Relation Extraction II

Luke Zettlemoyer

CSE 517

Winter 2013

[with slides adapted from many people, including Bill MacCartney, Raphael Hoffmann, Dan Jurafsky, Rion Snow, Jim Martin, Chris Manning, William Cohen, and others]

Supervised RE: summary

- Supervised approach can achieve high accuracy
  - At least, for *some* relations
  - If we have lots of hand-labeled training data
- But has significant limitations!
  - Labeling 5,000 relations (+ named entities) is expensive
  - Doesn’t generalize to different relations
- Next: beyond supervised relation extraction
  - Distantly supervised relation extraction
  - Unsupervised relation extraction

Relation extraction: 5 easy methods

1. Hand-built patterns
2. Bootstrapping methods
3. Supervised methods
4. Distant supervision
5. Unsupervised methods

Extracting structured knowledge

Each article can contain hundreds or thousands of items of knowledge


article | discussion | edit this page | history

Log in / create account

Lawrence Livermore National Laboratory

From Wikipedia, the free encyclopedia

Coordinates: 37.686024°N 121.709547°W

The **Lawrence Livermore National Laboratory (LLNL)** in Livermore, California is a scientific research laboratory founded by the University of California in 1952. It is funded by the United States Department of Energy (DOE) and managed by Lawrence Livermore National Security, LLC (LLNS), a partnership of the University of California, Bechtel Corporation, Babcock and Wilcox, the URS Corporation, and Battelle Memorial Institute. On October 1, 2007 LLNS assumed management of LLNL from the University of California, which had exclusively managed and operated the Laboratory since its inception 55 years before.

Contents [hide]

1 Background
2 Origins
3 Weapons projects
4 Plutonium research
5 National Ignition Facility and photon science
6 Global security program
7 Other programs
8 Key accomplishments
9 Unique facilities
10 World-class computers
11 Sponsors
12 Directors
13 Organization
14 Footnotes
15 References
16 External links and sources

Lawrence Livermore National Laboratory

University of California
Lawrence Livermore National Laboratory

Motto	"Science in the national interest"
Established	1952 by the University of California
Research Type	National security, nuclear science
Budget	US$1.6 billion
Director	George H. Miller
Staff	6,800
Location	Livermore, California
Campus	3.2 km² (800 acres)
Operating Agency	Lawrence Livermore National Security, LLC
Website	www.llnl.gov

Background [edit]

LLNL is self-described as "a premier research and development institution for science and technology applied to national security."[1] Its principal responsibility is ensuring the safety, security and reliability of the nation's nuclear weapons through the application of advanced science, engineering and technology. The Laboratory also applies its special expertise and multidisciplinary capabilities to preventing the proliferation and use of weapons of mass destruction, bolstering homeland security and solving other nationally important problems, including energy and environmental security, basic science and economic competitiveness.

LLNL is home to many unique facilities and a number of the most powerful computer systems in the world, according to the TOP500 list, including Blue Gene/L, the world's fastest computer from 2004 until Los Alamos National Laboratory's Roadrunner supercomputer surpassed it in 2008. The Lab is a leader in technical innovation: since 1978, LLNL has received a total of 118 prestigious R&D 100 Awards, including

Aerial view of Lawrence Livermore National Laboratory

WIKIPEDIA The Free Encyclopedia

navigation: Main page, Contents, Featured content, Current events, Random article

search: Go, Search

interaction: About Wikipedia, Community portal, Recent changes, Contact Wikipedia, Donate to Wikipedia, Help

toolbox: What links here, Related changes, Upload file, Special pages, Printable version, Permanent link, Cite this page

languages: Deutsch, فارسی, Français, Bahasa Indonesia, Italiano, 日本語, Norsk (bokmål), Polski, Русский, 中文

"The Lawrence Livermore National Laboratory (LLNL) in Livermore, California is ascientific research laboratory founded by the University of California in 1952."

LLNL EQ Lawrence Livermore National Laboratory
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 supervision

Snow, Jurafsky, Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. NIPS 17

Mintz, Bills, Snow, Jurafsky. 2009. Distant supervision for relation extraction without labeled data. ACL-2009.

- 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 to get lots of *noisy* training examples
  - instead of hand-creating seed tuples (bootstrapping)
  - instead of using hand-labeled corpus (supervised)

Benefits of distant supervision

- Has advantages of supervised approach
  - leverage rich, reliable hand-created knowledge
  - relations have canonical names
  - can use rich features (e.g. syntactic features)

- Has advantages of unsupervised approach
  - leverage unlimited amounts of text data
  - allows for very large number of weak features
  - not sensitive to training corpus: genre-independent

Hypernyms via distant supervision

We construct a noisy training set consisting of occurrences from our corpus that contain a hyponym-hypernym pair from WordNet.


The New York Times
Politics
Companies
The New York Times
Research

WORLD | U.S. | N.Y. / REGION | BUSINESS | TECHNOLOGY | SCIENCE | HEALTH | SPORTS | OPINION | ARTS | STYLE | TRAVEL | JOBS | REAL ESTATE | AUTOS

RESEARCH | FITNESS & NUTRITION | MONEY & POLICY | VIEWS | HEALTH GUIDE

Search Health 3,000+ Topics

Brain Researchers Open Door to Editing Memory

By BENEDICT CAREY
Published: April 5, 2009

Suppose scientists could erase certain memories by tinkering with a single substance in the brain. Could make you forget a chronic fear, a traumatic loss, even a bad habit.

Researchers in Brooklyn have recently accomplished comparable feats, with a single dose of an experimental drug delivered to areas of the brain critical for holding specific types of memory, like emotional associations, spatial knowledge or motor skills.

The drug blocks the activity of a substance that the brain apparently needs to retain much of its learned information. And if enhanced, the substance could help ward off dementias and other memory problems.

So far, the research has been done only on animals. But scientists say this memory system is likely to work almost identically in people.

The discovery of such an apparently critical memory molecule, and its many potential uses, are part of the buzz surrounding a field that, in just the past few years, has made the seemingly impossible suddenly probable: neuroscience, the study of the brain.

"If this molecule is as important as it appears to be, you can see the possible implications," said Dr. Todd C. Sacktor, a 52-year-old neuroscientist who leads the team at the SUNY Downstate Medical Center, in Brooklyn, which demonstrated its effect on memory. "For trauma.

Fred R. Conrad/The New York Times
André A. Fenton studies spatial memory in mice and rats.

Brain Power
The Speed-Dial Molecule
For all that scientists have studied it, the brain remains the most complex and mysterious human organ — and, now, the focus of billions of dollars' worth of research to penetrate its secrets.

Multimedia

Well
Tara Parker-Pope on Health
When Medicine Makes You Fat
Doctors Talk, but Do Patients Understand?
Believing in Treatments That Don't Work
Good Grief, Nurse Brown
Life Lessons From the Family Dog

Travel Deals E-mail newsletter
Sign up for travel deals and discounts on airfare, hotels, transportation and more!

Health & Fitness Tools
BMI Calculator
Calorie Calculator for Goal Weight

This yields high-signal examples like:

"...consider authors like Shakespeare..."
"Some authors (including Shakespeare)..."
"Shakespeare was the author of several..."
"Shakespeare, author of *The Tempest...*"

slide adapted from Rion Snow

Hypernyms via distant supervision

We construct a noisy training set consisting of occurrences from our corpus that contain a hyponym-hypernym pair from WordNet.

This yields high-signal examples like:

"...consider authors like Shakespeare..."
"Some authors (including Shakespeare)..."
"Shakespeare was the author of several..."
"Shakespeare, author of *The Tempest...*"

But also noisy examples like:

"The author of *Shakespeare in Love*..."
"...authors at the Shakespeare Festival..."

slide adapted from Rion Snow

Learning hypernym patterns

Key idea: work at corpus level (entity pairs), instead of sentence level!

1. Take corpus sentences

   ... doubly heavy hydrogen ***atom called deuterium*** *...*

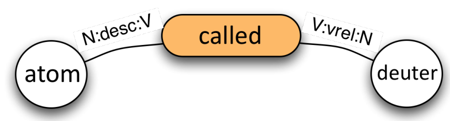
2. Collect noun pairs

   e.g. (atom, deuterium)
   752,311 pairs from 6M sentences of newswire

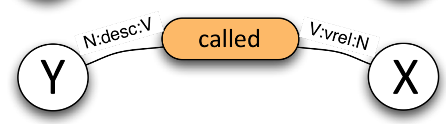
3. Is pair an IS-A in WordNet?

   14,387 yes; 737,924 no

4. Parse the sentences



5. Extract patterns



   69,592 dependency paths with >5 pairs

6. Train classifier on patterns

   logistic regression with 70K features
   (converted to 974,288 bucketed binary features)

slide adapted from Rion Snow

One of 70,000 patterns

Pattern: <superordinate> called <subordinate>

Learned from cases such as:

(sarcoma, cancer) …an uncommon bone cancer called osteogenic sarcoma and to…

(deuterium, atom) …heavy water rich in the doubly heavy hydrogen atom called deuterium.

New pairs discovered:

(efflorescence, condition) …and a condition called efflorescence are other reasons for…

(O'neal_inc, company) …The company, now called O'Neal Inc., was sole distributor of…

(hat_creek_outfit, ranch) …run a small ranch called the Hat Creek Outfit.

(hiv-1, aids_virus) …infected by the AIDS virus, called HIV-1.

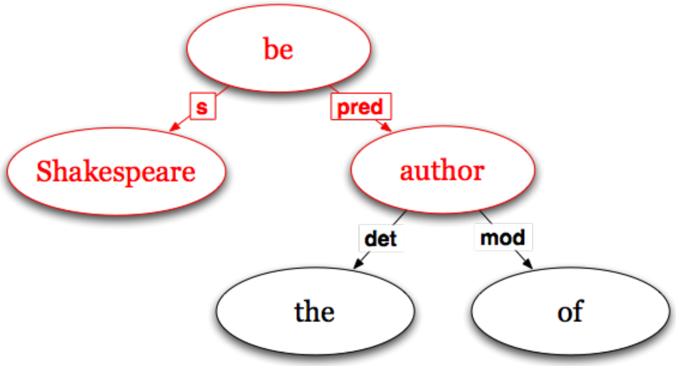
(bateau_mouche, attraction) …local sightseeing attraction called the Bateau Mouche...

Syntactic dependency paths

Patterns are based on paths through dependency parses generated by MINIPAR (Lin, 1998)

Example word pair: (Shakespeare, author)
Example sentence: "Shakespeare was the author of several plays..."

