Automatic Entity Recognition and Typing for Massive Text Data

—A Phrase and Network Mining Approach—

XIANG REN (UIUC), AHMED EL-KISHKY (UIUC),

HENG JI (RPI), JIAWEI HAN (UIUC)

ACM SIGMOD 2016 CONFERENCE TUTORIAL

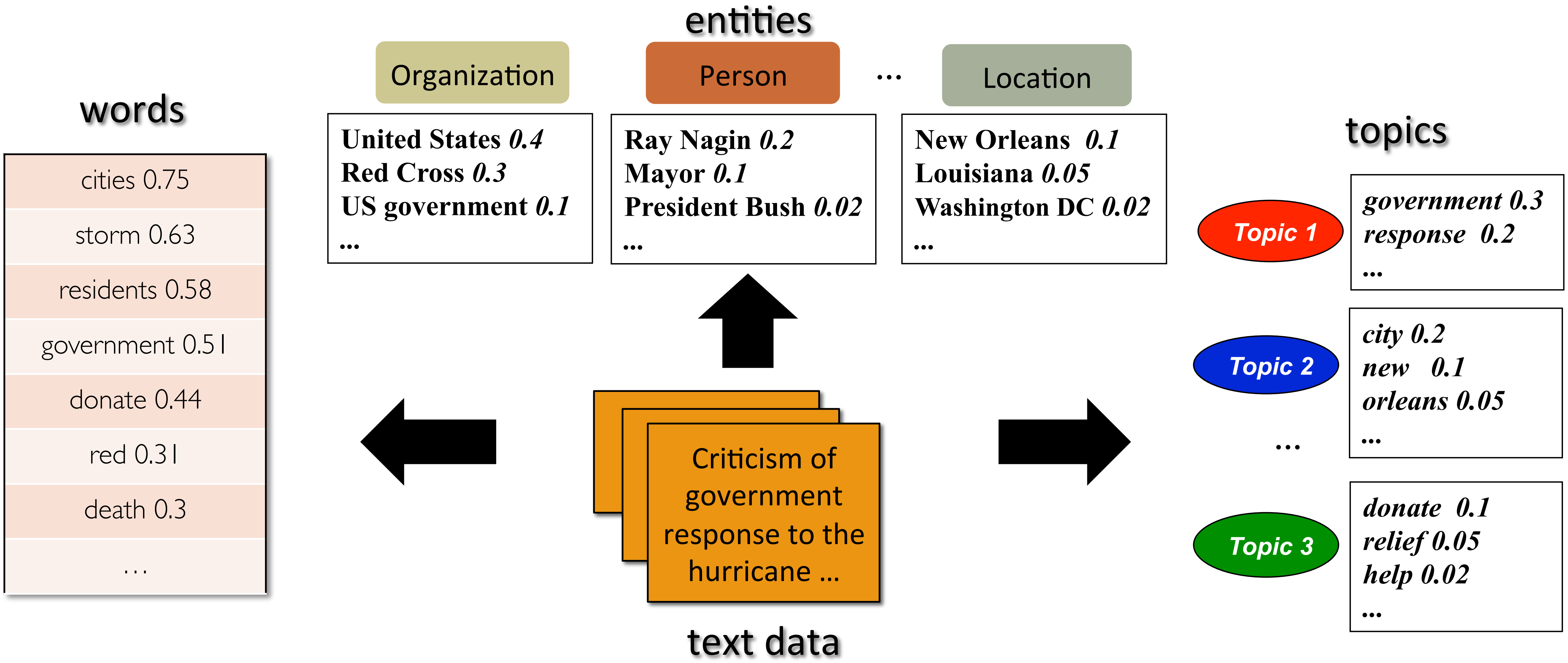
JULY 1, 2016

Outline

1. Introduction to entity recognition and typing
2. Entity recognition: An overview and phrase mining approach
3. Entity typing: An overview and network mining approach
4. Trends and research problems

Motivation of Entity Recognition and Typing

- Making sense of massive text data



Example: Linking Entities to Knowledge Base

The criticism consisted primarily of condemnations of mismanagement in response to Hurricane Katrina. Specifically, there was a delayed response to the flooding of New Orleans, Louisiana. New Orleans Mayor Ray Nagin was also criticized for failing to implement his evacuation plan.	Bush was criticized for not returning to Washington, D.C. from his vacation in Texas until after Wednesday afternoon. On the morning of August 28, the president telephoned Mayor Nagin to "plead" for a mandatory evacuation of New Orleans, and Nagin and Gov. Blanco decided to evacuate the city in response to that request

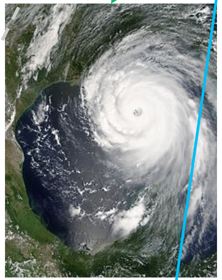
Criticism of government response to the hurricane …

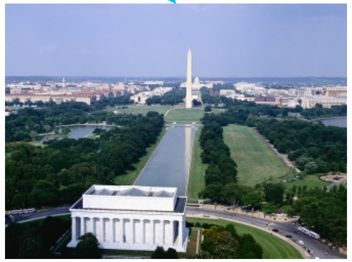
WIKIPEDIA
The Free Encyclopedia

Link entity mentions to knowledge base entries for in-depth entity information

Example: Linking Entities to Knowledge Base

The criticism consisted primarily of condemnations of mismanagement in response to Hurricane Katrina. Specifically, there was a delayed response to the flooding of New Orleans, Louisiana. New Orleans Mayor Ray Nagin was also criticized for failing to implement his evacuation plan.	Bush was criticized for not returning to Washington, D.C. from his vacation in Texas until after Wednesday afternoon. On the morning of August 28, the president telephoned Mayor Nagin to "plead" for a mandatory evacuation of New Orleans, and Nagin and Gov. Blanco decided to evacuate the city in response to that request





WIKIPEDIA
The Free Encyclopedia

“Entities” are what a large part of our knowledge is about

Motivation of Entity Recognition and Typing

- Organizing and exploring text data

The prevalence of unstructured text data → Structures are useful for knowledge discovery

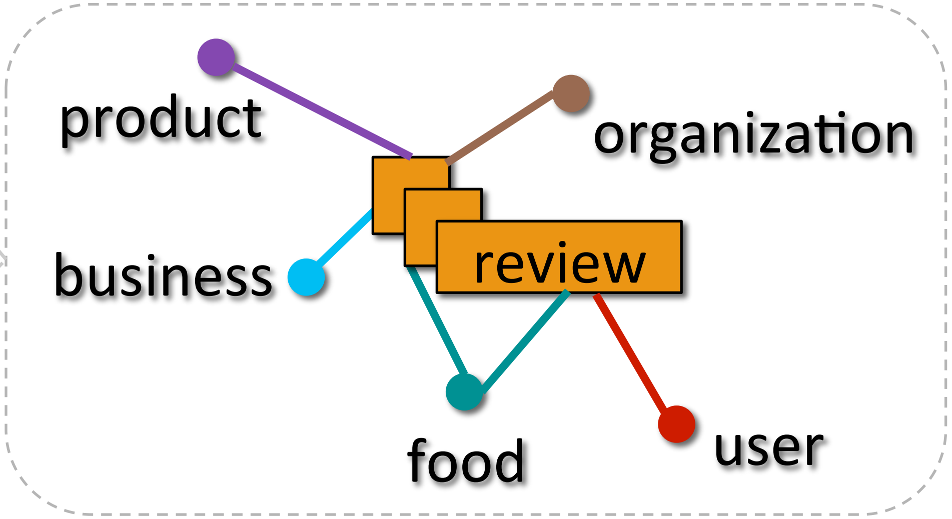
Too expensive to be structured by human:

Automated & scalable

Vast majority of the CEOs expressed frustration over their organization's inability to glean insights from available data
-- IBM study with1500+ CEOs

Example: Business Reviews

- Every year, hundreds of thousands papers are published
  - Loosely structured entities: business name, user, location
  - Unstructured data: review text
  - Extracted entities: food, product, organization

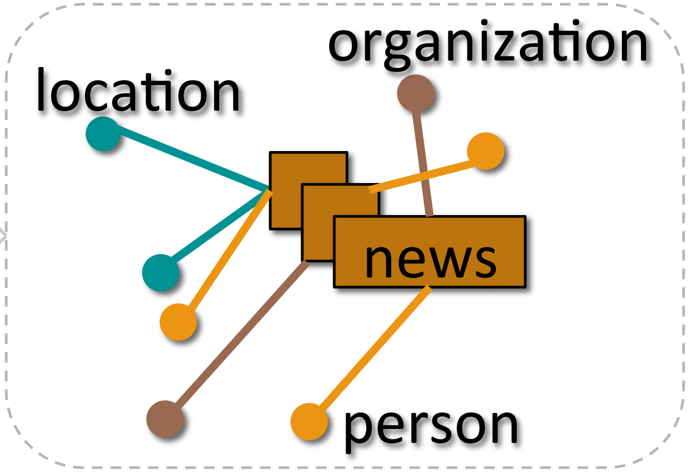


Example: Social Media

- Every second, >150K tweets are sent out
  - Loosely structured entities: users, hashtags, URLs, …
  - Unstructured data: tweet content
  - Extracted entities: person, location, organization, event

Example: News Articles

- Every day, >90,000 news articles are produced
  - Unstructured data: news content
  - Extracted entities: persons, locations, organizations, …



What Power can We Gain if More Structures Are Available?

- Structured database queries
- Information network analysis, …

Example: DBLP -- A Computer Science bibliographic database

26 Bib TeX XML Yizhou Sun, Jiawei Han, Charu C. Aggarwal, Nitesh V. Chawla: When will it happen?: relationship prediction in heterogeneous information networks. WSDM 2012: 663-672

Knowledge hidden in DBLP Network	Mining Functions
Who are the **leading** researchers on Web search?	Ranking
Who are the **peer** researchers of Jure Leskovec?	Similarity Search
Whom **will** Christos Faloutsos **collaborate with**?	Relationship Prediction
Which types of **relationships** are most **influential** for an author to decide her topics?	Relation Strength Learning
How was the field of Data Mining **emerged** or **evolving**?	Network Evolution
Which authors are **rather different** from his/her peers in IR?	Outlier/anomaly detection

What Is Entity Recognition and Typing (ER)

- **Identify token spans of entity mentions in text, and classify them into predefined set of types of interest**

  [Barack Obama] arrived this afternoon in [Washington, D.C.]. [President Obama]'s wife [Michelle] accompanied him

  [TNF alpha] is produced chiefly by activated [macrophages]

What Is Entity Recognition and Typing (ER)

- **Identify token spans of entity mentions in text, and classify them into predefined set of types of interest**

[Barack Obama] arrived this afternoon in [Washington, D.C.]. [President Obama]'s wife [Michelle] accompanied him

PERSON
LOCATION

[TNF alpha] is produced chiefly by activated [macrophages]

PROTEIN
CELL

Why Are Entity Recognition and Typing Challenging?

- Many entities may share the same surface name
  - "*Washington*" → Government? State? Sport team?...
  - Name ambiguity!

Why Are Entity Recognition and Typing Challenging?

- Many entities may share the same surface name
  - "*Washington*" → Government? State? Sport team?...
  - Name ambiguity!
- An entity may have multiple surface names
  - Barack Obama, Obama, President Obama, president, …

Why Are Entity Recognition and Typing Challenging?

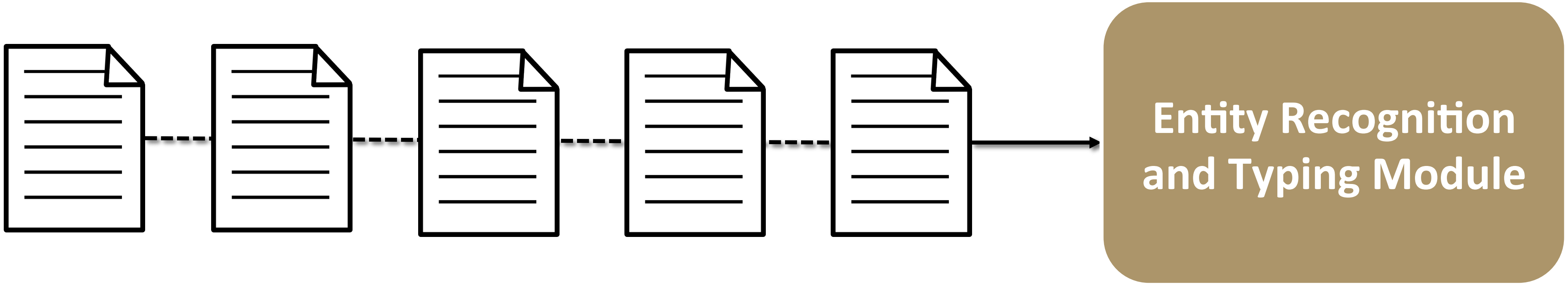
- Many entities may share the same surface name
  - "*Washington*" → Government? State? Sport team?...
  - Name ambiguity!
- An entity may have multiple surface names
  - Barack Obama, Obama, President Obama, president, …
- An entity may associate with multiple types
  - Person, Politician, US president, US congressman, …
  - Type ambiguity!

Why Are Entity Recognition and Typing Challenging?

