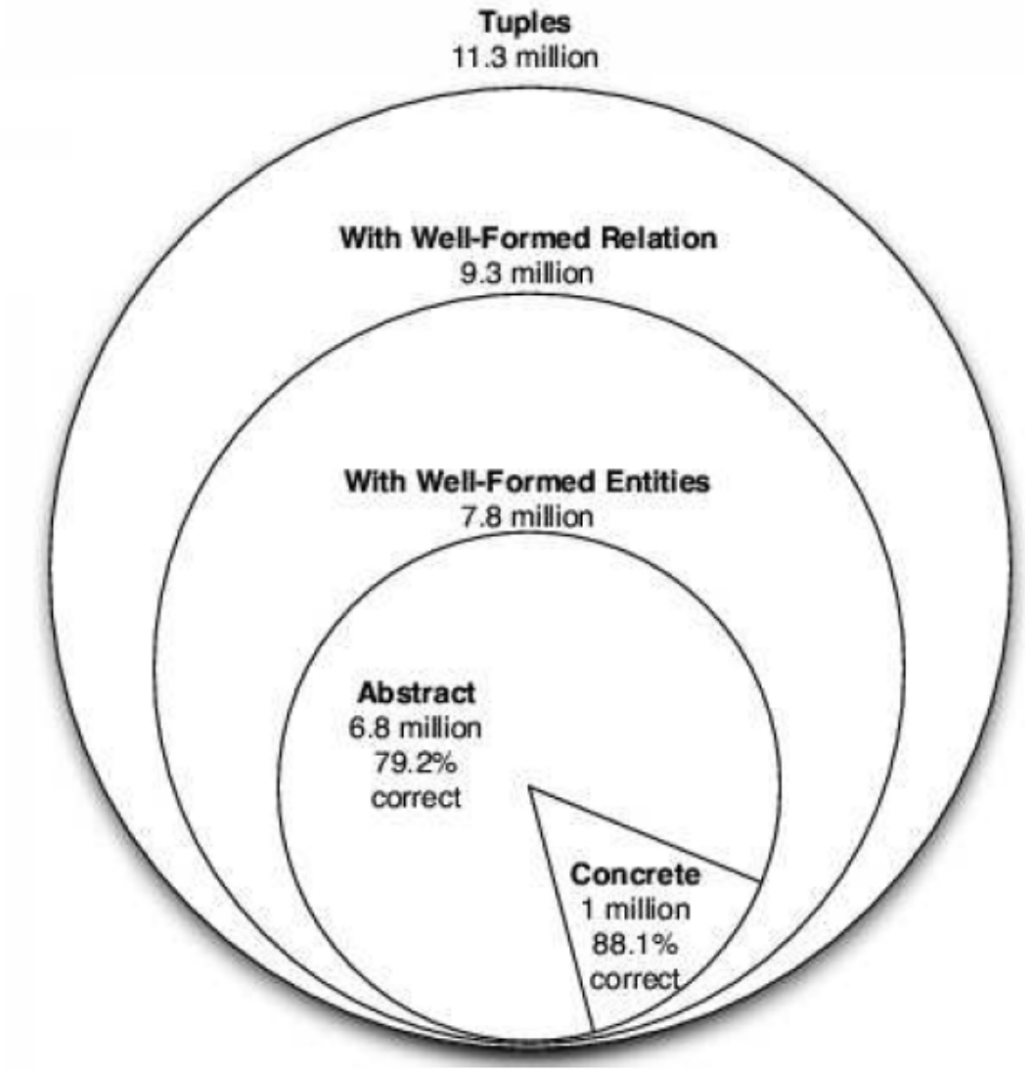
Extracted 60.5 million tuples

- ⟨*FCI, specializes in, software development*⟩

Evaluation

- Not well formed:
  - ⟨*demands, of securing, border*⟩ ⟨*29, dropped, instruments*⟩
- Abstract:
  - ⟨*Einstein, derived, theory*⟩ ⟨*executive, hired by, company*⟩
- True, concrete:
  - ⟨*Tesla, invented, coil transformer*⟩

Evaluations



Open Relation Extraction – Recent Progress

- **Reverb**: Identifying Relations for Open Information Extraction
- **ClausIE**: Clause-Based Open Information Extraction
- **OLLIE**: Open Language Learning for Information Extraction
- **CSDIE**: Open Information Extraction via Contextual Sentence Decomposition

Main Problem of Open Relation Extraction

- Extracted relations are not canonicalized!
- It is not trivial to judge if two relational phrases are semantically identical
  - "is born in" = "has the birthplace of"
- Can you develop a Relational Paraphrase Detection system to solve this problem?

Open Relation Extraction – Summary

- Pros
  - require no assumptions about domain knowledge
  - require no prior information of relation types
  - extract a large number of relations with high coverage

- Cons
  - result is noisy and contains some errors
  - extracted relations are not canonicalized
  - far from high quality

Available Toolkits for Relation Extraction

- Parser
  - Stanford parser: syntax and dependency parser (Java)
  - MST parser: dependency parser (Java)
  - Collins parser: syntax parser (C++) ; Dan Bikel duplicates in Java.
  - Charniak parser: syntax parser (C++)
- English NP chunker
  - OpenNLP: Java
  - GATE: Java
  - Ramshaw&Marcus: Java
- Named Entities Recognizer
  - Stanford NER: Java
  - MinorThird: Java (from William Cohen's group at CMU)
  - OpenNLP
  - GATE
- Tree Kernels in SVM-light

Questions?