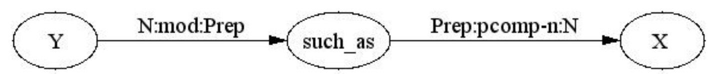
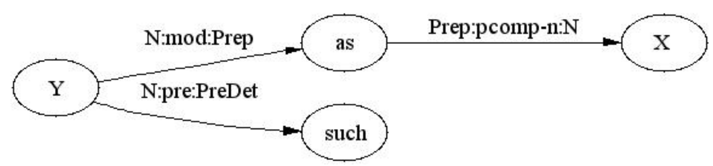
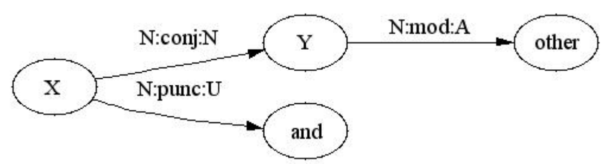
Minipar parse:



Extract shortest path:
-N:s:VBE, be, VBE:pred:N

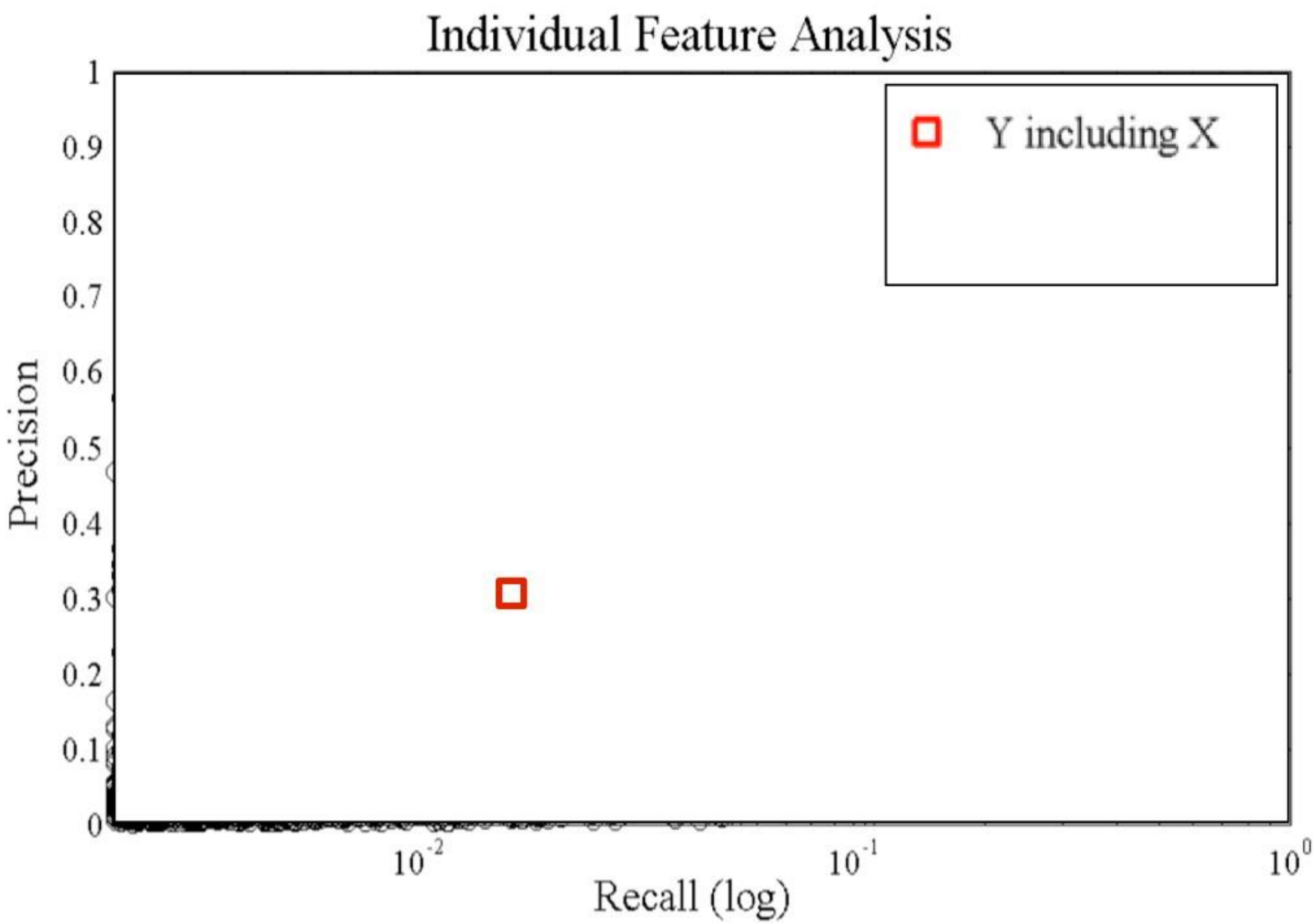
slide adapted from Rion Snow

Hearst patterns to dependency paths

Hearst Pattern	MINIPAR Representation
Y such as X …	 -N:pcomp-n:Prep,such_as,such_as,-Prep:mod:N
Such Y as X …	 -N:pcomp-n:Prep,as,as,-Prep:mod:N,(such,PreDet:pre:N)}
X … and other Y	 (and,U:punc:N),N:conj:N, (other,A:mod:N)