- Many entities may share the same surface name
  - "*Washington*" → Government? State? Sport team?...
  - Name ambiguity!
- An entity may have multiple surface names
  - Barack Obama, Obama, President Obama, president, …
- An entity may associate with multiple types
  - Person, Politician, US president, US congressman, …
  - Type ambiguity!
- Entity may have grammatically informal name
  - "in-and-out"
- …

Scenario I: Sequential Text Stream as Input

- Process one text fragment (document) at a time



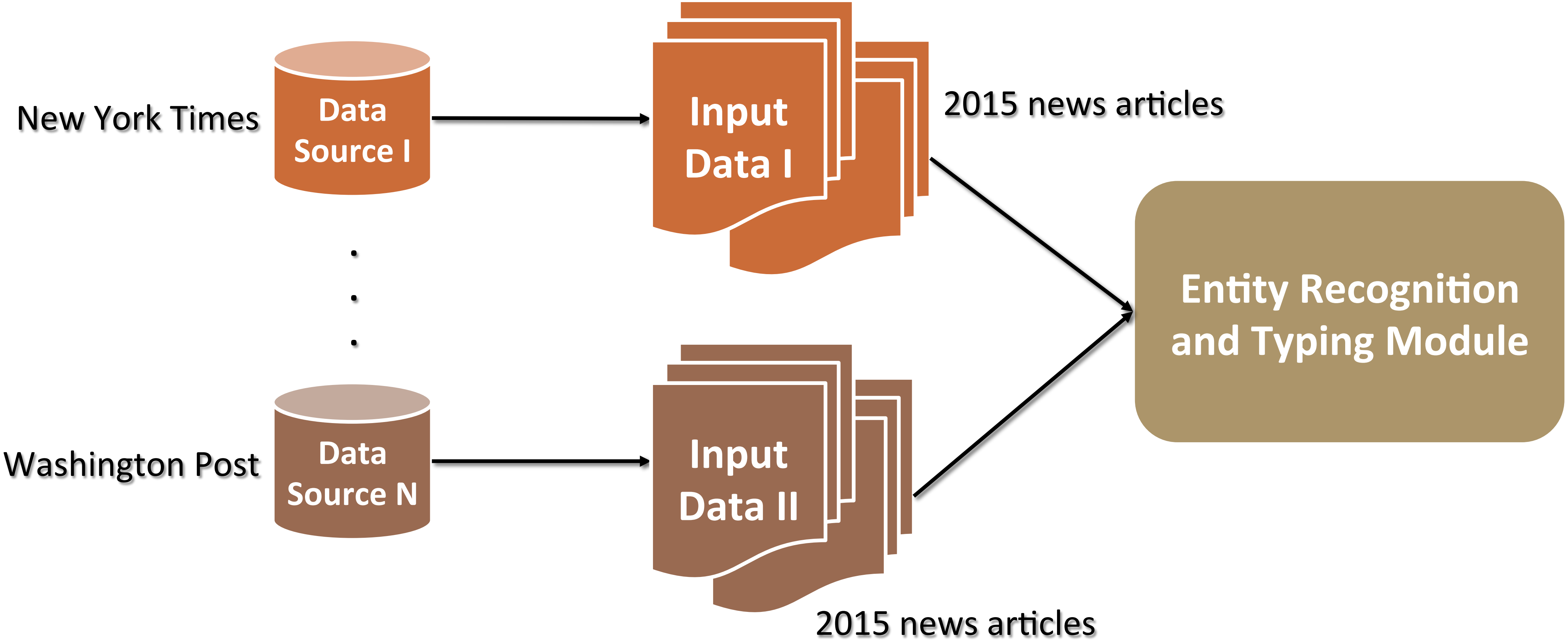
Web API

Query Search Intent

Question Answering

Scenario II: Large Text Data as Input

- Process large document collection(s) in a batch



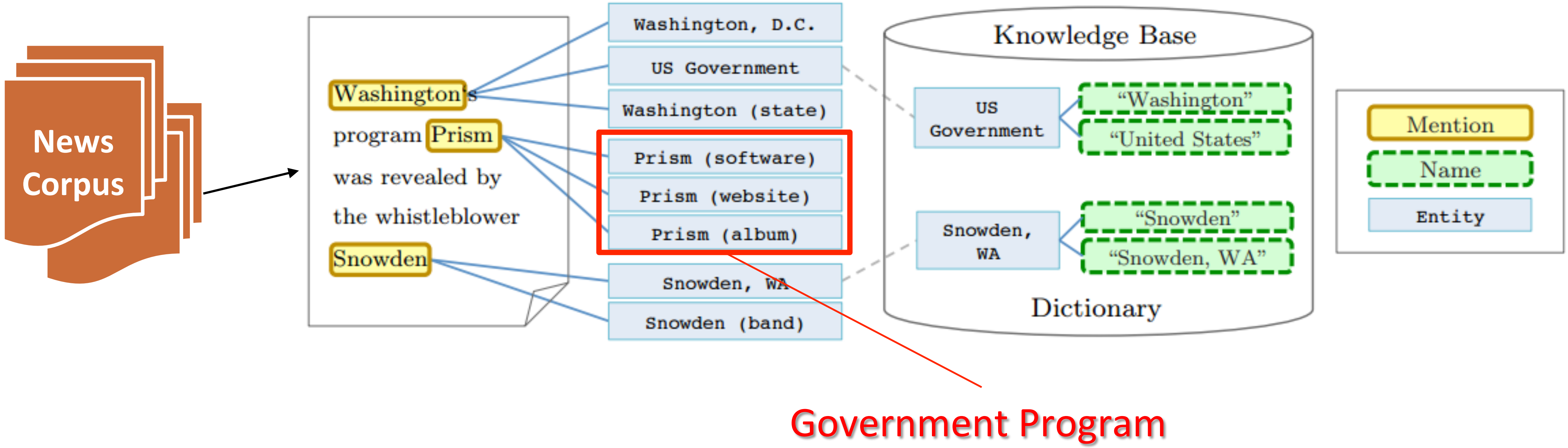
Example: Business Intelligence

- Top 10 active politicians regarding healthcare issues?
- Influential high-tech companies in Silicon Valley?

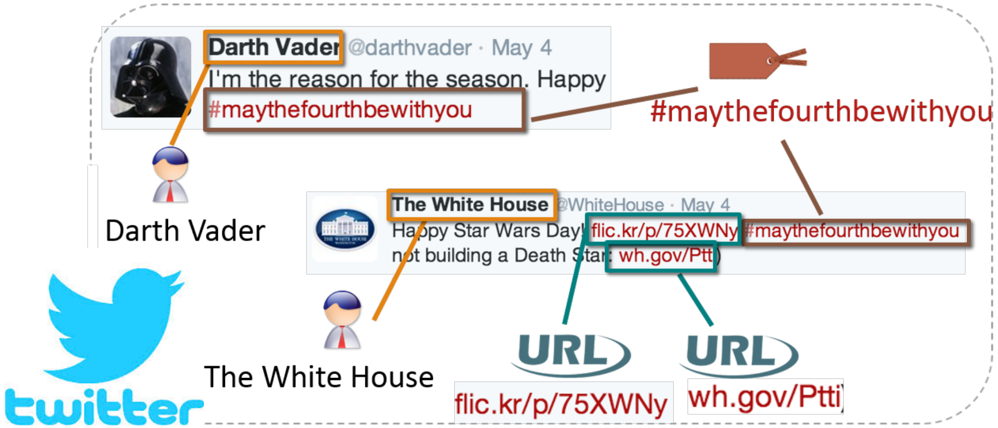
Type	Entity	Mention
politician		*Obama says more than 6M signed up for health care…*
high-tech company	APPLE	*Apple leads in list of Silicon Valley's most-valuable brands…*

Example: Knowledge-Base Population

- As the primitive step in identifying newly emerging entities from dynamic text corpora (e.g., news, microblogs, tweets)



PubMed
dblp
ACM DIGITAL LIBRARY



amazon
Angie's list
Google+ Local
dex
yp
Y!
yelp
bing
Google places
Epinions.com
Citysearch

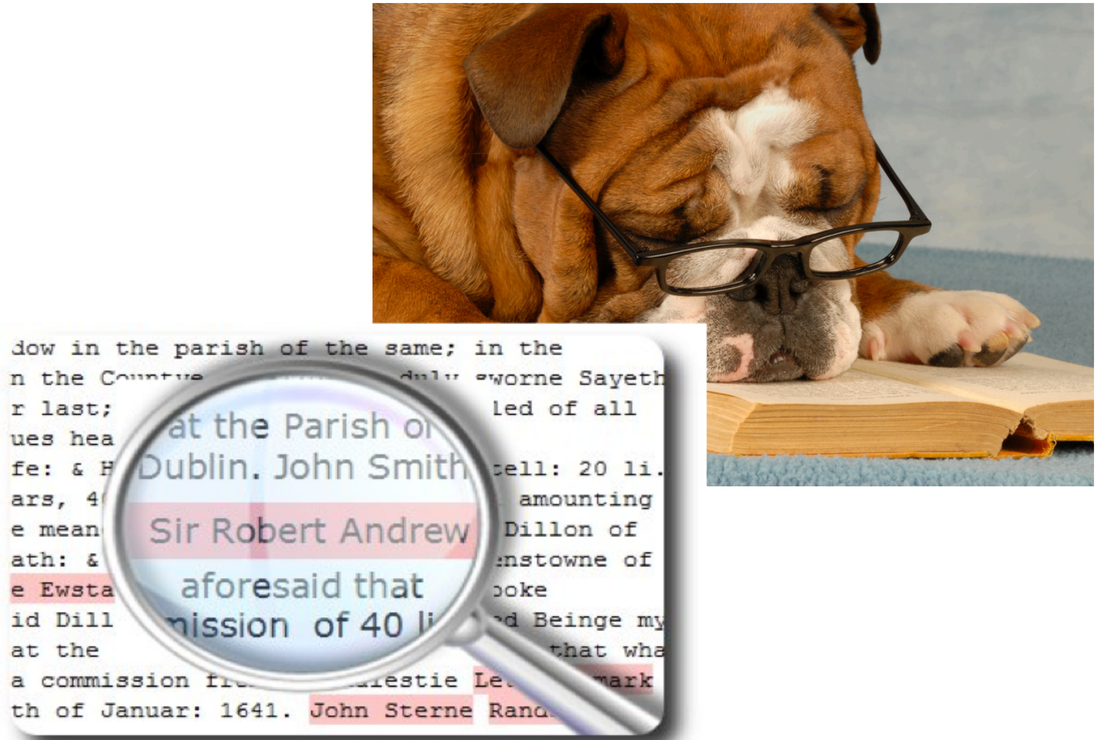
Focus of This Tutorial: Large Text Corpus

Characteristics of Text Corpus

- **General** vs. **specific** domain
  - News vs. social media content
  - Good amount of labeled data vs. few (no) open labeled data

Tagged datasets for named entity recognition tasks

1. 1999 Information Extraction – Entity Recognition Evaluation
   Notes: This dataset is apparently in public domain.
2. MUC-3 and MUC-4 datasets
   Notes: This dataset is apparently in public domain.
3. Language-Independent Named Entity Recognition at CoNLL-2003
   Notes: This dataset is a manual annotatation of a subset of RCV1 (Reuters C
   CoNLL site. The raw text of RCV1 documents must be requested from NIST
4. Message Understanding Conference (MUC) 6
   Notes: Consult the LDC Web site for current pricing and usage agreement.
5. Message Understanding Conference (MUC) 6 Additional News Text
   Notes: Consult the LDC Web site for current pricing and usage agreement.
6. Message Understanding Conference (MUC) 7
   Notes: Consult the LDC Web site for current pricing and usage agreement.



Characteristics of Text Corpus

- **Formal** vs. **informal** text
  - News vs. tweets, customer reviews
  - Regular grammars vs. irregular grammar, capitalization, punctuation

…
The prime minister's reaction was risky and foolish: he asked the Greek people to reject a proposal which, at the moment they voted on it, did not exist. The referendum supplied the result Mr Tsipras wanted but in many ways his position has deteriorated. His opportunistic manoeuvre infuriated almost every other European leader. The prospect of Grexit suddenly became more real.
…

MacroPolis @MacroPolis_gr · 3h
Total of 17 coalition MPs, including 2 ministers, have failed to support the gov't proposals: 7 absent, 8 abstained & 2 voted no #Greece
25 7

Zoe Mavroudi @zoemavroudi · 3h
Today we watched a European coup in our parliament. Government MPs who voted yes, were only translating from german. #Greece
19 16

Kathimerini English @ekathimerini · 4h
Proposals submitted by Greek coalition approved by 250 MPs. 32 vote against and 8 abstain #Greece
39 7

Yannis Koutsomitis @YanniKouts · 4h
#Greece | Conclusion:
-Bailout bill passes with s wide majority of 250 of 300 votes
-Gov't narrowly escapes collapse of coalition majority

Outline

1. Introduction to entity recognition and typing
2. Entity recognition: An overview and phrase mining approach
3. Entity typing: An overview and network mining approach
4. Trends and research problems

Entity Mention Detection

- Entity mention detection seeks to identify ***spans of tokens*** in text for analysis in whether they align to certain pre-defined categories such as:
  - names of people, organizations, locations, dishes, concepts, etc

__Barack Obama__ arrived this afternoon in __Washington, D.C.__.
__President Obama__'s wife __Michelle__ accompanied him

- To effectively detect these candidate, intuitively requires the underlying **grammatical structure of sentences** and answer such questions as:
  - which words go together as phrases, subject and object of verbs/verb phrases, etc
- Fortunately this is extensively studied in NLP!

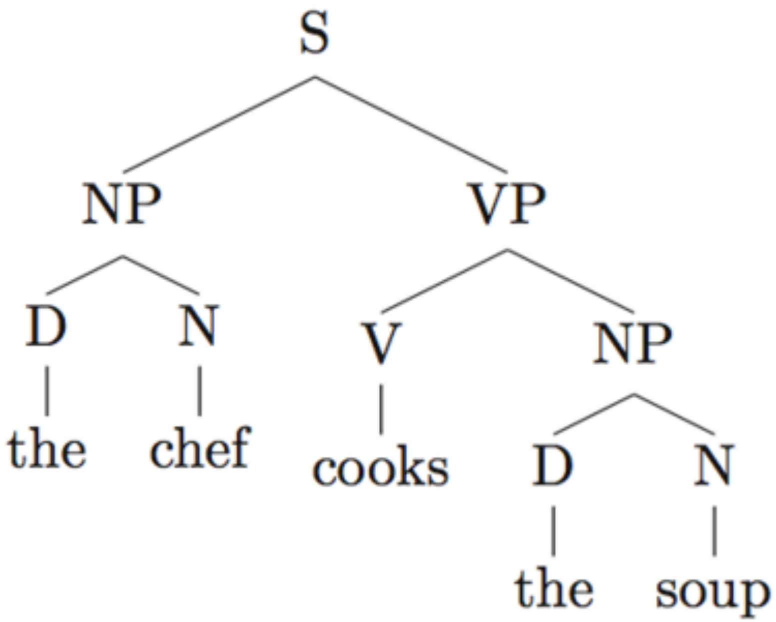
Full Sentence Parsing

- Partitioning sentences into grammatical text segments

Raw text sentence (string) → *Full-text Parsing* → Full parse tree (grammatical analysis)

Parsing segments input text sentences into parse trees. Noun Phrases indicate entity mention candidates

- *Full syntax understanding*



- *Low accuracy*
- *Adapts poorly to new domains (Twitter)*
- *Computationally Slow (Intractable on web-scale)*

Inefficiencies of Full Parsing

1. Parsing yields low accuracy in identifying entity mentions
2. Parsing requires non-trivial training data – manually curated
3. Parsing adapts poorly to new domains (e.g. twitter, biomedical, yelp)
4. Parsing is computationally slow. Cannot be applied on web-scale data

Motivates a family of “shallow” entity detection techniques.

Alternatives to Full Parsing: Direct Detection of Entity Mentions

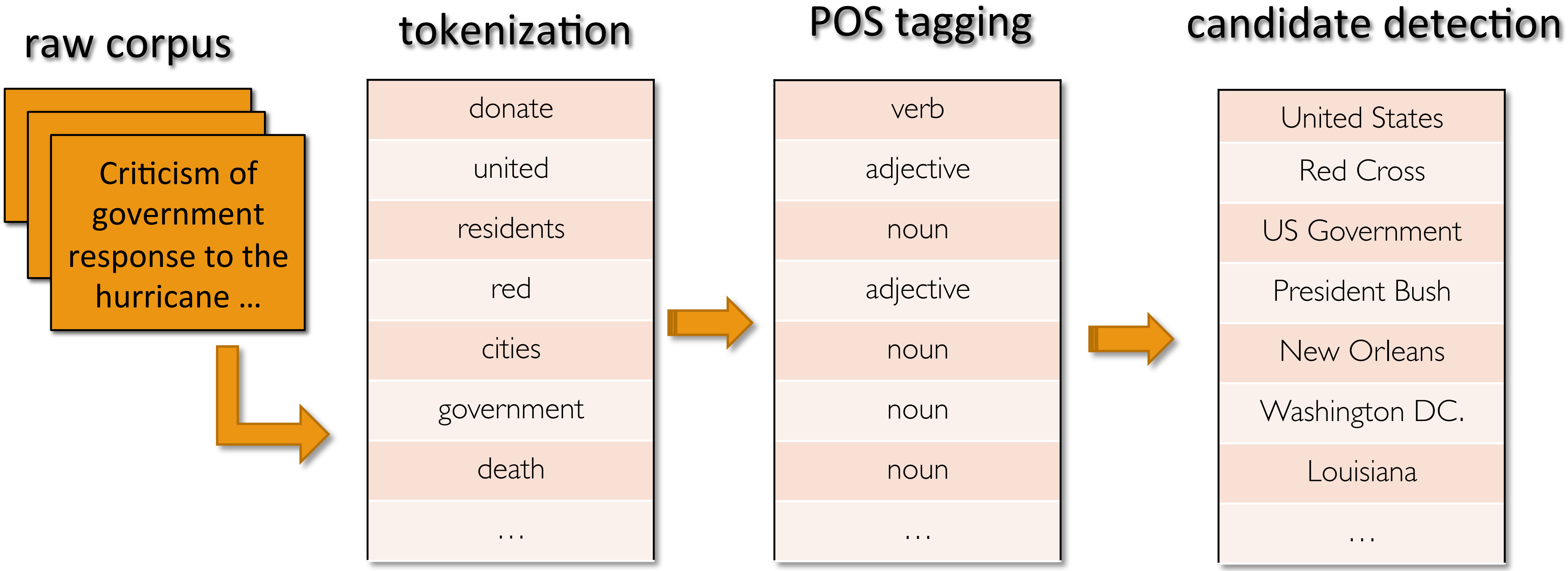
A. Supervised/Semi-supervised Entity Mention Detection

B. Unsupervised Entity Mention Detection

C. Weakly and Distantly Supervised Mention Detection

Entity Mention Pipeline

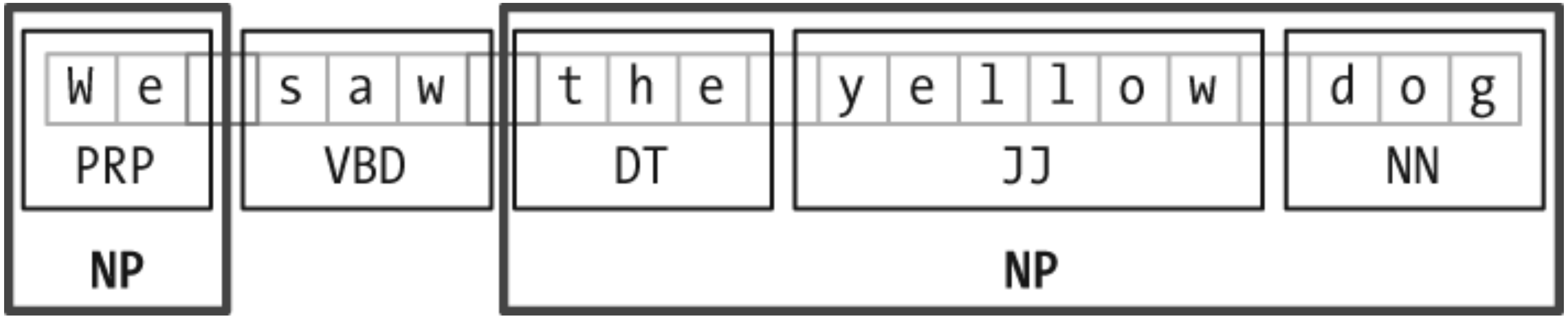
- Making sense of large text corpora



Segmentation → Part-of-speech tagging → Entity Mention Detection

Noun Phrase Chunking

1. Apply tokenization and part-of-speech tagging to each sentence
2. Search for noun phrase chunks



Things to think about

- *Not all phrases are useful for entity mentions*
- *Can other signals in addition to POS tags be helpful?*
- *Noun chunks often smaller than noun phrases*

Three Families of Methods

A. Supervised/Semi-supervised Entity Mention Detection

B. Unsupervised Entity Mention Detection

C. Weakly and Distantly Supervised Mention Detection

Supervised Entity Mention Detection

Assumptions

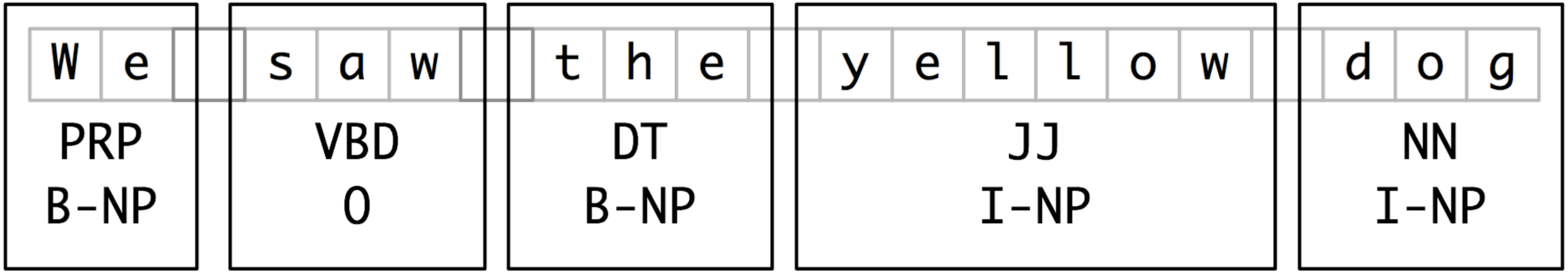
1. Unsupervised methods cannot possibly take into consideration the innumerable features, signals, and cues for entity mentions
2. Training data for entity mentions is more expensive than POS tagging, but less so than full parsing

- Training data consisting of chunked data can be used for supervised training of entity mention chunkers

[Barack Obama] arrived this afternoon in [Washington DC] . [President Obama]'s wife [Michelle] accompanied him

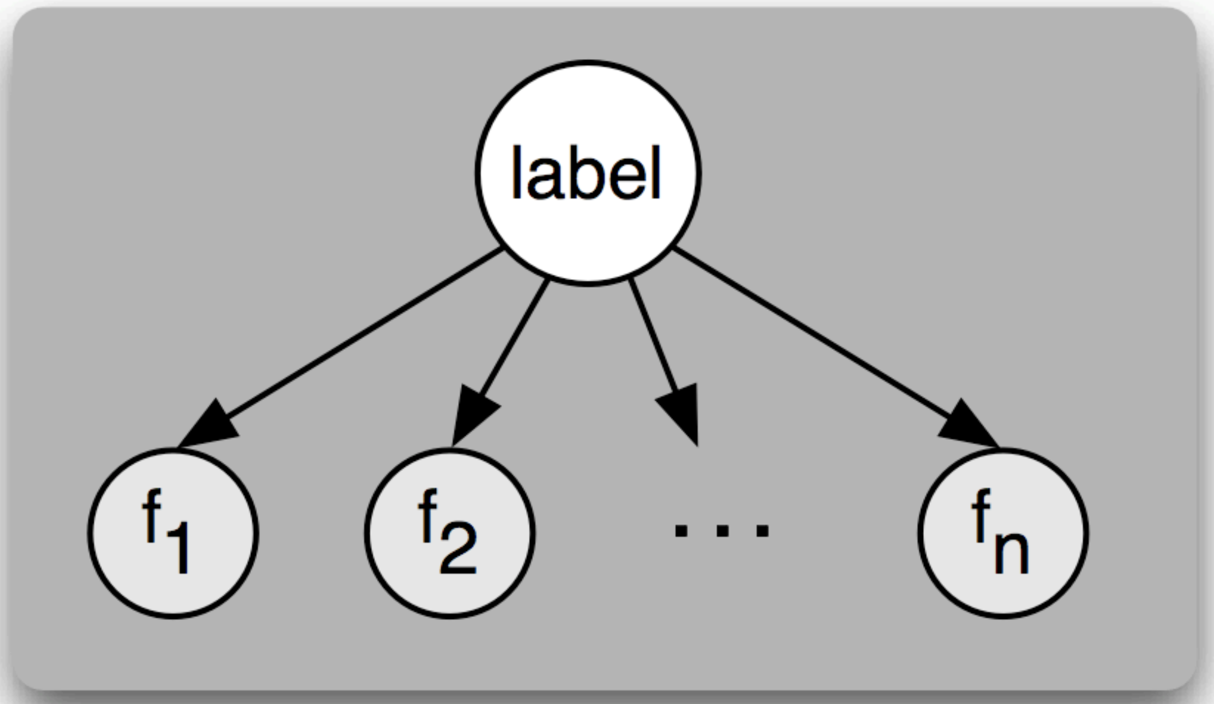
The I-O-B Representation

- (Inside – Outside – Beginning)
  - I – Denotes token inside of a chunk
  - O – Denotes tokens outside of a chunk
  - B – Denotes token at the beginning of a chunk



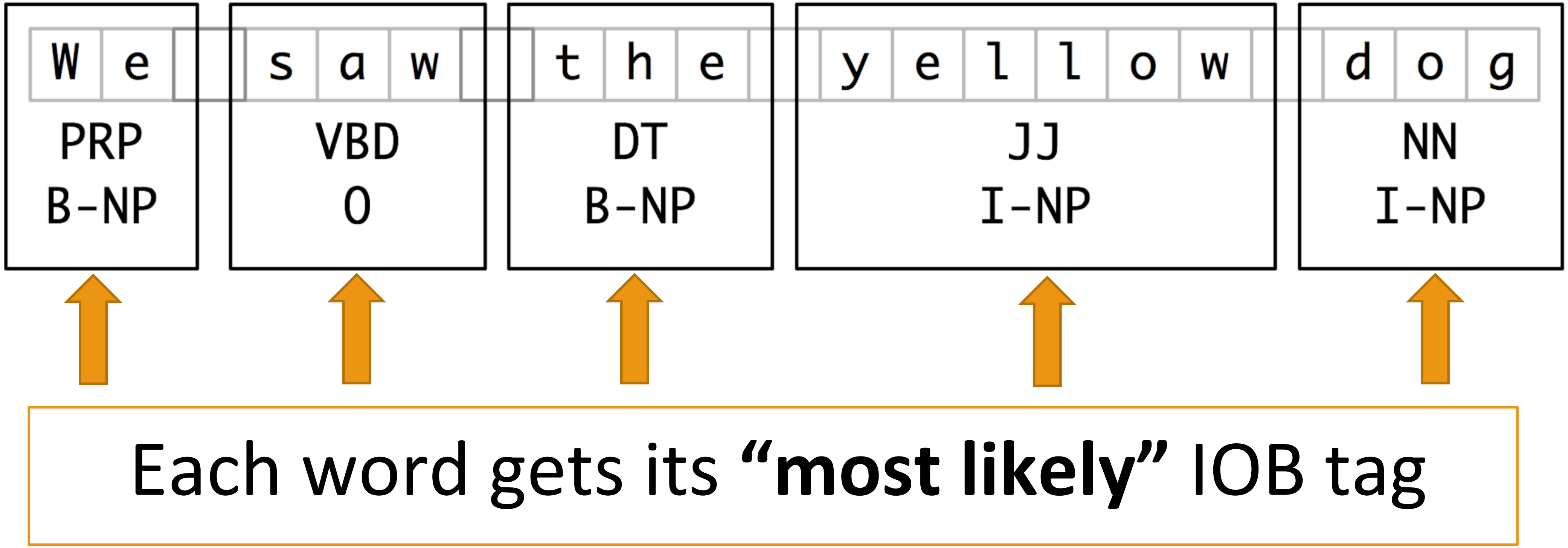
NP Chunkers as Classifiers

- Insights
  - Under the I-O-B Representation, each word should be tagged with its I-O-B label
  - Like POS Taggers, I-O-B taggers can be solved through standard classification methods such as Naïve Bayes or more sophisticated methods



Unigram Chunking

- Given each word's POS tag, one can directly classify each word to its IOB chunk