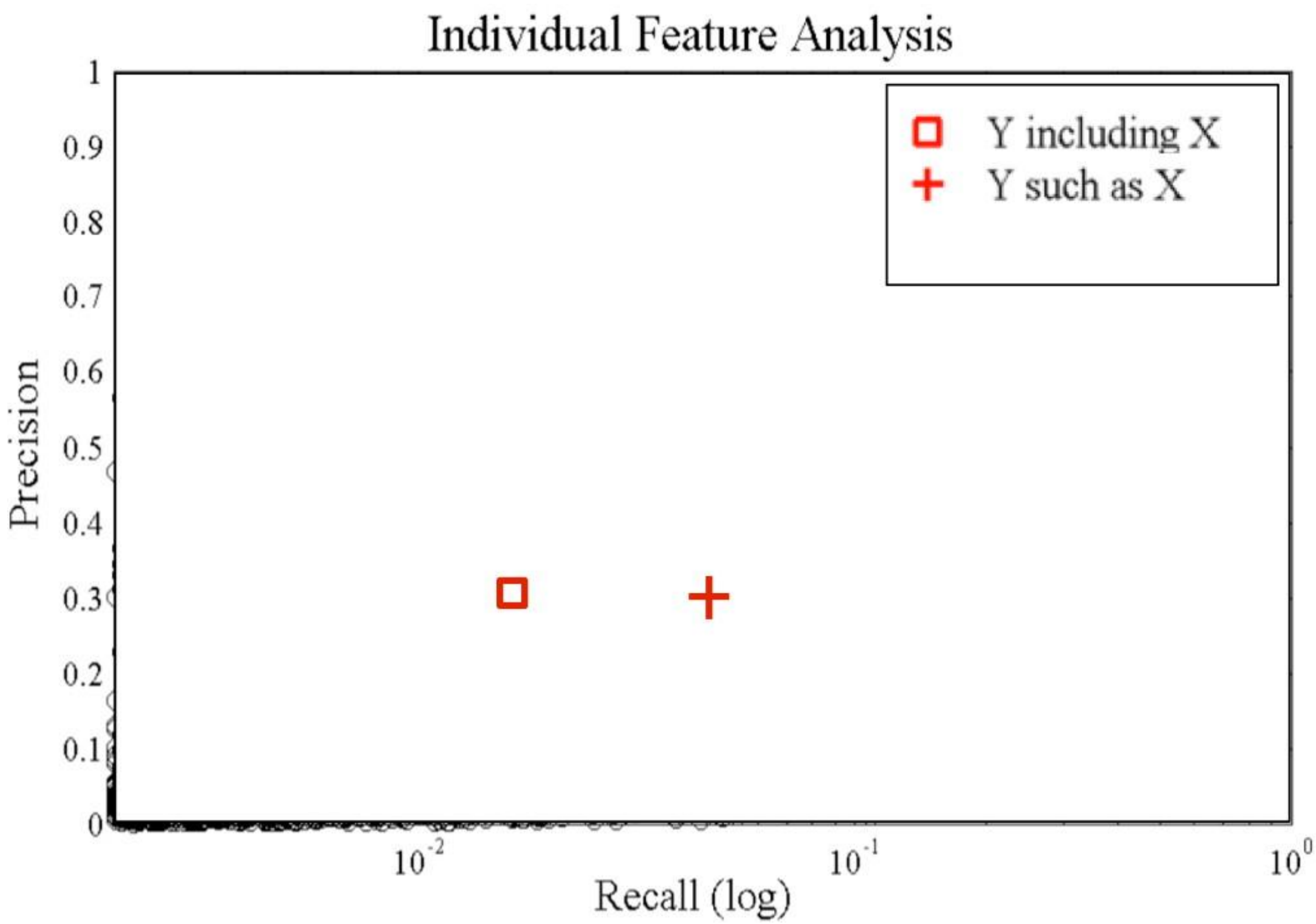
slide adapted from Rion Snow

P/R of hypernym extraction patterns



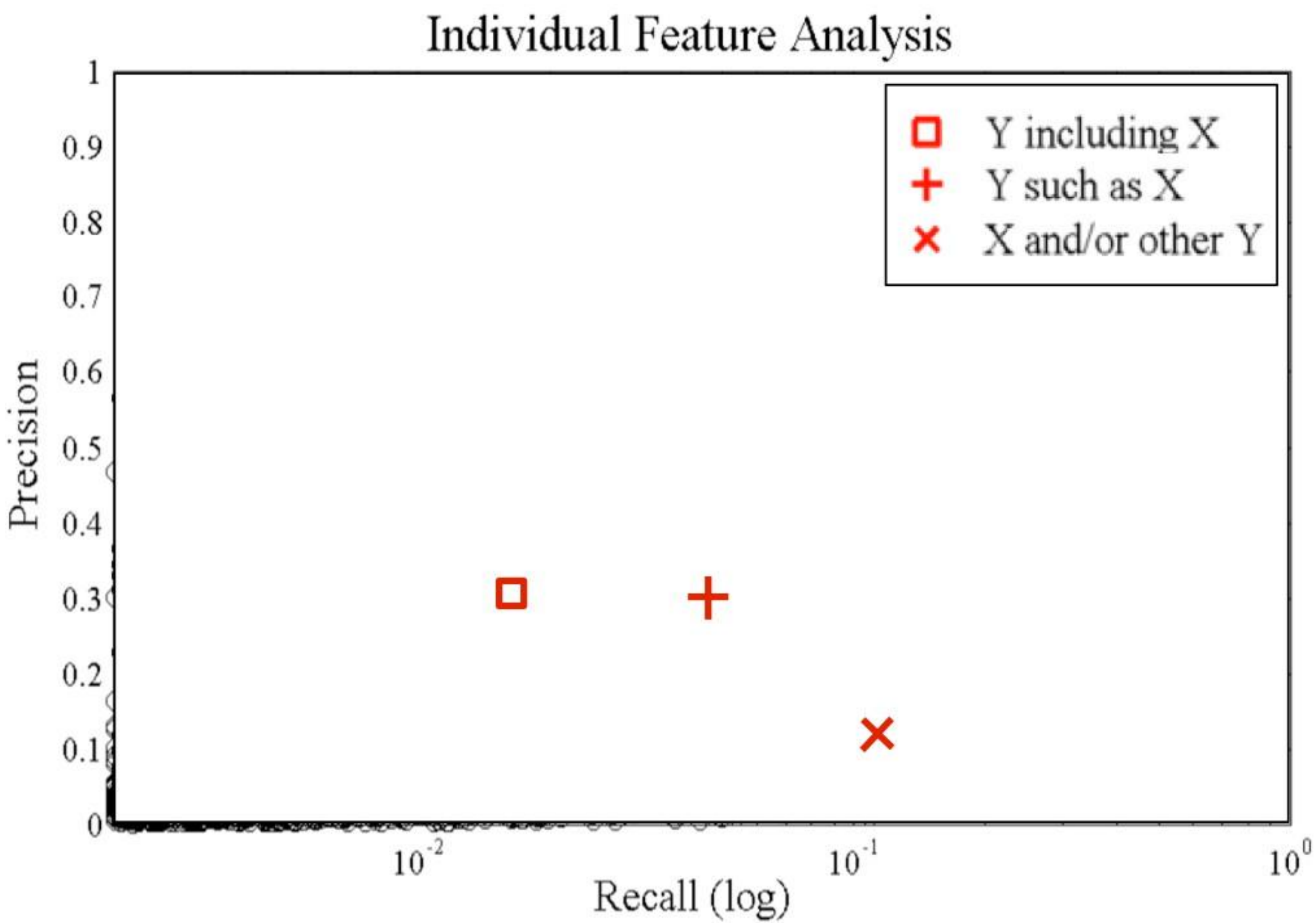
slide adapted from Rion Snow

P/R of hypernym extraction patterns

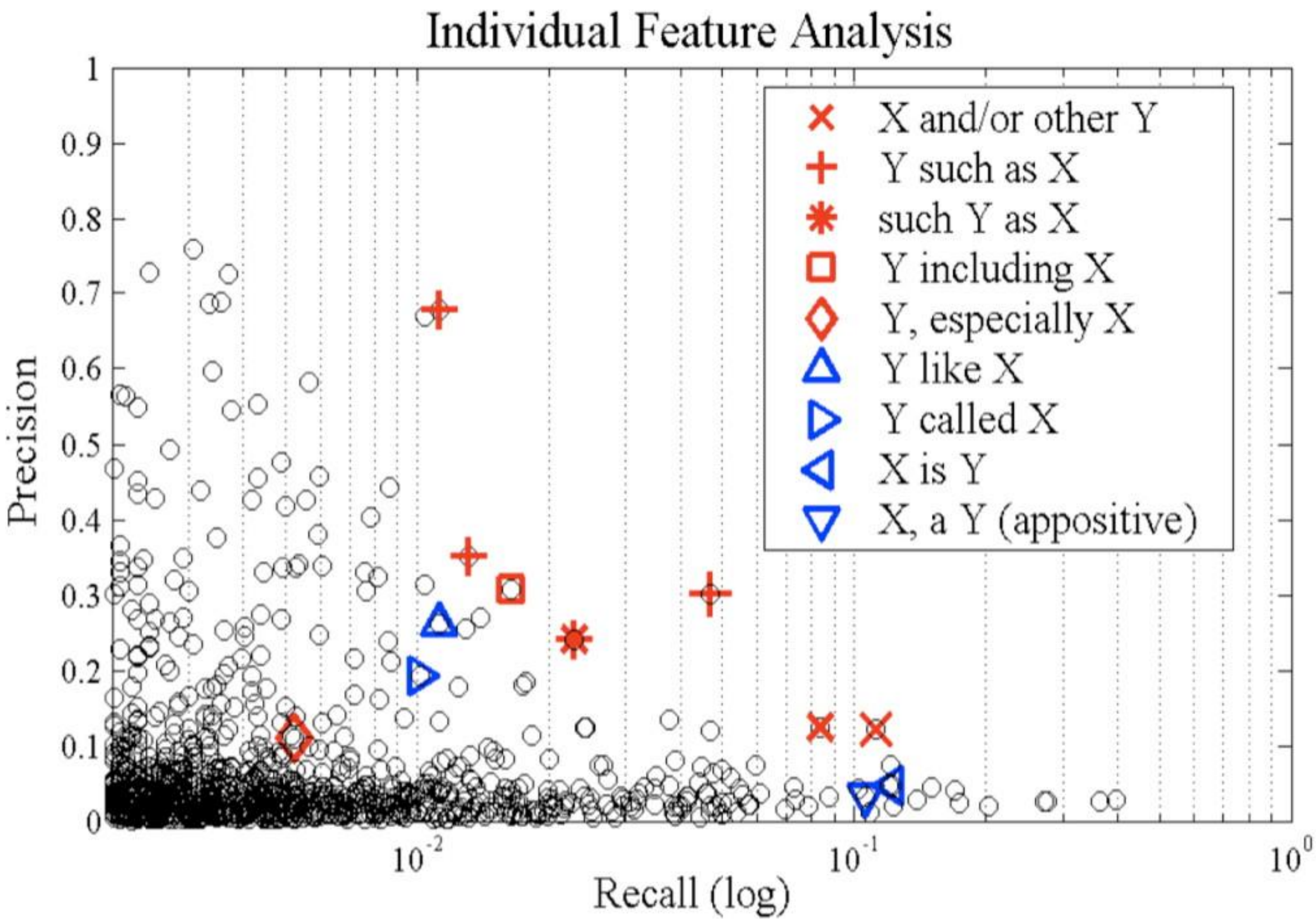


slide adapted from Rion Snow

P/R of hypernym extraction patterns

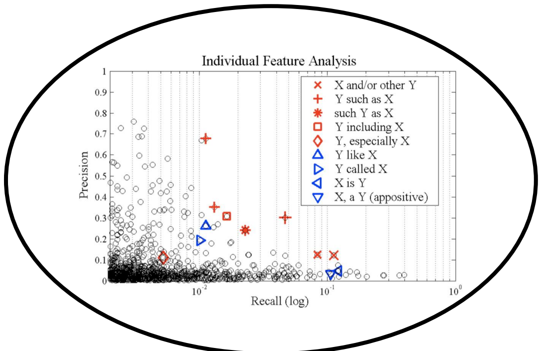


P/R of hypernym extraction patterns



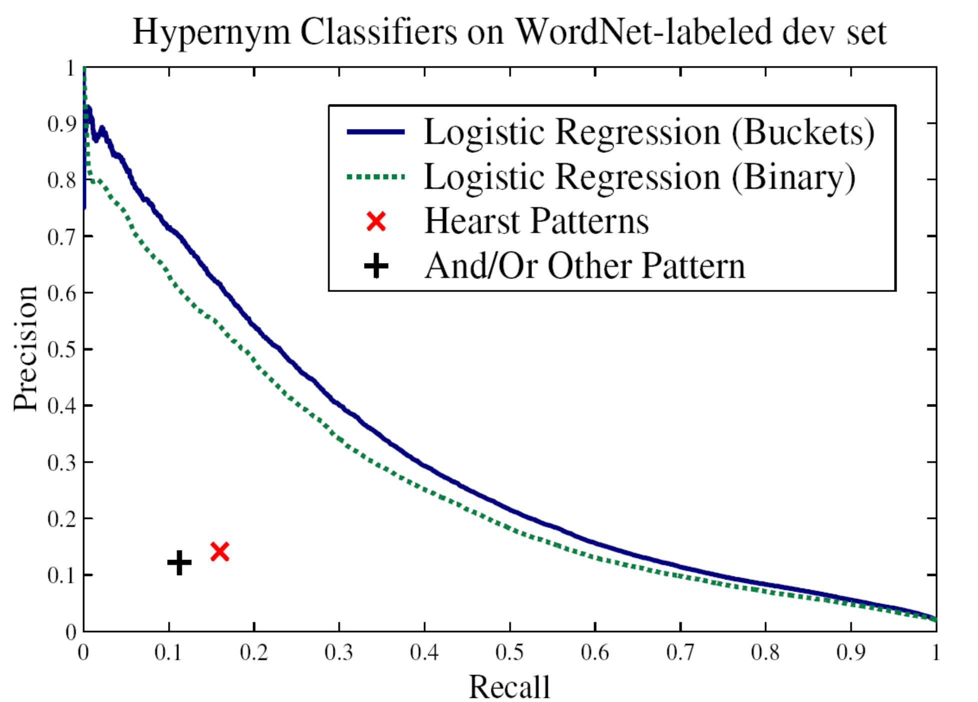
slide adapted from Rion Snow

P/R of hypernym classifier



logistic regression

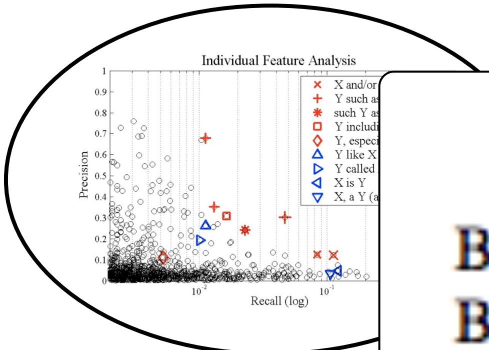
$$P(R|E) = \frac{1}{1 + e^{-\sum w_i x_i}}$$

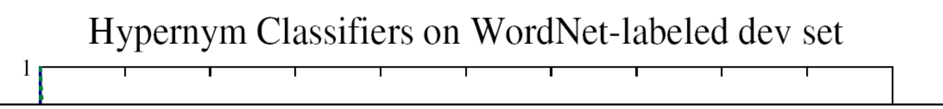


10-fold Cross Validation on
14,000 WordNet-Labeled Pairs

slide adapted from Rion Snow

P/R of hypernym classifier





	F-score
Best Logistic Regression (Buckets):	0.3480
Best Logistic Regression (Binary):	0.3200
Best Multinomial Naive Bayes:	0.3175
Best Complement Naive Bayes:	0.3024
Hearst Patterns:	0.1500
"And/Or Other" Pattern:	0.1170

logistic regres

$$P(R|E) = \frac{1}{1 + e^{-\sum w_i x_i}}$$

10-fold Cross Validation on
14,000 WordNet-Labeled Pairs

slide adapted from Rion Snow

What about other relations?

Mintz, Bills, Snow, Jurafsky (2009).
Distant supervision for relation extraction without labeled data.

Training set

Freebase

102 relations
940,000 entities
1.8 million instances

Corpus

Wikipedia

1.8 million articles
25.7 million sentences

slide adapted from Rion Snow

Frequent Freebase relations

Relation name	Size	Example
/people/person/nationality	281,107	John Dugard, South Africa
/location/location/contains	253,223	Belgium, Nijlen
/people/person/profession	208,888	Dusa McDuff, Mathematician
/people/person/place_of_birth	105,799	Edwin Hubble, Marshfield
/dining/restaurant/cuisine	86,213	MacAyo's Mexican Kitchen, Mexican
/business/business_chain/location	66,529	Apple Inc., Apple Inc., South Park, NC
/biology/organism_classification_rank	42,806	Scorpaeniformes, Order
/film/film/genre	40,658	Where the Sidewalk Ends, Film noir
/film/film/language	31,103	Enter the Phoenix, Cantonese
/biology/organism_higher_classification	30,052	Calopteryx, Calopterygidae
/film/film/country	27,217	Turtle Diary, United States
/film/writer/film	23,856	Irving Shulman, Rebel Without a Cause
/film/director/film	23,539	Michael Mann, Collateral
/film/producer/film	22,079	Diane Eskenazi, Aladdin
/people/deceased_person/place_of_death	18,814	John W. Kern, Asheville
/music/artist/origin	18,619	The Octopus Project, Austin
/people/person/religion	17,582	Joseph Chartrand, Catholicism
/book/author/works_written	17,278	Paul Auster, Travels in the Scriptorium
/soccer/football_position/players	17,244	Midfielder, Chen Tao
/people/deceased_person/cause_of_death	16,709	Richard Daintree, Tuberculosis
/book/book/genre	16,431	Pony Soldiers, Science fiction
/film/film/music	14,070	Stavisky, Stephen Sondheim
/business/company/industry	13,805	ATS Medical, Health care

Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Training data

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

(Bill Gates, Harvard)
Label: CollegeAttended
Feature: X attended Y

Collecting training data

Corpus text

Bill Gates founded Microsoft in 1975.
Bill Gates, founder of Microsoft, …
Bill Gates attended Harvard from…
Google was founded by Larry Page …

Freebase

Founder: (Bill Gates, Microsoft)
Founder: (Larry Page, Google)
CollegeAttended: (Bill Gates, Harvard)

Training data

(Bill Gates, Microsoft)
Label: Founder
Feature: X founded Y
Feature: X, founder of Y

(Bill Gates, Harvard)
Label: CollegeAttended
Feature: X attended Y

(Larry Page, Google)
Label: Founder
Feature: Y was founded by X

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.

Corpus text

Larry Page took a swipe at Microsoft...
...after Harvard invited Larry Page to...
Google is Bill Gates' worst fear ...

Training data

(Larry Page, Microsoft)
Label: NO_RELATION
Feature: X took a swipe at Y

(Larry Page, Harvard)
Label: NO_RELATION
Feature: Y invited X

(Bill Gates, Google)
Label: NO_RELATION
Feature: Y is X's worst fear

Preparing test data

Test data

Corpus text

Henry Ford founded Ford Motor Co. in…
Ford Motor Co. was founded by Henry Ford…
Steve Jobs attended Reed College from…

Preparing test data

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y

Corpus text

Henry Ford founded Ford Motor Co. in…
Ford Motor Co. was founded by Henry Ford…
Steve Jobs attended Reed College from…

Preparing test data

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y
Feature: Y was founded by X

Corpus text

Henry Ford founded Ford Motor Co. in…
Ford Motor Co. was founded by Henry Ford…
Steve Jobs attended Reed College from…

Preparing test data

Corpus text

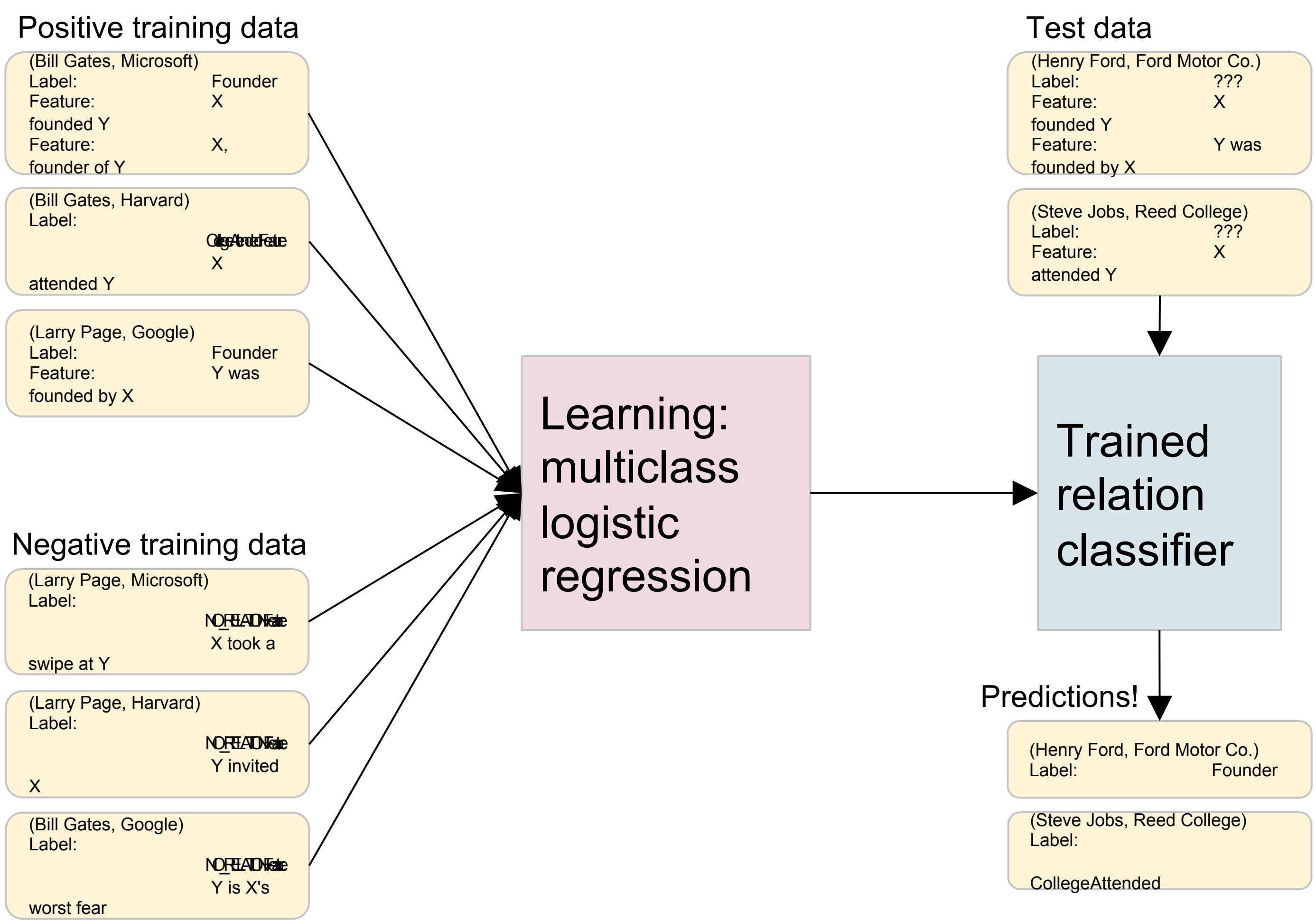
Henry Ford founded Ford Motor Co. in…
Ford Motor Co. was founded by Henry Ford…
Steve Jobs attended Reed College from…

Test data

(Henry Ford, Ford Motor Co.)
Label: ???
Feature: X founded Y
Feature: Y was founded by X

(Steve Jobs, Reed College)
Label: ???
Feature: X attended Y

The experiment

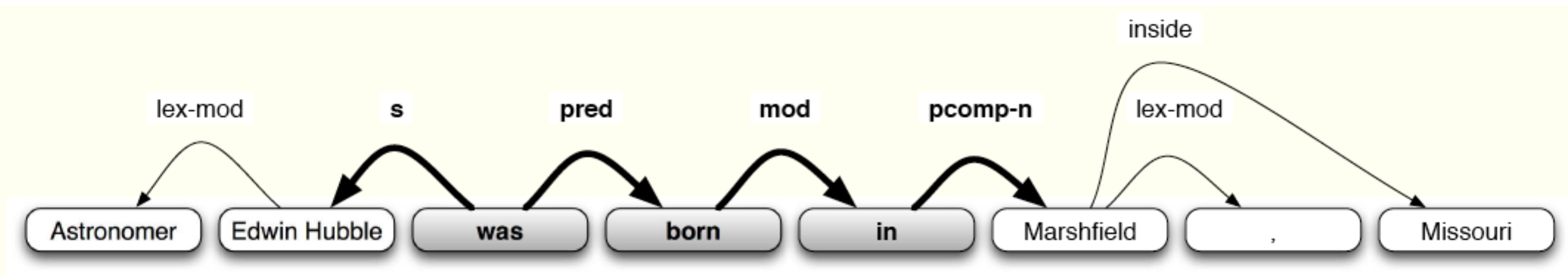


Advantages of the approach

- ACE paradigm: labeling sentences
- This paradigm: labeling entity pairs
- We make use of multiple appearances of entities
- If a pair of entities appears in 10 sentences, and each sentence has 5 features extracted from it, the entity pair will have 50 associated features

Lexical and syntactic features

Astronomer Edwin Hubble was born in Marshfield, Missouri.