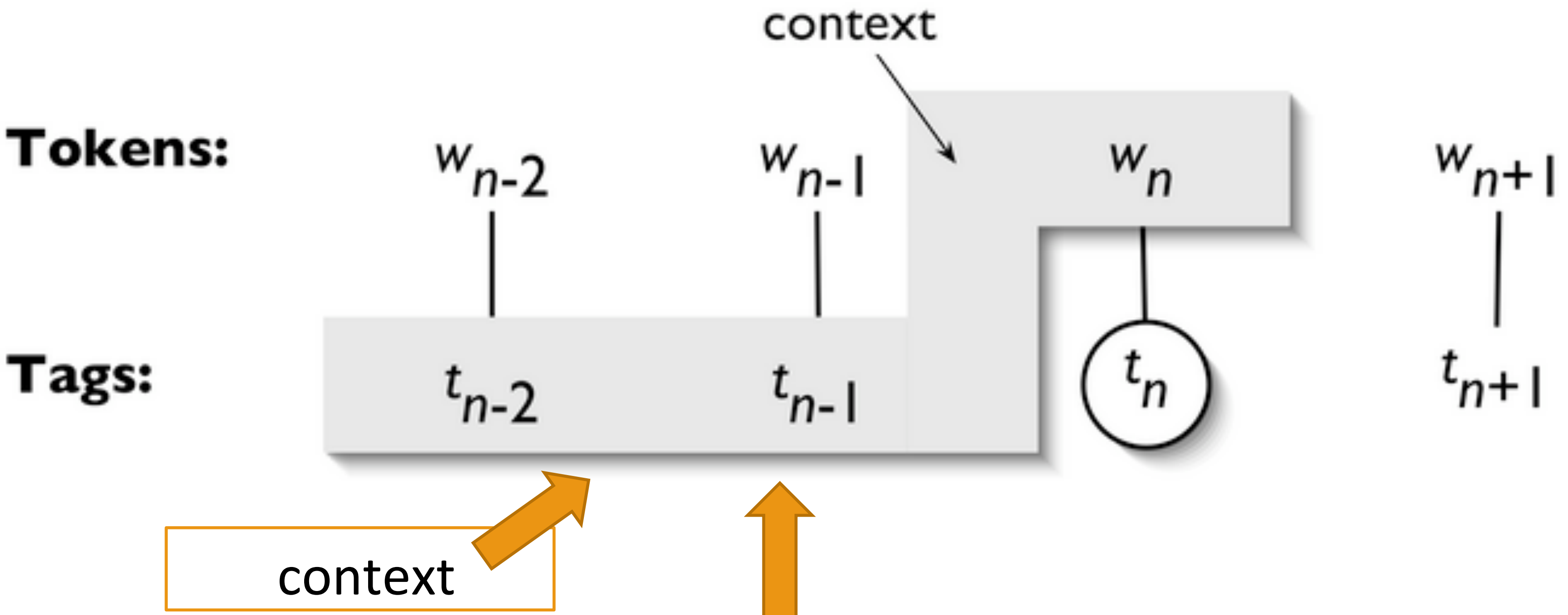


IOB Accuracy: 93% Precision: 80% Recall: 87% F1: 83%

Pretty good! Can we do better?

Higher-Order Chunkers

- To improve beyond using only the current unigram in isolation of any context, we can look at higher order contexts.



Results for Bigram Chunker

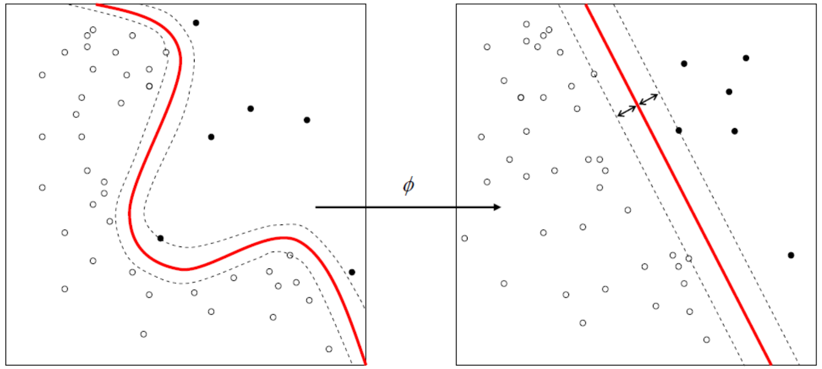
Measure	Score
IOB Accuracy	93%
Precision	82%
Recall	87%
F1	85%

Improvement over unigram

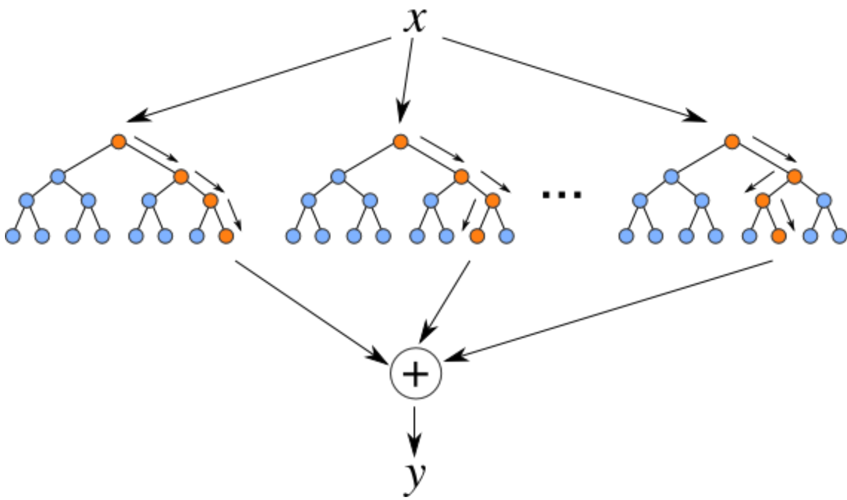
This IOB chunk of a word chosen in consideration of tags of previous two

Classical/Non-sequential Classifiers

- These methods consider higher-order features to classify each word into its appropriate I-O-B tag. Any classifier can be used for this task including:
  - Support Vector Machines
  - Ensemble Methods
  - Naïve Bayes
  - Logistic Regression
  - Etc.



Support Vector Machine



Random Forest

Support Vector Machine Chunking

- Weighted vote of 8 Support Vector Machines trained on 8 distinct chunk representations

		→	Direction	→	
Word:	w_{i-2}	w_{i-1}	w_i	w_{i+1}	w_{i+1}
POS:	t_{i-2}	t_{i-1}	t_i	t_{i+1}	t_{i+1}
Chunk:	c_{i-2}	c_{i-1}	$\boxed{c_i}$		

I-O-B Representation (4 variants)

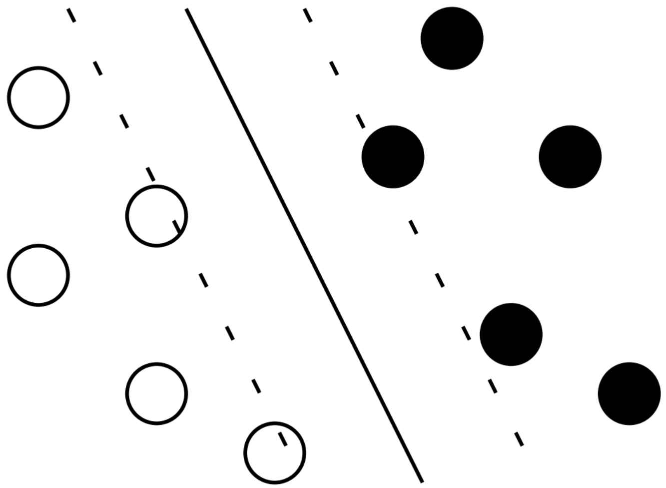
Two Directions:

- **Forward Parsing**
- **Backward Parsing**

4 * 2 = 8

Cross validation to set weights

Forward Parsing Context



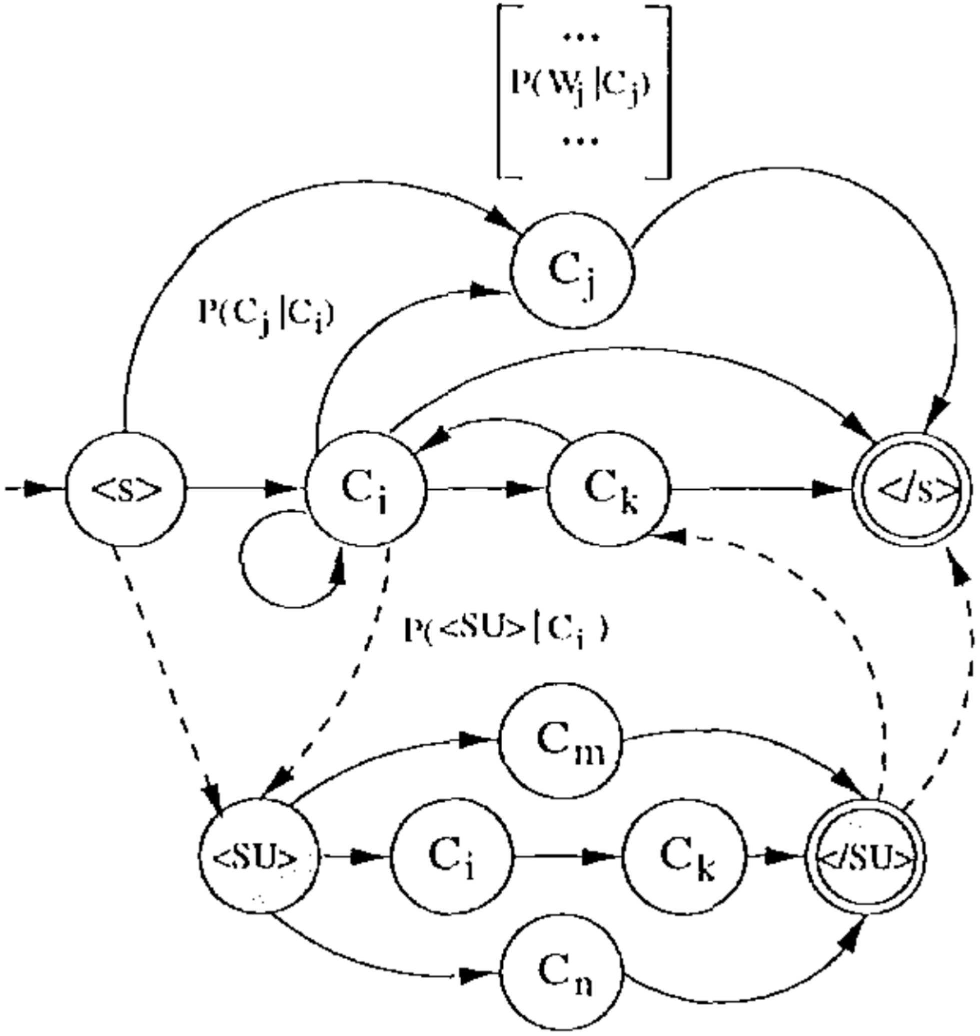
Joint Tagging & Chunking with Bigrams

Three separate models are learned

1. **Contextual Language Model**
   - A smoothed bigram model learnt from the sequences of part-of-speech tags and chunk descriptors in a training corpus
2. **Chunking Model**
   - Smoothed bigram model learnt from the sequences of part-of-speech tags corresponding to chunks in the training corpus
3. **Lexical Probabilities**
   - Estimated using word frequencies, tag frequencies, word-per-tag frequencies (smoothing is performed for unseen categories)

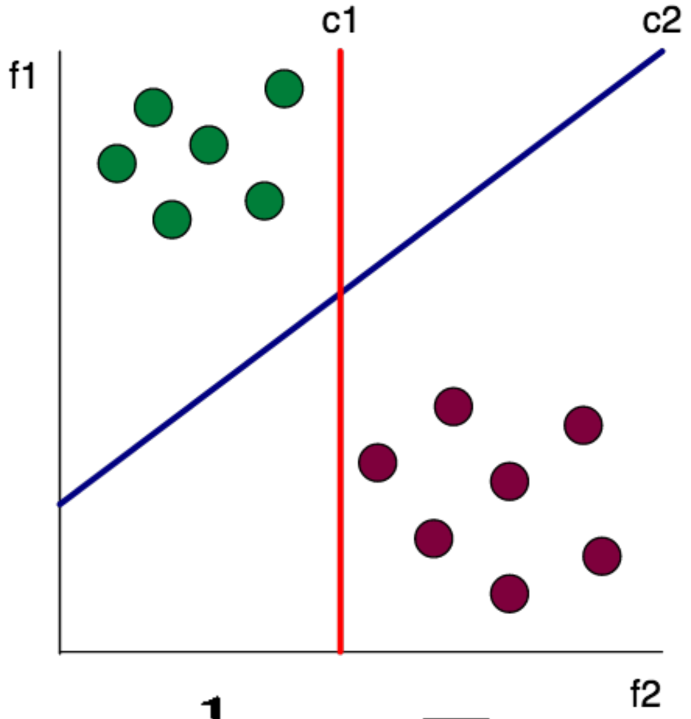
Joint Tagging & Chunking with Bigrams

- Combines different knowledge sources to obtain corresponding POS Tags and Chunks
- Once all the LM's have been learnt, they are combined into an Integrated LM
  - Shows possible concatenations of lexical tags, syntactical units, and their transition probabilities / lexical probabilities
- Tagging/shallow parsing performed by using dynamic programming (Viterbi) to find the maximum probability sequence of states



Maximum Entropy Classifier

- Maximum Entropy classifiers are based on the assumption that the probability distribution which best represents the current state of knowledge is the one with largest entropy
  - External Features
    - Current Word
    - POS tag of current word
    - Surrounding words
    - POS tags of surrounding words
  - Model Generated Features
    - Chunk tags of previous words



$$P(w|h) = \frac{1}{Z(h)} \cdot e^{\sum_i \lambda_i f_i(h,w)}$$

Ranking Algorithms for Entity Mentions

Insight

- Reranking the top-N hypotheses from a maximum-entropy tagger may improve recovery of entity boundaries from text corpora

Methodology

1. Use a state-of-the-art max-ent tagger to generate top N segmentations
2. Re-rank these segmentations using global features and proposed methods (boosting and voted perceptron)

Results for Precision/Recall/F-Measure

Method	Precision	Recall	F-Measure
Max-Ent	84.4	86.3	85.3
Boosting	87.3(18.6)	87.9 (11.6)	87.6 (15.6)
Voted Perceptron	87.3(18.6)	88.6 (16.8)	87.9 (17.7)

Parenthesis indicate relative improvement in error rate.

Global Features

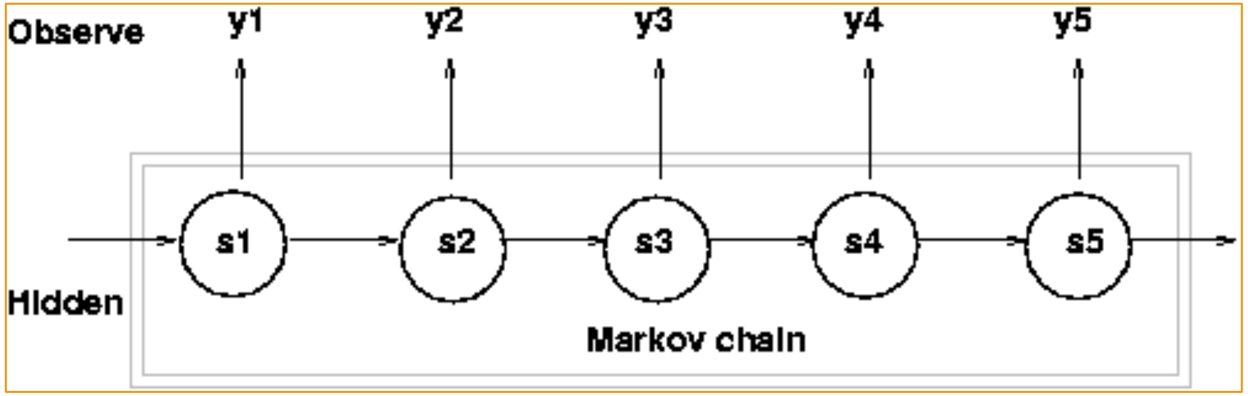
- May be tied to each candidate segmentation's boundaries, Quotation marks, Number of uppercase words, etc.

Classifiers for Sequential Data Models

- Moving forward from classical classifiers that use only features for classification, many state-of-the-art methods apply sequential data models to detect these temporal patterns

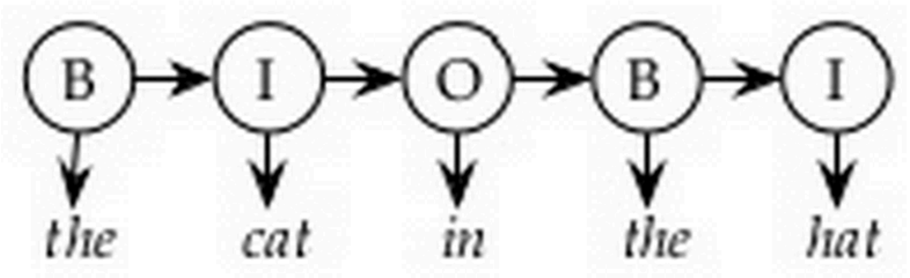
- Successfully applied to part-of-speech tagging, sequential data models posit that sequential observations are related to each other such as through a Markov process, in contrast to traditional models that assume independence

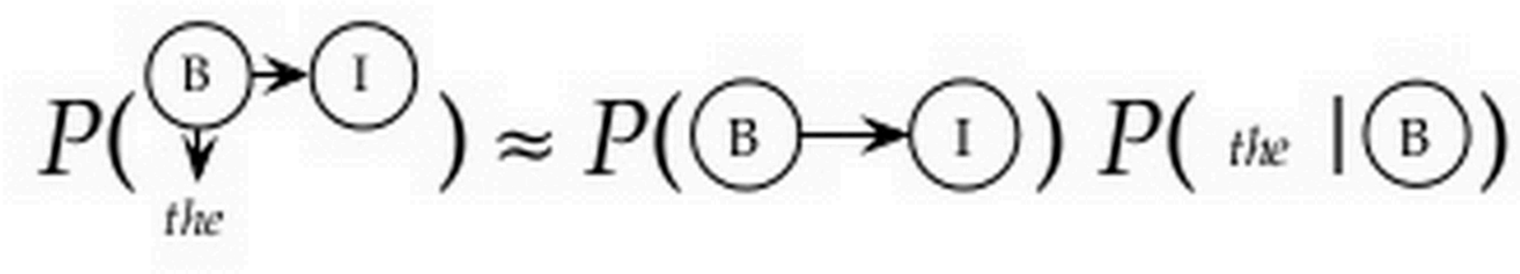
In a Markov Model, hidden states and their transitions explain observations.



Hidden Markov Models for Mention Detection

- A HMM is a finite state automaton with stochastic transitions defined on states and observations
  - For state **s**, p(**s**|**s'**)
  - For observation **o**, p(**o**|**s**)
- Markov Assumption, Stationary Assumption, and Output Independence Assumption
- The task resorts to inferring most likely latent states given observations (words)





Hidden Markov Models for Mention Detection

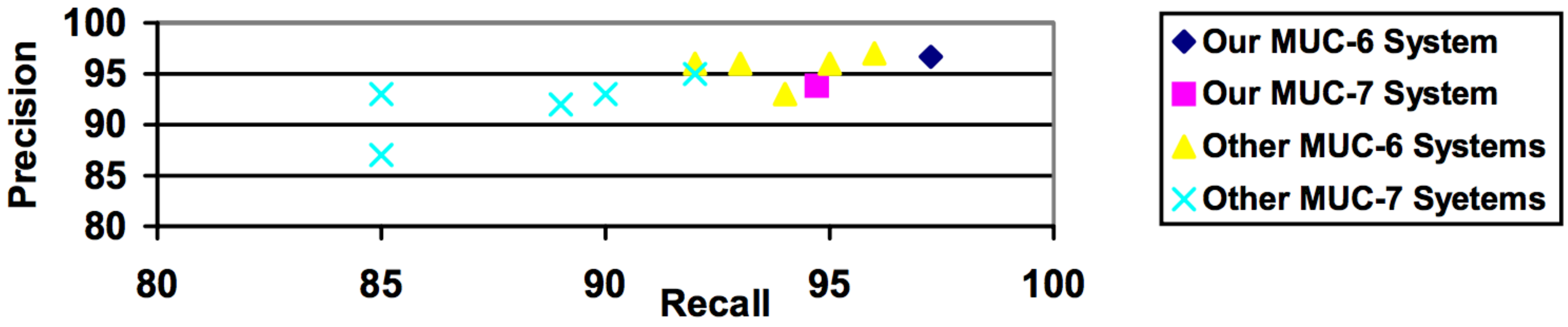


Figure 1: Comparison of our system with others on MUC-6 and MUC-7 NE tasks

Effect of adding additional features

Composition	F	P	R
$f = f^1$	77.6	81.0	74.1
$f = f^1 f^2$	87.4	88.6	86.1
$f = f^1 f^2 f^3$	89.3	90.5	88.2
$f = f^1 f^2 f^4$	92.9	92.6	93.1
$f = f^1 f^2 f^3 f^4$	94.1	93.7	94.5

Shortcomings of HMMs

Shortcomings of HMM

1. HMM's maximize likelihood of observation sequence (metric divergence problem)
2. Don't consider non-independent observational variables or difficult to enumerate observational variables

Addressing HMM Shortcomings

1. Instead of modeling the joint probability of state and observation $p(O_T,S_T)$, model the discriminative probability, $p(S_T|O_T)$.
2. This allows for a plethora of features that can be used
   - words
   - line length
   - grammatical
   - contextual

$$p(O_T,S_T) \quad \text{vs} \quad p(S_T|O_T)$$

Generative

Discriminative

Maximum Entropy Markov Models

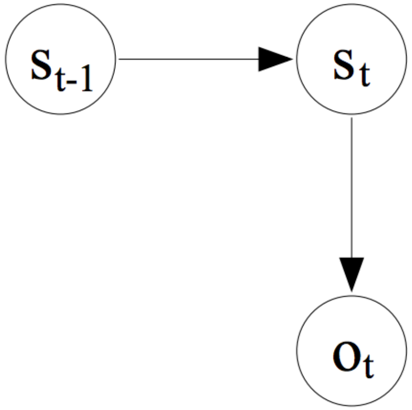
Max-Ent Markov Models

- Conditional model represents the probability of reaching a state given an observation and the previous state
- Conditional probabilities are specified by exponential models based on arbitrary observation features

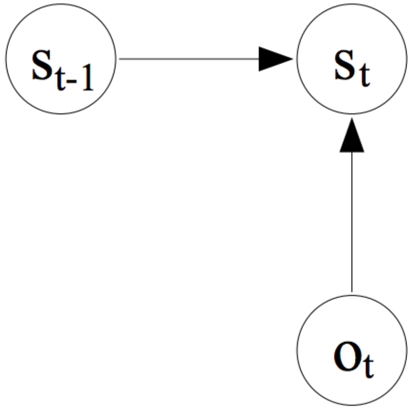
Learning

- Given **O** and **S**, find *M* such that p(**S|O,***M*) is maximized (maximum likelihood)

HMM



Conditional Maximum Entropy Markov Model



Maximum Entropy Markov Models

1. **ME-Stateless:** 24 Features, no context
2. **TokenHMM:** Traditional, fully-conencted HMM (model switches states at line boundaries)
3. **FeatureHMM:** Similar to TokenHMM but lines are converted into features
4. **Maximum Entropy Markov Model:**

Learner	*COAP*	*SegPrec*	*SegRecall*
ME-Stateless	0.520	0.038	0.362
TokenHMM	0.865	0.276	0.140
FeatureHMM	0.941	0.413	0.529
MEMM	0.965	0.867	0.681

COAP: COo-occurrence agreement probability

SegPrec: Segmentation Precision

SegRegall: Segmentation Probability

Conditional Random Fields for Entity Mentions

Insights

- Discriminative models often achieve better results than fully generative models (HMM)
- As such training Conditional Random Fields is natural method for effective noun-phrase chunking

Best of both words:

- Like classification models, they can accommodate many statistically correlated features of the inputs, and they are trained discriminatively
- Like generative models, they can trade off decisions at different sequence positions to obtain a globally optimal labeling

Conditional Random Fields for Entity Mentions

CRF's outperform other state-of-the-art methodologies including **MEMM** and **SVM**

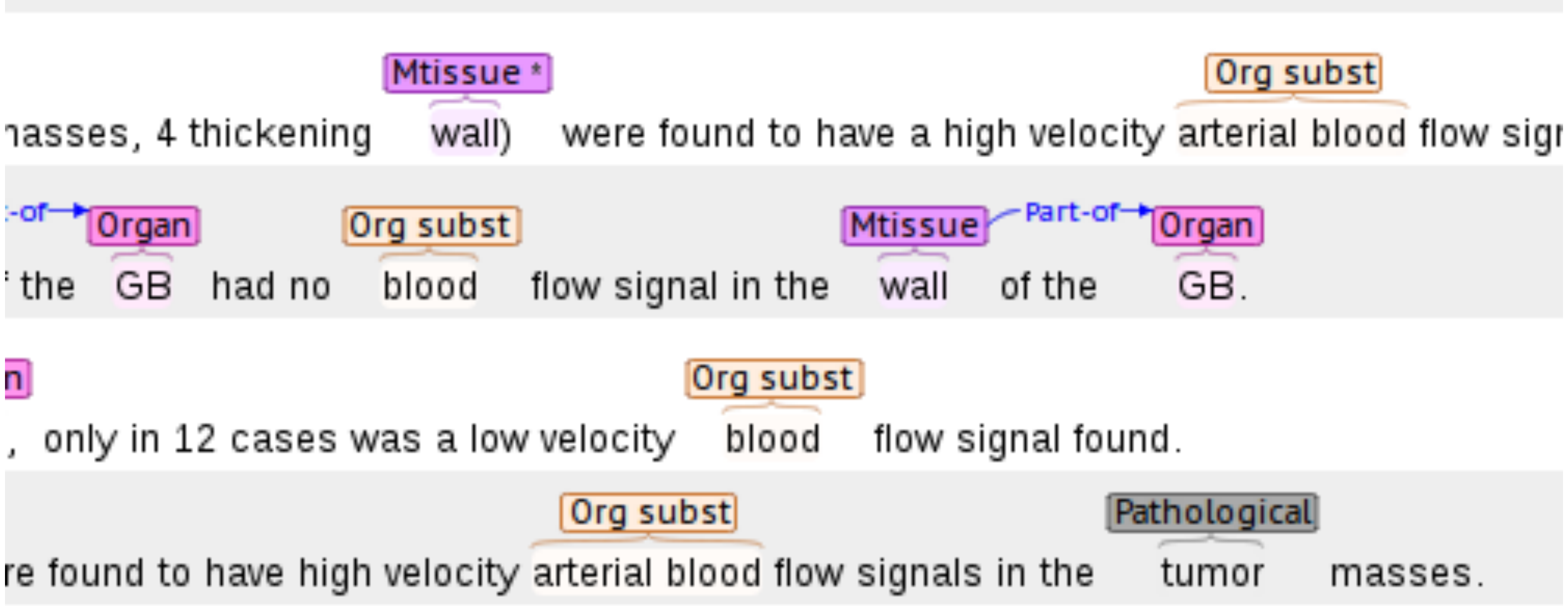
Model	F score
SVM combination (Kudo and Matsumoto, 2001)	94.39%
CRF	94.38%
Generalized winnow (Zhang et al., 2002)	93.89%
Voted perceptron	94.09%
MEMM	93.70%

Application: Anatomical Entity Mention Detection

Anatomical entities such as ***kidney, muscle, blood*** are prevalent in the life-science and biomedical literature

- Detection of these entities is therefore quite invaluable in the automatic analysis of the structure of these ***domain texts***