Feature type	Left window	NE1	Middle	NE2	Right window
Lexical	[]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[]
Lexical	[Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[,]
Lexical	[#PAD#, Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[, Missouri]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{inside}$ Missouri]

High-weight features

Relation	Feature type	Left window	NE1	Middle	NE2	Right window
/architecture/structure/architect	LEX$\curvearrowleft$		ORG	, the designer of the	PER	
	SYN	designed $\Uparrow_s$	ORG	$\Uparrow_s$ designed $\Downarrow_{by-subj}$ by $\Downarrow_{pcn}$	PER	$\Uparrow_s$ designed
/book/author/works_written	LEX		PER	s novel	ORG	
	SYN		PER	$\Uparrow_{pcn}$ by $\Uparrow_{mod}$ story $\Uparrow_{pred}$ is $\Downarrow_s$	ORG	
/book/book_edition/author_editor	LEX$\curvearrowleft$		ORG	s novel	PER	
	SYN		PER	$\Uparrow_{nn}$ series $\Downarrow_{gen}$	PER	
/business/company/founders	LEX		ORG	co - founder	PER	
	SYN		ORG	$\Uparrow_{nn}$ owner $\Downarrow_{person}$	PER	
/business/company/place_founded	LEX$\curvearrowleft$		ORG	- based	LOC	
	SYN		ORG	$\Uparrow_s$ founded $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC	
/film/film/country	LEX		PER	, released in	LOC	
	SYN	opened $\Uparrow_s$	ORG	$\Uparrow_s$ opened $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC	$\Uparrow_s$ opened
/geography/river/mouth	LEX		LOC	, which flows into the	LOC	
	SYN	the $\Downarrow_{det}$	LOC	$\Uparrow_s$ is $\Downarrow_{pred}$ tributary $\Downarrow_{mod}$ of $\Downarrow_{pcn}$	LOC	$\Downarrow_{det}$ the
/government/political_party/country	LEX$\curvearrowleft$		ORG	politician of the	LOC	
	SYN	candidate $\Uparrow_{nn}$	ORG	$\Uparrow_{nn}$ candidate $\Downarrow_{mod}$ for $\Downarrow_{pcn}$	LOC	$\Uparrow_{nn}$ candidate
/influence/influence_node/influenced	LEX$\curvearrowleft$		PER	, a student of	PER	
	SYN	of $\Uparrow_{pcn}$	PER	$\Uparrow_{pcn}$ of $\Uparrow_{mod}$ student $\Uparrow_{appo}$	PER	$\Uparrow_{pcn}$ of
/language/human_language/region	LEX		LOC	- speaking areas of	LOC	
	SYN		LOC	$\Uparrow_{lex-mod}$ speaking areas $\Downarrow_{mod}$ of $\Downarrow_{pcn}$	LOC	
/music/artist/origin	LEX$\curvearrowleft$		ORG	based band	LOC	
	SYN	is $\Uparrow_s$	ORG	$\Uparrow_s$ is $\Downarrow_{pred}$ band $\Downarrow_{mod}$ from $\Downarrow_{pcn}$	LOC	$\Uparrow_s$ is
/people/deceased_person/place_of_death	LEX		PER	died in	LOC	
	SYN	hanged $\Uparrow_s$	PER	$\Uparrow_s$ hanged $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC	$\Uparrow_s$ hanged
/people/person/nationality	LEX		PER	is a citizen of	LOC	
	SYN		PER	$\Downarrow_{mod}$ from $\Downarrow_{pcn}$	LOC	
/people/person/parents	LEX		PER	, son of	PER	
	SYN	father $\Uparrow_{gen}$	PER	$\Uparrow_{gen}$ father $\Downarrow_{person}$	PER	$\Uparrow_{gen}$ father
/people/person/place_of_birth	LEX$\curvearrowleft$		PER	is the birthplace of	PER	
	SYN		PER	$\Uparrow_s$ born $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC	
/people/person/religion	LEX		PER	embraced	LOC	
	SYN	convert $\Downarrow_{appo}$	PER	$\Downarrow_{appo}$ convert $\Downarrow_{mod}$ to $\Downarrow_{pcn}$	LOC	$\Downarrow_{appo}$ convert

Implementation

- Classifier: multi-class logistic regression optimized using L-BFGS with Gaussian regularization (Manning & Klein 2003)
- Parser: MINIPAR (Lin 1998)
- POS tagger: MaxEnt tagger trained on the Penn Treebank (Toutanova et al. 2003)
- NER tagger: Stanford four-class tagger {PER, LOC, ORG, MISC, NONE} (Finkel et al. 2005)
- 3 configurations: lexical features, syntax features, both

Experimental set-up

- 1.8 million relation instances used for training
  - Compared to 17,000 relation instances in ACE
- 800,000 Wikipedia articles used for training, 400,000 different articles used for testing
- Only extract relation instances not already in Freebase

Newly discovered instances

Ten relation instances extracted by the system that weren't in Freebase

Relation name	New instance
/location/location/contains	Paris, Montmartre
/location/location/contains	Ontario, Fort Erie
/music/artist/origin	Mighty Wagon, Cincinnati
/people/deceased_person/place_of_death	Fyodor Kamensky, Clearwater
/people/person/nationality	Marianne Yvonne Heemskerk, Netherlands
/people/person/place_of_birth	Wavell Wayne Hinds, Kingston
/book/author/works_written	Upton Sinclair, Lanny Budd
/business/company/founders	WWE, Vince McMahon
/people/person/profession	Thomas Mellon, judge

Human evaluation

Precision, using Mechanical Turk labelers:

Relation name	100 instances			1000 instances		
	Syn	Lex	Both	Syn	Lex	Both
/film/director/film	**0.49**	0.43	0.44	**0.49**	0.41	0.46
/film/writer/film	**0.70**	0.60	0.65	**0.71**	0.61	0.69
/geography/river/basin_countries	0.65	0.64	**0.67**	**0.73**	0.71	0.64
/location/country/administrative_divisions	0.68	0.59	**0.70**	**0.72**	0.68	**0.72**
/location/location/contains	0.81	**0.89**	0.84	**0.85**	0.83	0.84
/location/us_county/county_seat	0.51	0.51	**0.53**	0.47	**0.57**	0.42
/music/artist/origin	0.64	0.66	**0.71**	0.61	**0.63**	0.60
/people/deceased_person/place_of_death	0.80	0.79	**0.81**	0.80	**0.81**	0.78
/people/person/nationality	0.61	0.70	**0.72**	0.56	0.61	**0.63**
/people/person/place_of_birth	**0.78**	0.77	**0.78**	0.88	0.85	**0.91**
Average	0.67	0.66	**0.69**	**0.68**	0.67	0.67

- At recall of 100 instances, using both feature sets (lexical and syntax) offers the best performance for a majority of the relations
- At recall of 1000 instances, using syntax features improves performance for a majority of the relations

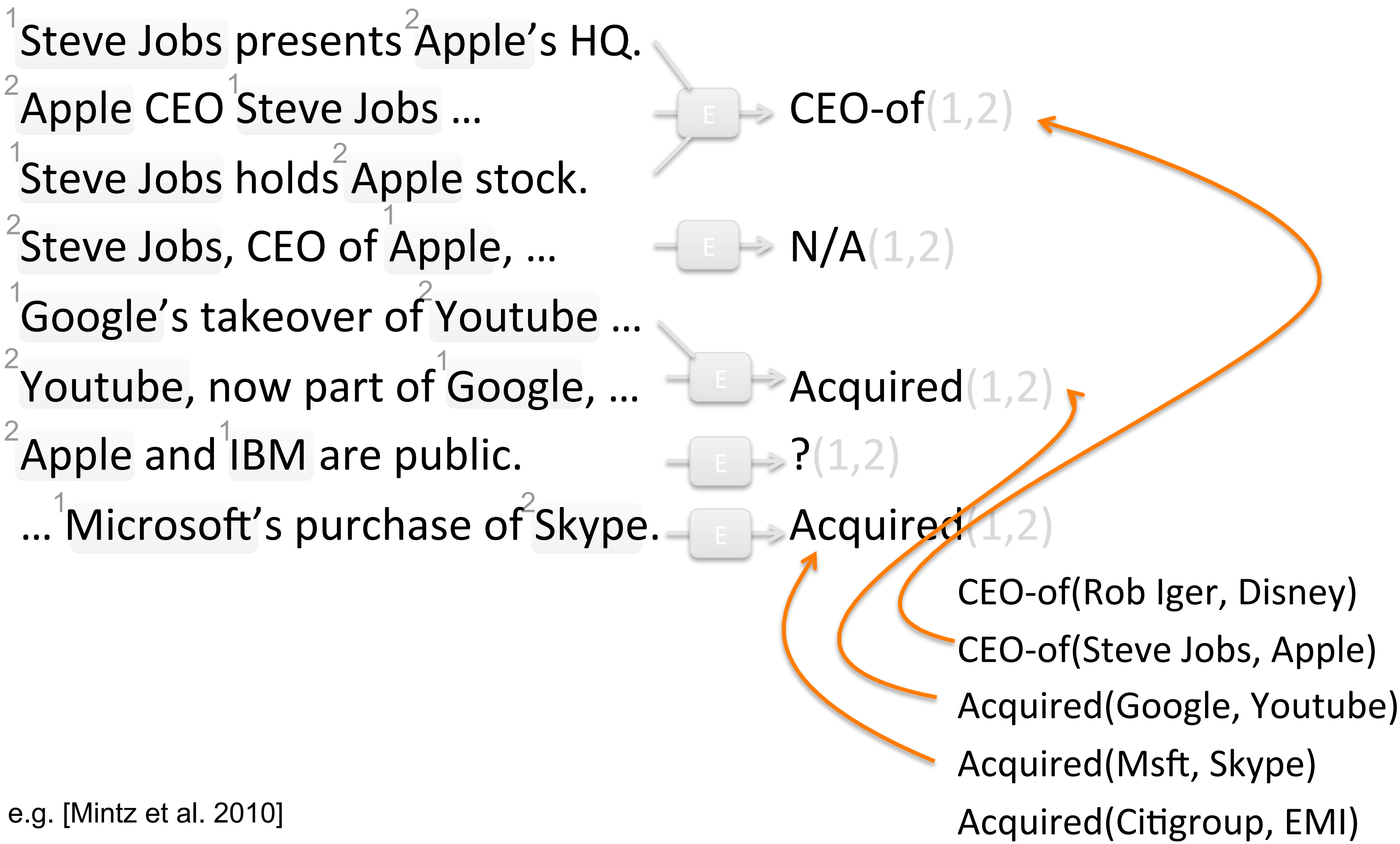
Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations

Raphael Hoffmann, Congle Zhang,
Xiao Ling, Luke Zettlemoyer, Daniel S. Weld

University of Washington

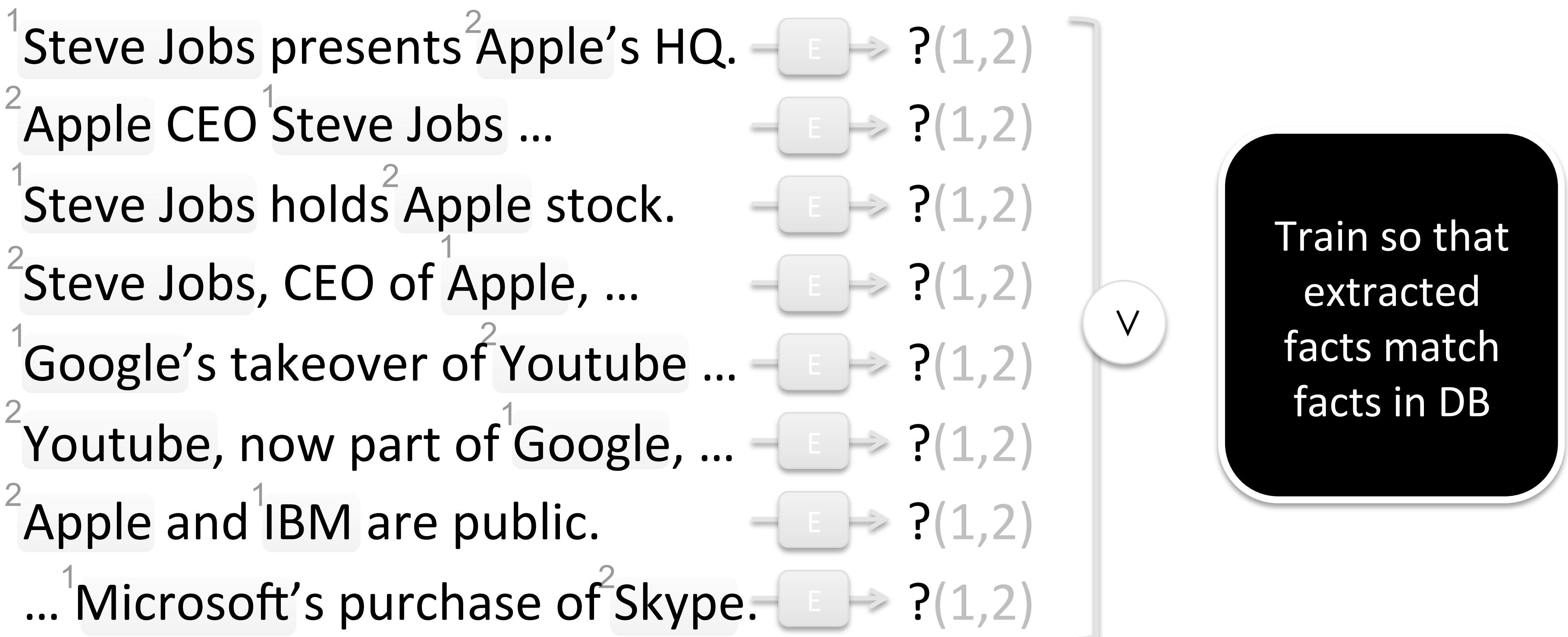
06/20/11

Previous Work: Aggregate Extraction



e.g. [Mintz et al. 2010]

This Talk: Sentence-level Reasoning



CEO-of(Rob Iger, Disney)

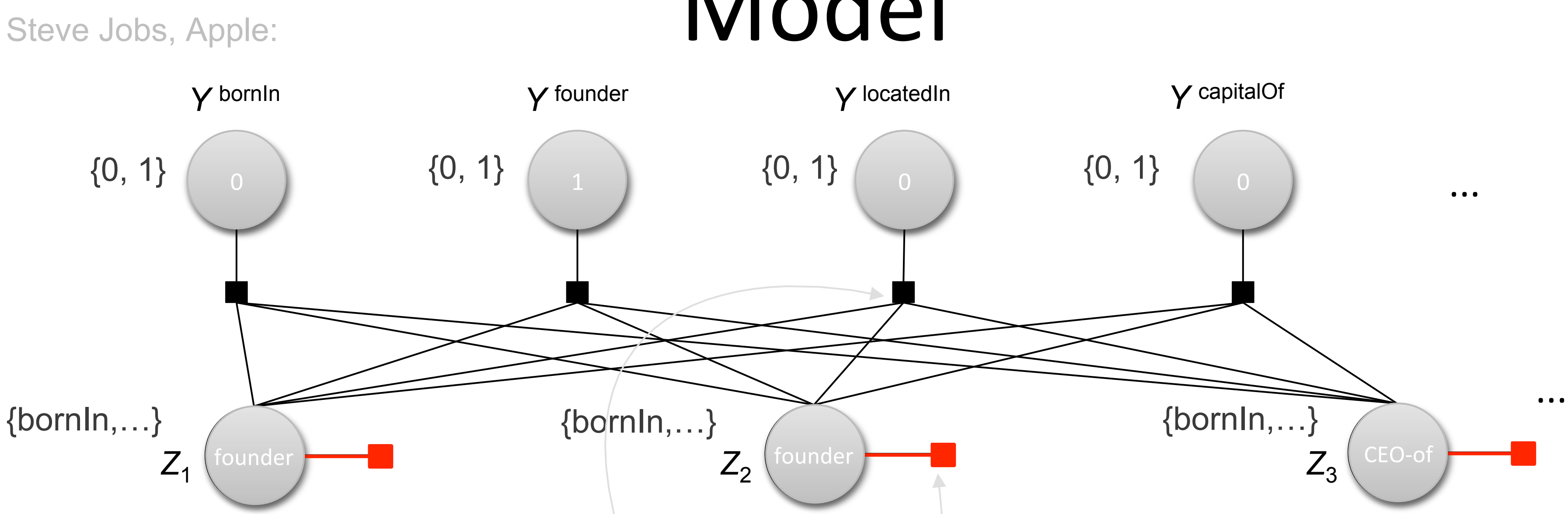
CEO-of(Steve Jobs, Apple)

Acquired(Google, Youtube)

Acquired(Msft, Skype)

Acquired(Citigroup, EMI)

Model



Steve Jobs was founder of **Apple**.

Steve Jobs, Steve Wozniak and Ronald Wayne founded **Apple**.

Steve Jobs is CEO of **Apple**.

...

$$p(\mathbf{Y}=\mathbf{y},\mathbf{Z}=\mathbf{z}|\mathbf{x};\theta) \stackrel{\text{def}}{=} \frac{1}{Z_x}\prod_r \Phi^{\text{join}}(y^r,\mathbf{z}) \prod_i \Phi^{\text{extract}}(z_i,x_i)$$

$$\Phi^{\text{join}}(y^r,\mathbf{z}) \stackrel{\text{def}}{=} \begin{cases} 1 & \text{if } y^r = true \wedge \exists i : z_i = r \\ 0 & \text{otherwise} \end{cases}$$

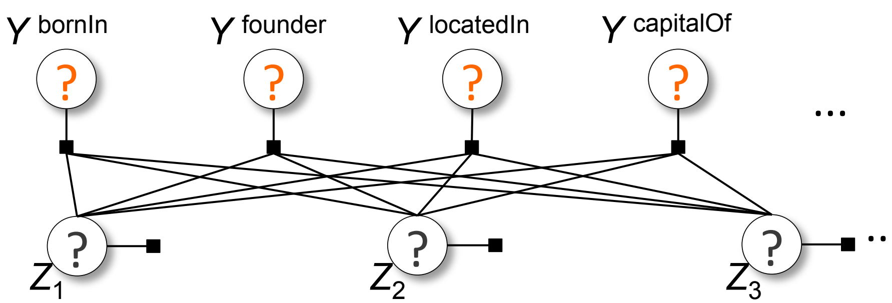
All features at sentence-level

(join factors are deterministic ORs)

Inference

Need:

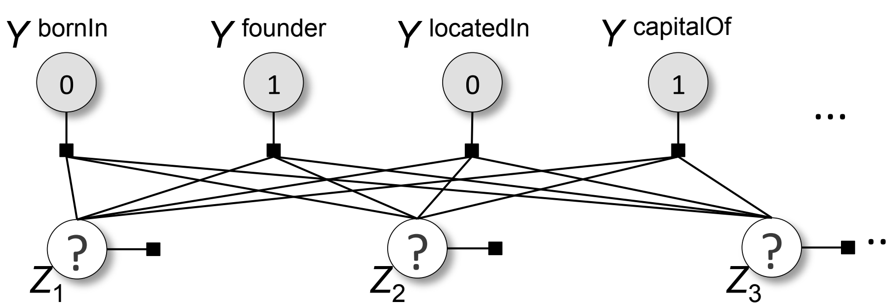
- Most likely sentence labels:



$$\arg\max_{\mathbf{y},\mathbf{z}} p(\mathbf{y},\mathbf{z}|\mathbf{x};\theta)$$

Easy

- Most likely sentence labels *given* facts:



$$\arg\max_{\mathbf{z}} p(\mathbf{z}|\mathbf{x},\mathbf{y};\theta)$$

Challenging

Learning: Hidden-Variable Perceptron

passes over dataset

for each entity pair *i*

most likely sentence labels and inferred facts (ignoring DB facts)

most likely sentence labels given DB facts

initialize parameter vector $\mathbf{\Theta} \leftarrow \mathbf{0}$

for $t = 1...T$ **do**

  for $i = 1...n$ **do**

    $(\mathbf{y}', \mathbf{z}') \leftarrow \arg\max_{\mathbf{y},\mathbf{z}} p(\mathbf{y}, \mathbf{z}|\mathbf{x_i}; \theta)$

    if $\mathbf{y}' \neq \mathbf{y_i}$ **then**

      $\mathbf{z}^* \leftarrow \arg\max_{\mathbf{z}} p(\mathbf{z}|\mathbf{x_i}, \mathbf{y_i}; \theta)$

      $\mathbf{\Theta} \leftarrow \mathbf{\Theta} + \phi(\mathbf{x_i}, \mathbf{z}^*) - \phi(\mathbf{x_i}, \mathbf{z}')$

    end if

  end for

end for

Return Θ

Experimental Setup

- Data as in Riedel et al. 10:
  – LDC NYT corpus, 2005-06 (training), 2007 (testing)
  – Data first tagged with Stanford NER system
  – Entities matched to Freebase, ~ top 50 relations
  – Mention-level features as in Mintz et al. 09
- Systems:
  – MultiR: proposed approach
  – SoloR: re-implementation of Riedel et al. 2010