- CRF for Entity Mention
- Meta-Map for Entity mention
- Combination Method



Semi-Markov CRF

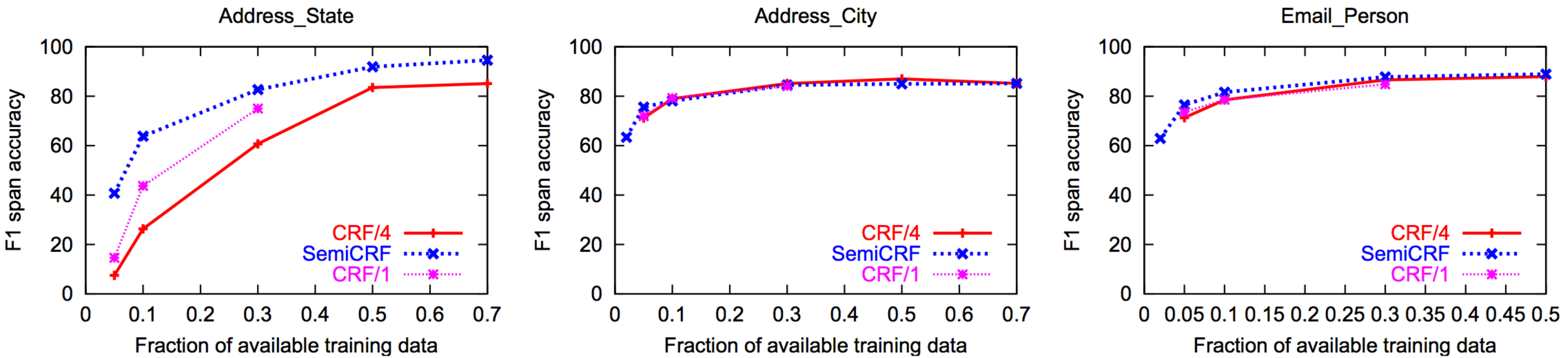
Relaxing the Markov Assumption

- Semi-Markov models extend traditional HMMs by relaxing the Markov assumption and allowing a state $\mathbf{S}_i$ to persist for a non-unit length of time
- These are also conditionally trained and therefore are discriminative and not generative

Features Used

- Indicators for key words within 3-word window
- Capitalization/letter patterns (digits, etc.) within 3-word window
- External dictionary for dictionary-derived features

Semi-Markov CRF



	CRF/1			CRF/4			semi-CRF
	$L=1$	$L=2$	$L=3$	$L=1$	$L=2$	$L=3$	
Address_State	20.8	20.1	19.2	15.0	16.4	16.4	**25.6**
Address_City	70.3	71.0	71.2	73.2	73.9	73.7	**75.9**
Email_persons	67.6	63.7	66.7	70.9	70.7	70.4	**72.2**

F1 values for different order CRFs

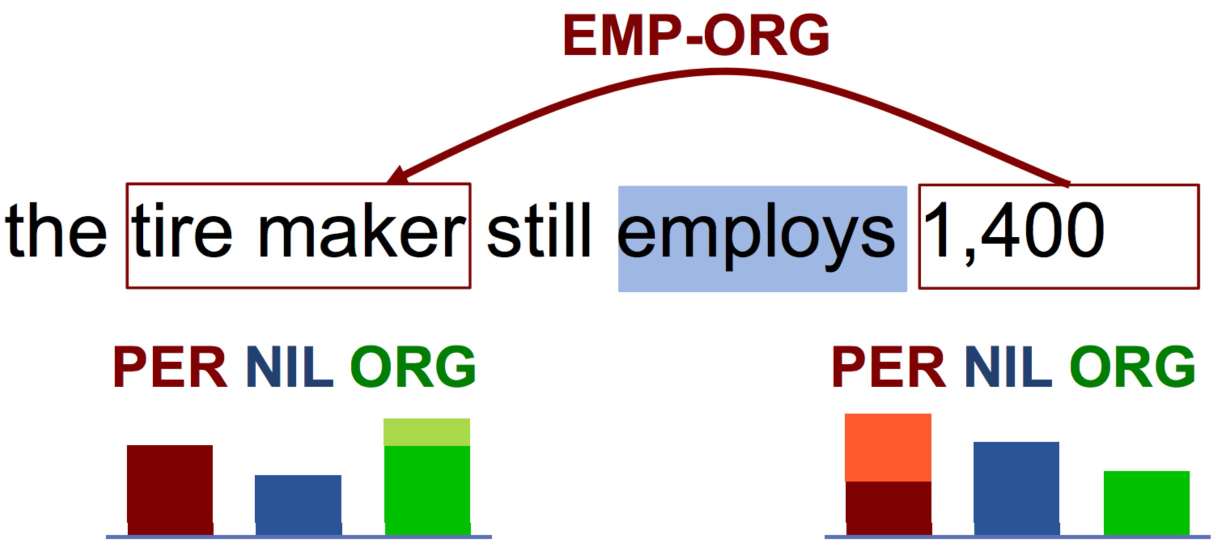
Incremental Joint Entity and Relation Detection

Insight

- Jointly extract both entities and relations to improve both subtasks

Joint Extraction

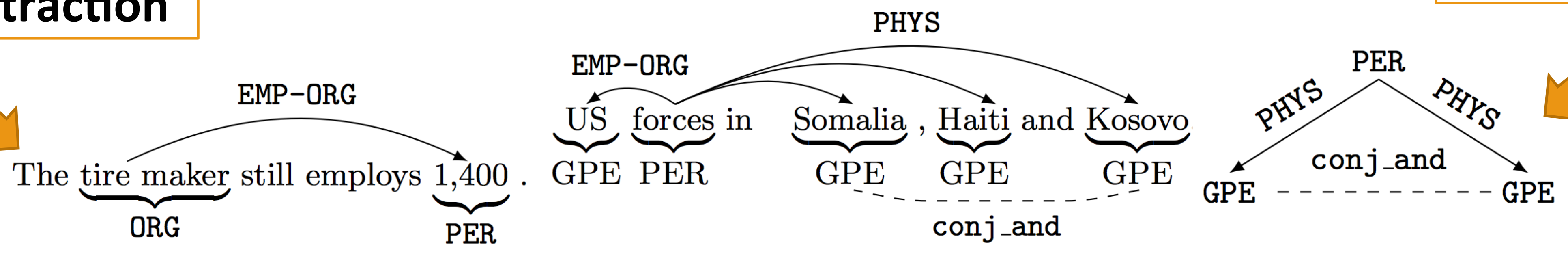
- Adopt segment-based decoder based on a semi-Markov chain (instead of token-based taggers)
- Incrementally detect mention & relation boundaries (detects mentions on the segment level)
- Global features used as soft constraints



Incremental Joint Entity and Extraction

Joint Extraction

Global Features



Model	Entity Mention (%)			Relation (%)			Entity Mention + Relation (%)		
Score	P	R	F_1	P	R	F_1	P	R	F_1
Pipeline	83.2	73.6	78.1	67.5	39.4	49.8	65.1	38.1	48.0
Joint w/ Local	84.5	76.0	80.0	68.4	40.1	50.6	65.3	38.3	48.3
Joint w/ Global	85.2	76.9	**80.8**	68.9	41.9	**52.1**	65.4	39.8	**49.5**
Annotator 1	91.8	89.9	90.9	71.9	69.0	70.4	69.5	66.7	68.1
Annotator 2	88.7	88.3	88.5	65.2	63.6	64.4	61.8	60.2	61.0
Inter-Agreement	85.8	87.3	86.5	55.4	54.7	55.0	52.3	51.6	51.9

Comparison of pipeline vs. joint extraction (global and local features)

LSTM for Entity Mention Detection

- A form of neural network known as a Long Short-Term Memory is applied to classify into entity mentions.
- Two passes are made in the inference.
  1. First pass is used to acquire information for disambiguation.
  2. Disambiguation information is used in the second pass.

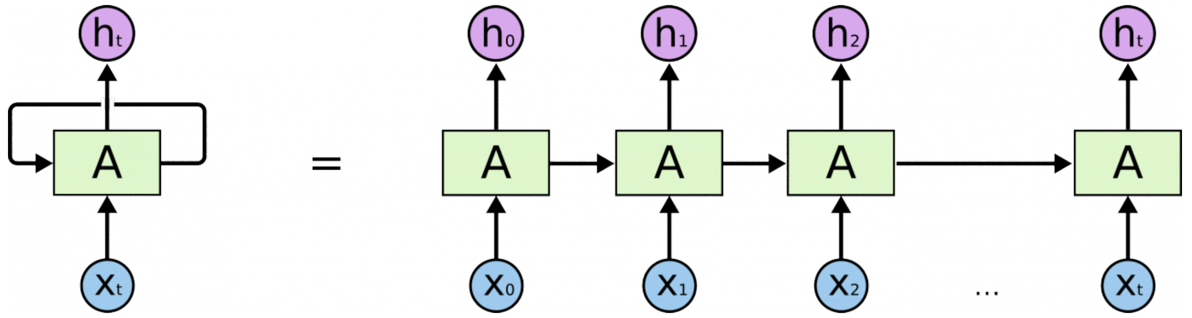
- Features are based on SARD-NET, a self organizing map for sequences used to generate representations for lexical items.
  - Without going into detail, takes a sequence and transforms it into a real-valued distributed representation.

Long Short-Term Memory Approach

- Long-Short Term Memory is a recurrent neural network architecture.
  - Well suited to learning from “experience” – that is well suited when there are long time-lags of unknown size between important events (entity mention appearance)

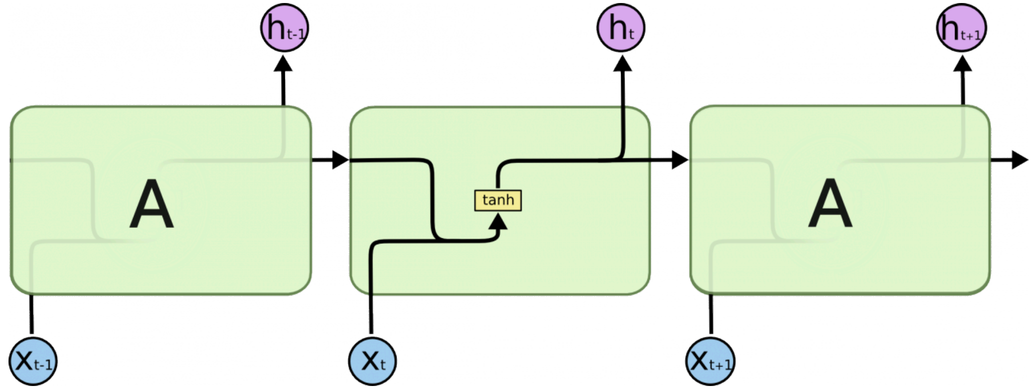
RNN

Unrolled almost like multiple NN, each passing a message to the next neural network. What about long-term dependency?



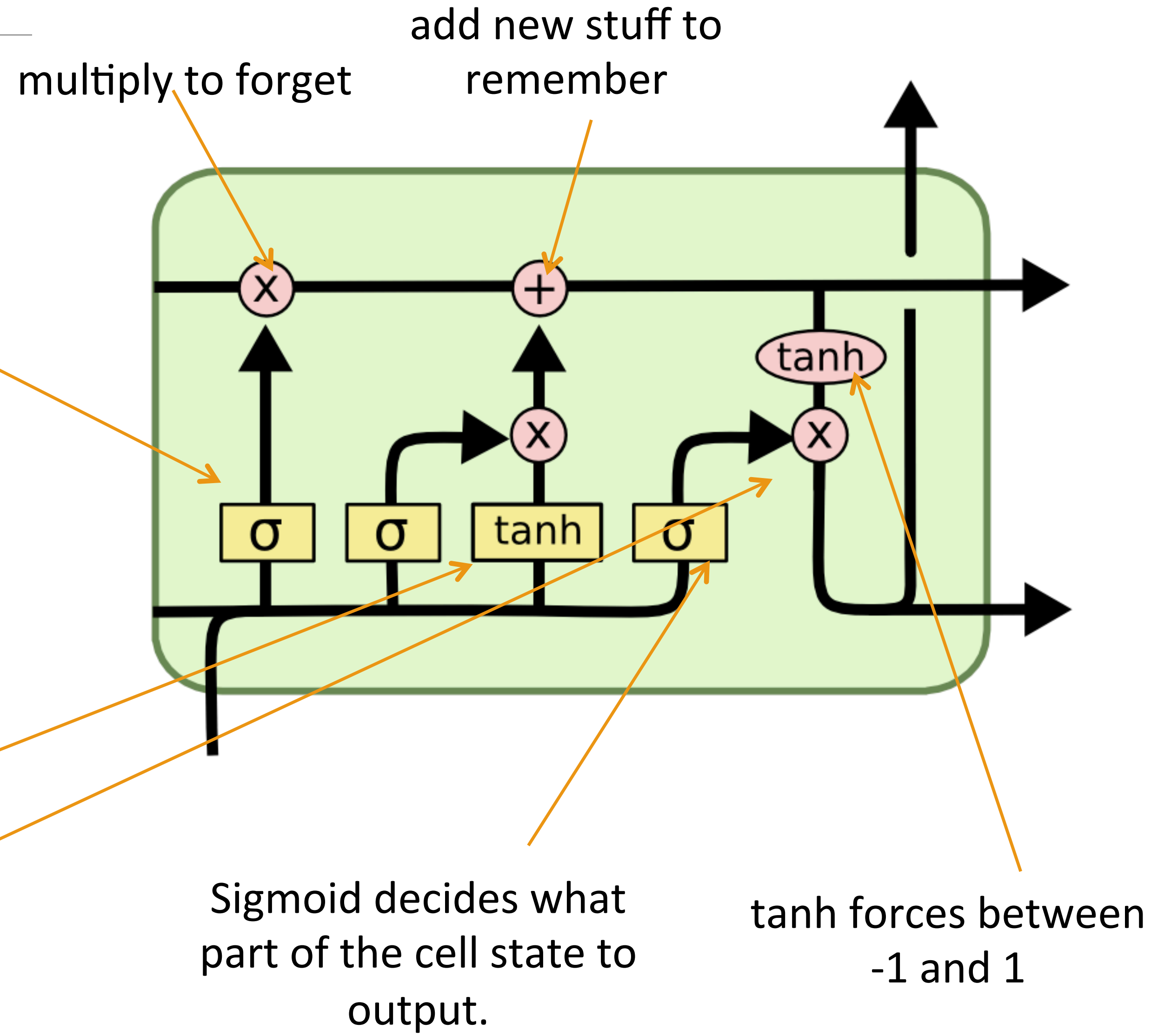
LSTM

At each state, decides what to forget, what new things to remember, and what to output to the next state.