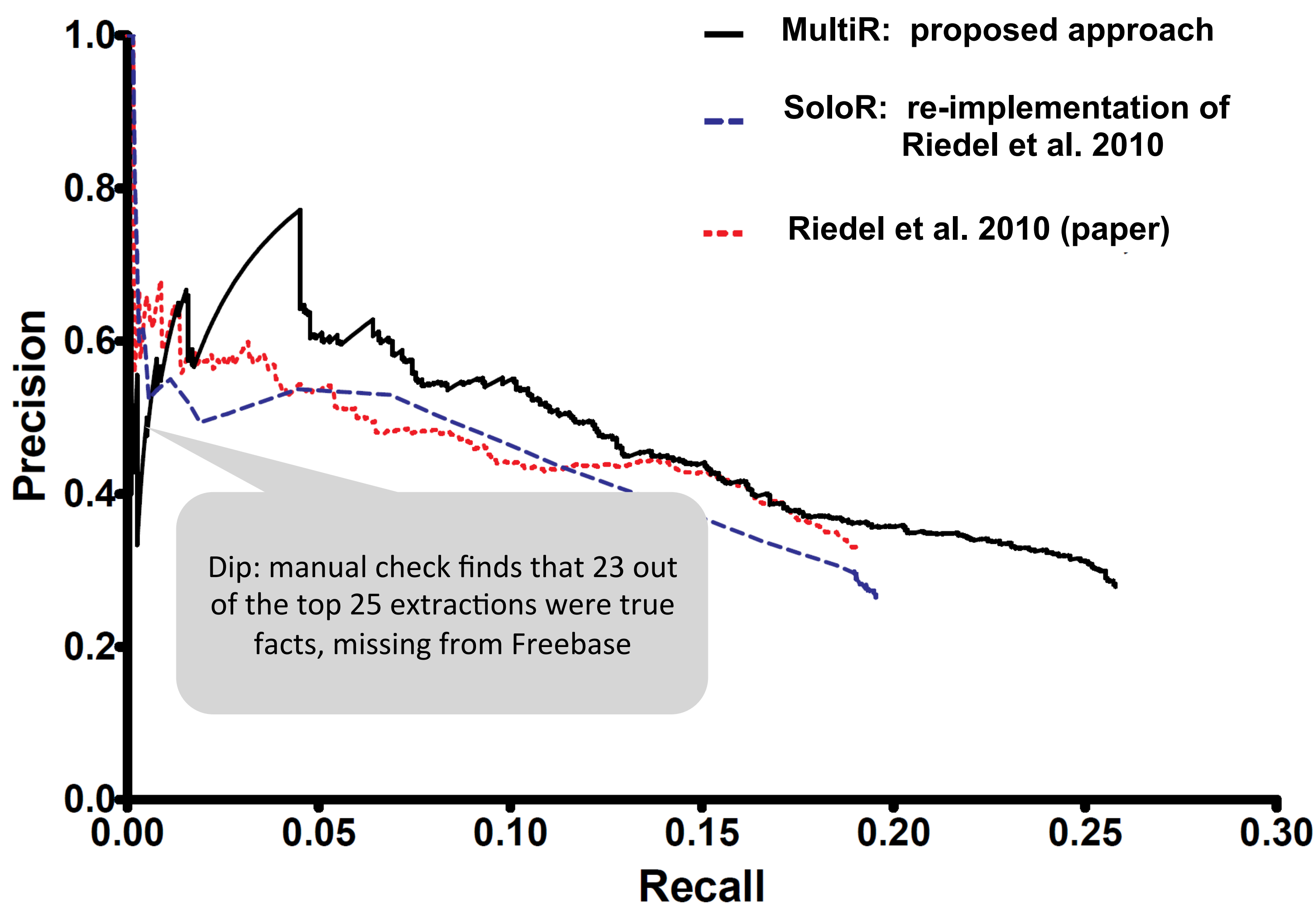
Aggregate Extraction

How does set of predicted facts match to facts in Freebase?

Metric

- For each entity pair compare inferred facts to facts in Freebase
- Automated, but underestimates precision

Aggregate Extraction



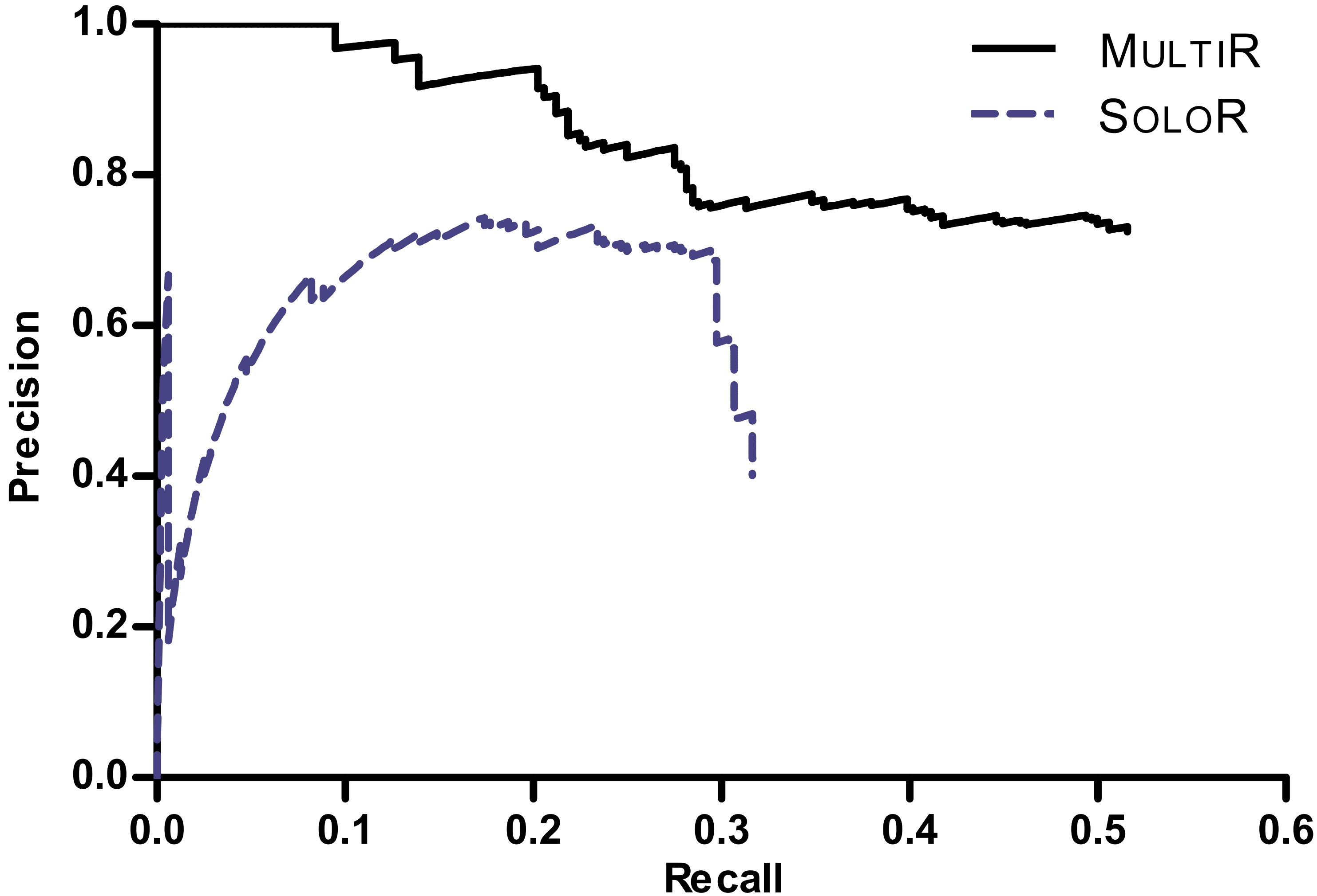
Sentential Extraction

How accurate is extraction from a given sentence?

Metric

- Sample 1000 sentences from test set
- Manual evaluation of precision and recall

Sentential Extraction



Relation-specific Performance

What is the quality of the matches for different relations?

How does our approach perform for different relations?

Metric:

- Select 10 relations with highest #matches
- Sample 100 sentences for each relation
- Manually evaluate precision and recall

Quality of the Matching

Relation	Freebase Matches #sents
/business/person/company	302
/people/person/place_lived	450
/location/location/contains	2793
/business/company/founders	95
/people/person/nationality	723
/location/neighborhood/neighborhood_of	68
/people/person/children	30
/people/deceased_person/place_of_death	68
/people/person/place_of_birth	162
/location/country/administrative_divisions	424

Quality of the Matching

Relation	Freebase Matches	
	#sents	% true
/business/person/company	302	89.0
/people/person/place_lived	450	60.0
/location/location/contains	2793	51.0
/business/company/founders	95	48.4
/people/person/nationality	723	41.0
/location/neighborhood/neighborhood_of	68	39.7
/people/person/children	30	80.0
/people/deceased_person/place_of_death	68	22.1
/people/person/place_of_birth	162	12.0
/location/country/administrative_divisions	424	0.2

Performance of MultiR

Relation	**Freebase Matches**		**MultiR**	
	#sents	% true	precision	recall
/business/person/company	302	89.0	100.0	25.8
/people/person/place_lived	450	60.0	80.0	6.7
/location/location/contains	2793	51.0	100.0	56.0
/business/company/founders	95	48.4	71.4	10.9
/people/person/nationality	723	41.0	85.7	15.0
/location/neighborhood/neighborhood_of	68	39.7	100.0	11.1
/people/person/children	30	80.0	100.0	8.3
/people/deceased_person/place_of_death	68	22.1	100.0	20.0
/people/person/place_of_birth	162	12.0	100.0	33.0
/location/country/administrative_divisions	424	0.2	N/A	0.0

Overlapping Relations

Relation	Freebase Matches		MultiR	
	#sents	% true	precision	recall
/business/person/company	302	89.0	100.0	25.8
/people/person/place_lived	450	60.0	80.0	6.7
/location/location/contains	2793	51.0	100.0	56.0
/business/company/founders	95	48.4	71.4	10.9
/people/person/nationality	723	41.0	85.7	15.0
/location/neighborhood/neighborhood_of	68	39.7	100.0	11.1
/people/person/children	30	80.0	100.0	8.3
/people/deceased_person/place_of_death	68	22.1	100.0	20.0
/people/person/place_of_birth	162	12.0	100.0	33.0
/location/country/administrative_divisions	424	0.2	N/A	0.0

Running Time

- MultiR
  - Training: 1 minute
  - Testing: 1 second
- SoloR
  - Training: 6 hours
  - Testing: 4 hours

Sentence-level extractions are efficient

Joint reasoning across sentences is computationally expensive

Distant supervision: conclusions

- Distant supervision extracts high-precision patterns for a variety of relations
- Can make use of 1000x more data than simple supervised algorithms
- Syntax features almost always help
- The combination of syntax and lexical features is sometimes even better
- Syntax features are probably most useful when entities are far apart, often when there are modifiers in between

Relation extraction: 5 easy methods

1. Hand-built patterns
2. Bootstrapping methods
3. Supervised methods
4. Distant supervision
5. Unsupervised methods

DIRT (Lin & Pantel 2003)

- DIRT = Discovery of Inference Rules from Text
- Looks at MINIPAR dependency paths between noun pairs
  - N:subj:V←find→V:obj:N→solution→N:to:N
  - i.e., X finds solution to Y
- Applies ”extended distributional hypothesis”
  - If two paths tend to occur in similar contexts, the meanings of the paths tend to be similar.
- So, defines path similarity in terms of cooccurrence counts with various slot fillers
- Thus, extends ideas of (Lin 1998) from *words* to *paths*

DIRT examples

The top-20 most similar paths to “X solves Y”:

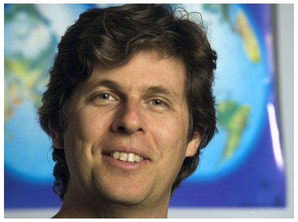
Y is solved by X
X resolves Y
X finds a solution to Y
X tries to solve Y
X deals with Y
Y is resolved by X
X addresses Y
X seeks a solution to Y
X do something about Y
X solution to Y

Y is resolved in X
Y is solved through X
X rectifies Y
X copes with Y
X overcomes Y
X eases Y
X tackles Y
X alleviates Y
X corrects Y
X is a solution to Y

Ambiguous paths in DIRT

- X addresses Y
  - I addressed my letter to him personally.
  - She addressed an audience of Shawnee chiefs.
  - Will Congress finally address the immigration issue?
- X tackles Y
  - Foley tackled the quarterback in the endzone.
  - Police are beginning to tackle rising crime.
- X is a solution to Y
  - (5, 1) is a solution to the equation $2x - 3y = 7$
  - Nuclear energy is a solution to the energy crisis.