LSTM Step-by-Step

1. What to forget?
   - Looks at the output from the previous layer and sigmoids to decide whether to forget?
2. What new stuff to remember in a cell?
   - Sigmoid decides what to update and tanh gets a set of candidates.
3. Combine cell-state decide what parts of the state to output.



LSTM Entity Mention Detection Results

- Barely above baseline on English and significantly above baseline in German.
- While not too impressive, it did open the floodgates into using LSTMs for entity recognition detection.
- Further works did improve significantly.

Net	Precision	Recall	Fscore	Range
Net1	61.42%	46.64%	52.98	49.16–54.30
Net2	62.42%	49.70%	55.30	53.75–56.92
Net3	62.80%	48.02%	54.41	52.24–55.74
Net4*	*75.27%*	*64.61%*	*69.53*	*68.55–70.60*
Net5*	75.03%	65.13%	69.73	68.05–70.58
Net6	67.92%	57.17%	62.08	59.26–64.14
Net7	68.04%	58.59%	62.95	61.25–64.86
Net8*	**76.37%**	**66.27%**	**70.96**	**69.46–72.88**
Basel.	78.33%	65.23%	71.18	n/a

Table 3: Results of named entity recognition on English development data for networks trained on the English training data. Results are averaged over 5 runs using different initial weights. * indicates use of the list of NEs. Italics indicate best result reported on first submission, whilst bold indicates best result achieved overall.

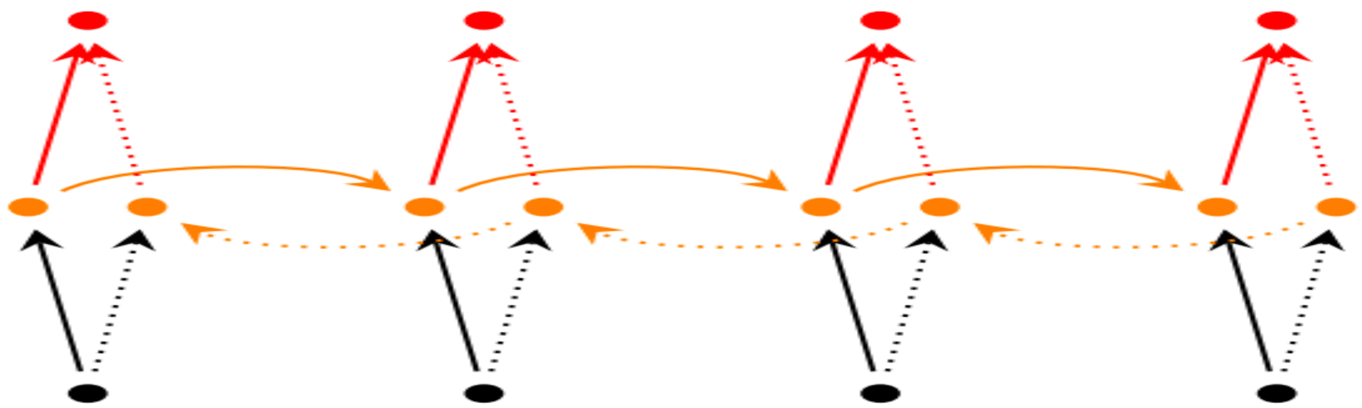
Bidirectional LSTM-CRF Model

- Recent works have proposed a variety of of LSTM models for sequential classification.

1. LSTM Networks which have shown to be powerful for sequential classification and applications such as entity mention detection
2. Bidirectional LSTMs which utilize both previous and future information. These have been shown to provide gains where “context” is needed.
3. LSTM-CRFs which are LSTMs with a conditional random field layer. These utilize sentence-level tag information thanks to the CRF layer.
4. Bidirectional LSTM-CRFs which combine the benefits of Bidirectional LSTMs and having sentence-level tag information via a CRF layer.

Bidirectional RNNs and LSTMS

- The main insight is that the output at time or location t depends not only on previous elements, but also future elements.
- This is essentially saying you may need to read a little further for context in disambiguating what the output should be – a reasonable assumption.



- Output is then computed based on the hidden states induced by the forward and backwards paths.

Features Used

- Spelling Features
  - whether start with a capital letter
  - whether has all capital letters
  - whether has all lower case letters
  - whether has non initial capital letters
  - whether mix with letters and digits
  - whether has punctuation
  - letter prefixes and suffixes (with window size of 2 to 5)
  - whether has apostrophe end (’s)
  - letters only, for example, I. B. M. to IBM
  - non-letters only, for example, A. T. &T. to ..&
  - word pattern feature, with capital letters,
  - etc
- Context Features
  - unigram features
  - bi-gram features
  - trigram features
- Word Embeddings
  - 130K vocabulary pre-trained embedding
  - 50-dimensional vector representation
  - Replaces one-hot with embedding

Performance in Sequential Tagging Tasks

		POS	CoNLL2000	CoNLL2003
Random	Conv-CRF (Collobert et al., 2011)	96.37	90.33	81.47
	LSTM	97.10	92.88	79.82
	BI-LSTM	97.30	93.64	81.11
	CRF	97.30	93.69	83.02
	LSTM-CRF	**97.45**	93.80	84.10
	BI-LSTM-CRF	97.43	**94.13**	**84.26**
Senna	Conv-CRF (Collobert et al., 2011)	97.29	94.32	88.67 (89.59)
	LSTM	97.29	92.99	83.74
	BI-LSTM	97.40	93.92	85.17
	CRF	97.45	93.83	86.13
	LSTM-CRF	97.54	94.27	88.36
	BI-LSTM-CRF	**97.55**	**94.46**	**88.83 (90.10)**

Performance on POS Tagging, Chunking and NER tasks

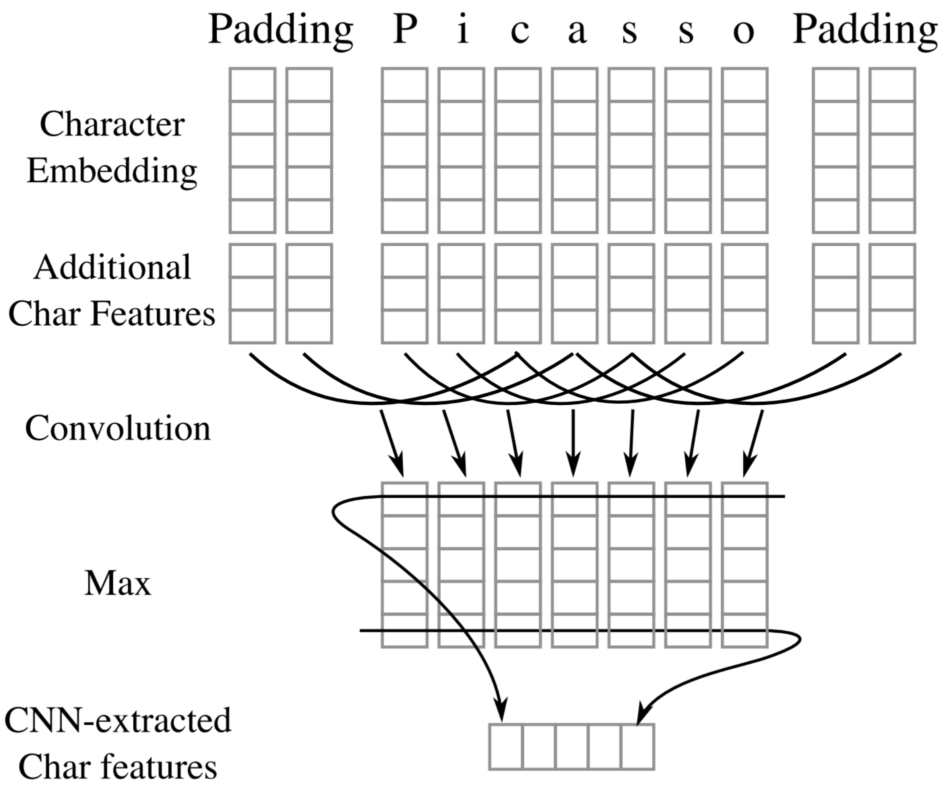
Bidirectional LSTM-CNNs

- Hybrid method that uses a hybrid LSTM and CNN architecture.
- Automatically detects word and character-level features.
- No need for costly feature engineering (less human work-needed).

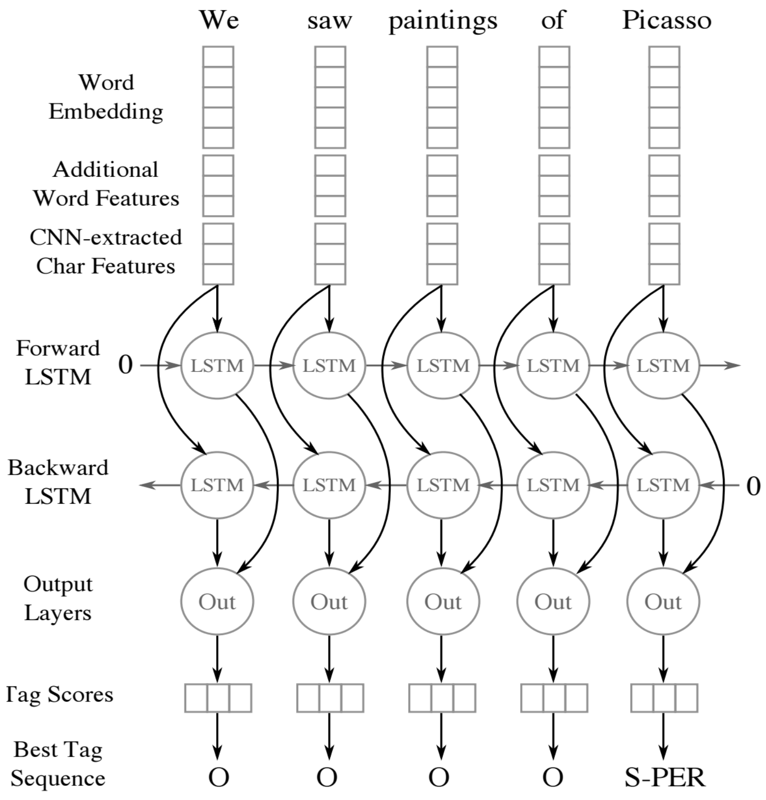
Model Pipeline

1. Utilize a convolutional neural network to induce character-level features
2. Lookup tables transform features such as words, characters, etc into continuous feature representation.
3. Concatenated continuous vectors are fed into a bi-directional Long Short-term Memory neural network model (LSTM)
4. Training done with mini-batch stochastic gradient descent

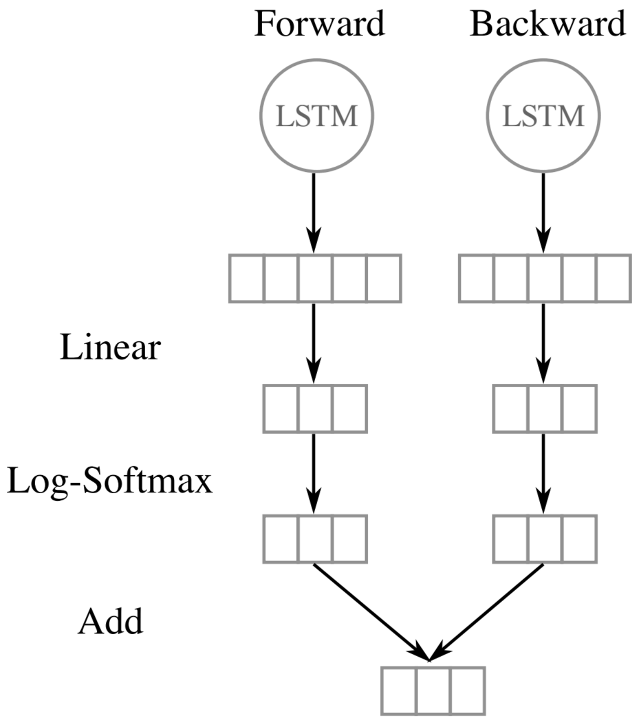
Bidirectional LSTM-CNNs



CNN extracts character features from each word.



B-LSTM tags entities (Classification)



B-LSTM Model

Bidirectional LSTM-CNNs: Results

Model	CoNLL-2003			OntoNotes 5.0		
	Prec.	Recall	F1	Prec.	Recall	F1
FFNN + emb + caps + lex	89.54	89.80	89.67 (± 0.24)	74.28	73.61	73.94 (± 0.43)
BLSTM	80.14	72.81	76.29 (± 0.29)	79.68	75.97	77.77 (± 0.37)
BLSTM-CNN	83.48	83.28	83.38 (± 0.20)	82.58	82.49	82.53 (± 0.40)
BLSTM-CNN + emb	90.75	91.08	90.91 (± 0.20)	85.99	86.36	86.17 (± 0.22)
BLSTM-CNN + emb + lex	91.39	**91.85**	**91.62** (± 0.33)	**86.04**	**86.53**	**86.28** (± 0.26)
Collobert et al. (2011b)	-	-	88.67	-	-	-
Collobert et al. (2011b) + lexicon	-	-	89.59	-	-	-
Huang et al. (2015)	-	-	90.10	-	-	-
Ratinov and Roth (2009)[18]	91.20	90.50	90.80	82.00	84.95	83.45
Lin and Wu (2009)	-	-	90.90	-	-	-
Finkel and Manning (2009)[19]	-	-	-	84.04	80.86	82.42
Suzuki et al. (2011)	-	-	91.02	-	-	-
Passos et al. (2014)[20]	-	-	90.90	-	-	82.24
Durrett and Klein (2014)	-	-	-	85.22	82.89	84.04
Luo et al. (2015)[21]	**91.50**	91.40	91.20	-	-	-

Results compared to literature and with various feature sets.

Category	SENNA	DBpedia
Location	36,697	709,772
Miscellaneous	4,722	328,575
Organization	6,440	231,868
Person	123,283	1,074,363
Total	171,142	2,344,578

Number of entries for each category.