TextRunner (Banko et al. 2007)



1. Self-supervised learner: automatically labels +/– examples & learns a crude relation extractor
2. Single-pass extractor: makes one pass over corpus, extracting candidate relations in each sentence
3. Redundancy-based assessor: assigns a probability to each extraction, based on frequency counts

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"

Step 3: Redundancy-based assessor

- Collect counts for each simplified tuple

  ⟨*scientists*, *are studying*, *stars*⟩ → 17

- Compute likelihood of each tuple
  - given the counts for each relation
  - and the number of sentences
  - and a combinatoric balls-and-urns model [Downey et al. 05]

$$P(x \in C | x \text{ appears } k \text{ times in } n \text{ draws}) \approx \frac{1}{1 + \frac{|E|}{|C|}\left(\frac{p_E}{p_C}\right)^k e^{n(p_C - p_E)}}$$

TextRunner demo

http://www.cs.washington.edu/research/textrunner/

(Note that they've re-branded TextRunner as ReVerb,
but it's largely the same system.)

TextRunner examples

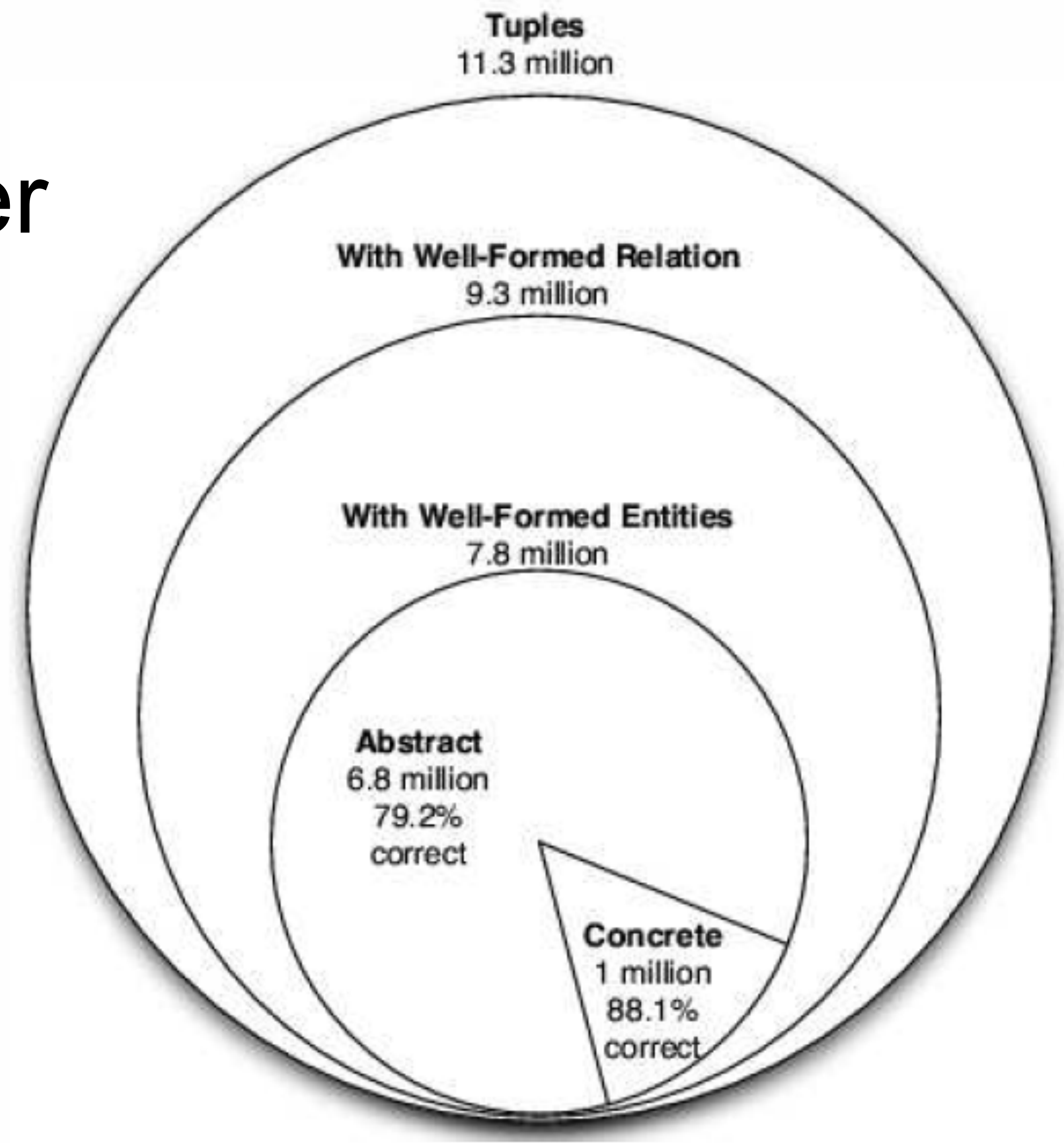
Probability	Count	Arg1	Predicate	Arg2
0.98	59	Smith	invented	the margherita
0.97	49	Al Gore	invented	the Internet
0.97	44	manufacturing plant	first invented	the automatic revolver
0.97	41	Alexander Graham Bell	invented	the telephone
0.97	36	Thomas Edison	invented	light bulbs
0.97	29	Eli Whitney	invented	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

slide from Oren Etzioni

TextRunner results

- 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⟩*

Evaluating TextRunner



Yao et al. 2012: motivation

- Goal: induce clusters of dependency paths which express the same semantic relation, like DIRT
- But, improve upon DIRT by properly handling semantic ambiguity of individual paths

Yao et al. 2012: approach

1. Extract tuples (entity, path, entity) from corpus
2. Construct feature representations of every tuple
3. Group the tuples for each path into sense clusters
4. Cluster the sense clusters into semantic relations

Extracting tuples

- Start with NYT corpus
- Apply lemmatization, NER tagging, dependency parsing
- For each pair of entities in a sentence:
  - Extract dependency path between them, as in Lin
  - Form a tuple consisting of the two entities and the path
- Filter rare tuples, tuples with two direct objects, etc.
- Result: 1M tuples, 500K entities, 1300 patterns

Feature representation

- Entity names, as bags of words, prefixed with "l:" or "r:"
  - ex: ("LA Lakers", "NY Knicks") => {l:LA, l:Lakers, r:NY, r:Knicks}
  - Using bag-of-words encourages overlap, i.e., combats sparsity

- Words between and around the two entities
  - Exclude stop words, words with capital letters
  - Include two words to the left and right

- Document theme (e.g. sports, politics, finance)
  - Assigned by an LDA topic model which treats NYTimes topic descriptors as words in a synthetic document

- Sentence theme
  - Assigned by a standard LDA topic model

Clustering tuples into senses

- Goal: group tuples for each path into coherent sense clusters
- Currently exploring multiple different approaches:
  - LDA-like topic models
  - Matrix factorization approaches
- Result: each tuple is assigned one topic/sense
- Tuples with the same topic/sense constitute a cluster

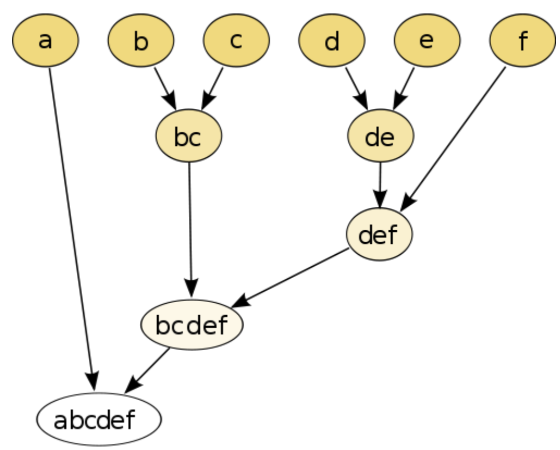
Sense cluster examples

Path	20:sports	30:entertainment	25:music/art
A play B	Americans, Ireland Yankees, Angels Ecuador, England Redskins, Detroit Red Bulls, F.C. Barcelona	Jean-Pierre Bacri, Jacques Rita Benton, Gay Head Dance Jeanie, Scrabble Meryl Streep, Leilah Kevin Kline, Douglas Fairbanks	Daniel Barenboim, recital of Mozart Mr. Rose, Ballade Gil Shaham, Violin Romance Ms. Golabek, Steinways Bruce Springsteen, Saints
doc theme	sports	music books television	music theater
sen theme	game yankees	theater production book film show	music reviews opera
lexical words	beat victory num-num won	played plays directed artistic	director conducted production
entity names	-	r:theater	r:theater r:hall r:york l:opera

Sense clusters for path “A play B”,
along with sample entity pairs and top features.

Clustering the clusters!

- Now cluster sense clusters from different paths into semantic relations — this is the part most similar to Lin & Pantel 2003
- Use Hierarchical Agglomerative Clustering (HAC)
- Start with minimal clustering, then merge progressively
- Uses cosine similarity between sense-cluster feature vectors
- Uses complete-linkage strategy



Semantic relation results

relation	paths
entertainment	A, who play B:30; A play B:30; star A as B:30
sports	lead A to victory over B:20; A play to B:20; A play B:20; A's loss to B:20; A beat B:20; A trail B:20; A face B:26; A hold B:26; A play B:26; A acquire (X) from B:26; A send (X) to B:26;
politics	A nominate B:39; A name B:39; A select B:39; A name B:42; A select B:42; A ask B:42; A choose B:42; A nominate B:42; A turn to B:42;
law	A charge B:39; A file against B:39; A accuse B:39; A sue B:39

Just like DIRT, each semantic relation has multiple paths.

But, one path can now appear in multiple semantic relations.

DIRT can't do that!

Evaluation against Freebase

System	Pairwise				B^3		
	Prec.	Rec.	F-0.5	MCC	Prec.	Rec.	F-0.5
Rel-LDA/300	0.593	0.077	0.254	0.191	0.558	0.183	0.396
Rel-LDA/1000	0.638	0.061	0.220	0.177	0.626	0.160	0.396
HAC	0.567	0.152	0.367	0.261	0.523	**0.248**	0.428
Local	0.625	0.136	0.364	0.264	0.626	0.225	0.462
Local+Type	0.718	0.115	0.350	0.265	**0.704**	0.201	0.469
Our Approach	**0.736**	**0.156**	**0.422**	**0.314**	0.677	0.233	**0.490**
Our Approach+Type	0.682	0.110	0.334	0.250	0.687	0.199	0.460

Automatic evaluation against Freebase

HAC = hierarchical agglomerative clustering alone

(i.e. no sense disambiguation — most similar to DIRT)

Sense clustering adds 17% to precision!