Dataset	Train	Dev	Test
CoNLL-2003	204,567 (23,499)	51,578 (5,942)	46,666 (5,648)
OntoNotes 5.0 / CoNLL-2012	1,088,503 (81,828)	147,724 (11,066)	152,728 (11,257)

Dataset size (tokens).

Three Families of Methods

A. Supervised/Semi-supervised Entity Mention Detection

B. Unsupervised Entity Mention Detection

C. Weakly and Distantly Supervised Mention Detection

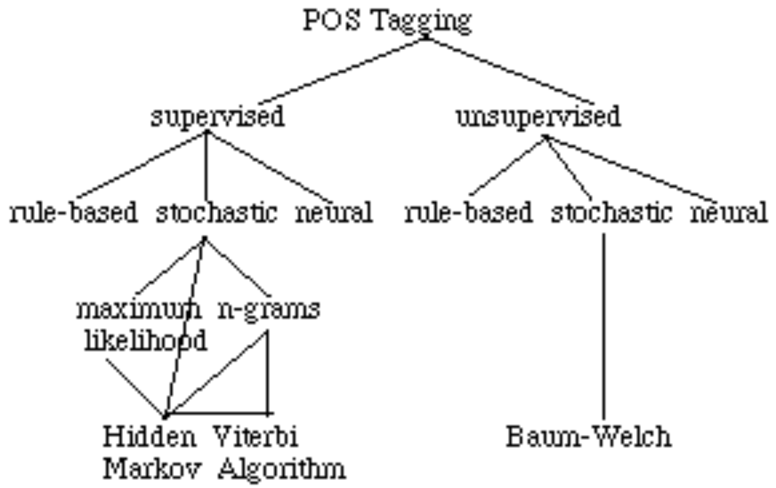
Unsupervised Entity Mention Detection

Assumptions

1. Part-of-speech tags are relatively inexpensive to obtain training data for
2. Part-of-speech tags generalize much better to new domains than parsing does
3. Training data is not available

As such, we consider the use of POS tags as an input to these methods.

There are a variety of methods in use for POS tagging



NP-Chunking with Chunking Grammars

- Observations
  - Noun phrase chunks are smaller than full noun phrases (NP Chunks should not contain other NP Chunks)

Grammar: <DT>?<JJ>*<NN> Noun Chunk Pattern

We saw **the big yellow dog**.

More Chunking Patterns

- After observing the data, one can define many relevant chunking patterns for entity mentions

<DT>?<JJ>*<NN>

<PP>?<JJ>*<NN>

<JJ>*<NN>+

<JJ>*<NNP>+

Improving Chunking

- Sometimes the Chunking Patterns may be less aggressive in identifying entity mentions
- One approach is to specify items (stopwords or POS tags) that can be used to split large noun chunks into smaller elements
- It may be easier to specify what ***shouldn't belong in a chunk***

Leveraging Corpus Level Information

Corpus-level entity mention detection has the benefit of leveraging corpus-level statistics to aid in determining mention boundaries

Insights

1. Redundancy: Core entity mentions likely appear multiple times in the corpus
2. Longer candidate entity mentions should not be favored over shorter, more common, sub-mentions without evidence

A Noun Collocation Mining Approach

Good entity mentions are noun phrases that appear more frequently in a corpus than expected.

- Humans can define high-precision chunking grammars
- Corpus level statistics through ***redundancy*** can aid entity mention detection

Detecting high-quality entity mention candidates requires ***both:***

- ***accurate POS-based pattern matching***
- ***Identification of significant patterns***

A Noun Collocation Mining Approach

- A framework for identifying entity mentions within domain-specific corpora

raw corpus → sentence segmentation → tokenization → stopword removal → frequent segment mining → entity mention identification

We identify these entity mentions using a **Significant Mention Chunking Algorithm**

Corpus Level Statistics

corpus

dropped nearly inches of snow in Western Oklahoma and at Dallas Fort Worth International Airport sleet and ice caused

mine/count frequent segments

dallas : 30
fort: 60
international: 80
airport: 40
worth : 80
dallas fort: 26
International airport: 35
…

- $\mathcal{V}$(segment) denotes the count of a segment
- Given two segments, we can obtain a ***significance*** of merging two such segments

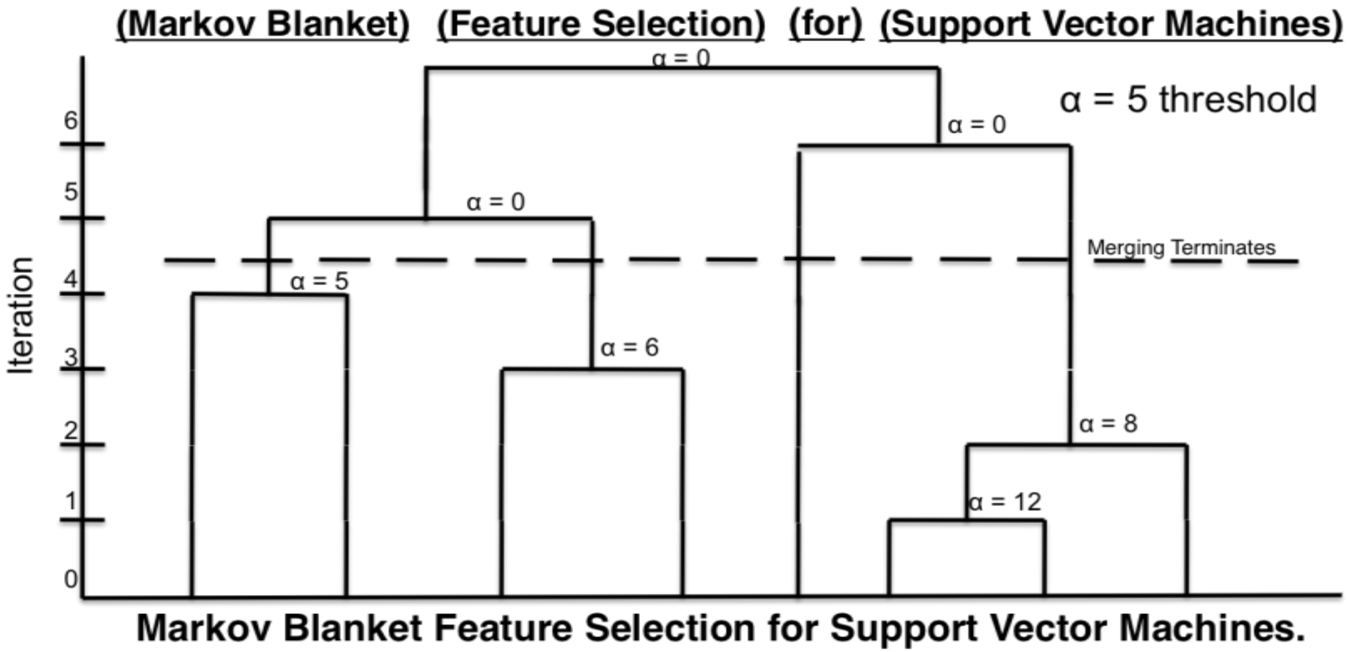
$$\rho_X(S_1, S_2) = \frac{v(S_1 \oplus S_2) - N\frac{v(S_1)}{N}\frac{v(S_2)}{N}}{\sqrt{v(S_1 \oplus S_2)}} \cdot I_X(S_1 \oplus S_2)$$

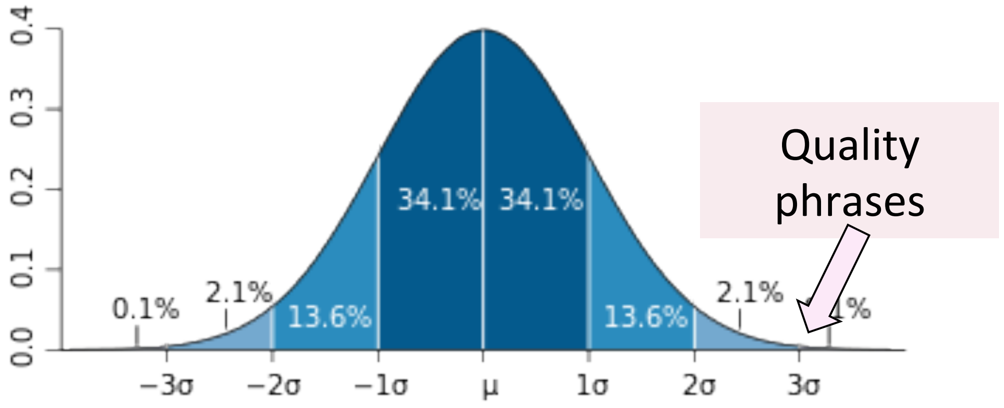
Differences from KeyPhrase Extraction

- Other methods may use significance score to rank methods that are significant highly
  - This may allow for low quality entity phrases that appear significant to rank highly

- This Noun Collocation mining differs from key phrase extraction in one major way
  - Noun Collocation Mining goes to the exact location where a candidate phrase occurs and ***segments the sentence*** which simultaneously filters out bad entity candidates

Significant Mention Chunking Algorithm





Based on significance score [Church et al.'91]:

$$\alpha(P_1, P_2) \approx (f(P_1 \bullet P_2) - \mu_0(P_1,P_2))/\sqrt{f(P_1 \bullet P_2)}$$

[Markov blanket] [feature selection] for [support vector machines]
[knowledge discovery] using [least squares] [support vector machine] [classifiers]
…[support vector] for [machine learning]…

Phrase	**Raw freq.**	**True freq.**
[support vector machine]	90	80
[vector machine]	95	0
[support vector]	100	20

Significant Mention Chunking Algorithm

1. With all stopwords removed from consideration, we search for chunks that meet the following grammar
2. Among grammar matches, only merge "significant" noun phrases

Entity Grammar

<JJ>*<NN>*

Not significant

Over the weekend the system dropped nearly inches of snow in western **[Oklahoma]** and at **[Dallas Fort Worth International Airport]** sleet and ice caused hundreds of [**flight cancellations]** … It is forecast to reach by **[Tuesday afternoon] [Washington]** and **[New York]** by **[Wednesday afternoon]**

Significant

Application: Significant Keyphrase Extraction

1. First take input text corpus and apply POS-Constrained Collocation Mining

Over the weekend the system dropped nearly inches of snow in Western Oklahoma and at Dallas Fort Worth International Airport sleet and ice caused hundreds of flight cancellations …	→	Over the weekend the system dropped nearly inches of snow in Western **Oklahoma** and at **[Dallas Fort Worth International Airport]** sleet and ice caused hundreds of **[flight cancellations]** …

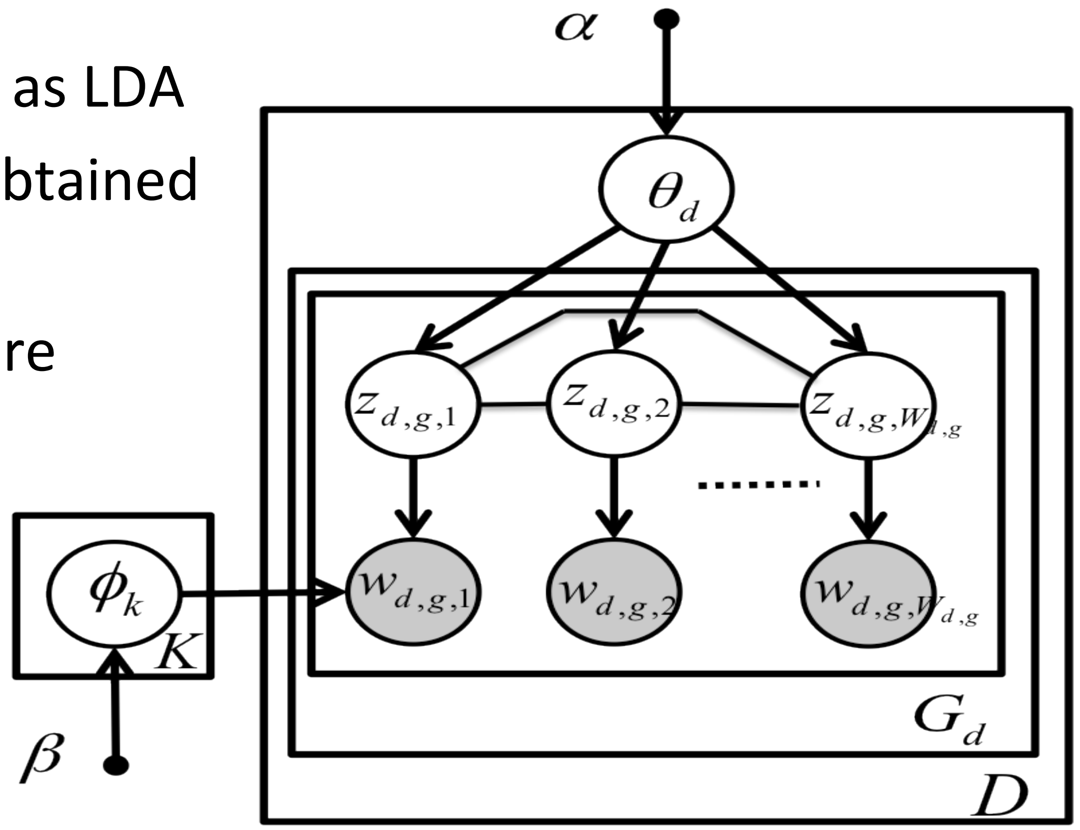
The POS constrain the collocation mining. This finds ***corpus-relevant*** key phrases.

These significant multi-word phrases can be used for a variety of applications.

Application: Topical Phrase Mining

2. **One application is applying phrase-based topic modeling.**

- The generative model for PhraseLDA is the same as LDA
- Difference: the model incorporates constraints obtained from the **"bag-of-phrases"** input
  - Chain-graph shows that all words in a phrase are constrained to take on the same topic values



Over the weekend the system dropped nearly inches of snow in Western **Oklahoma** and at **[Dallas Fort Worth International Airport]** sleet and ice caused hundreds **of** **[flight cancellations]** …

Topic model inference with phrase constraints

ToPMine: Topics on Associate Press News (1989)

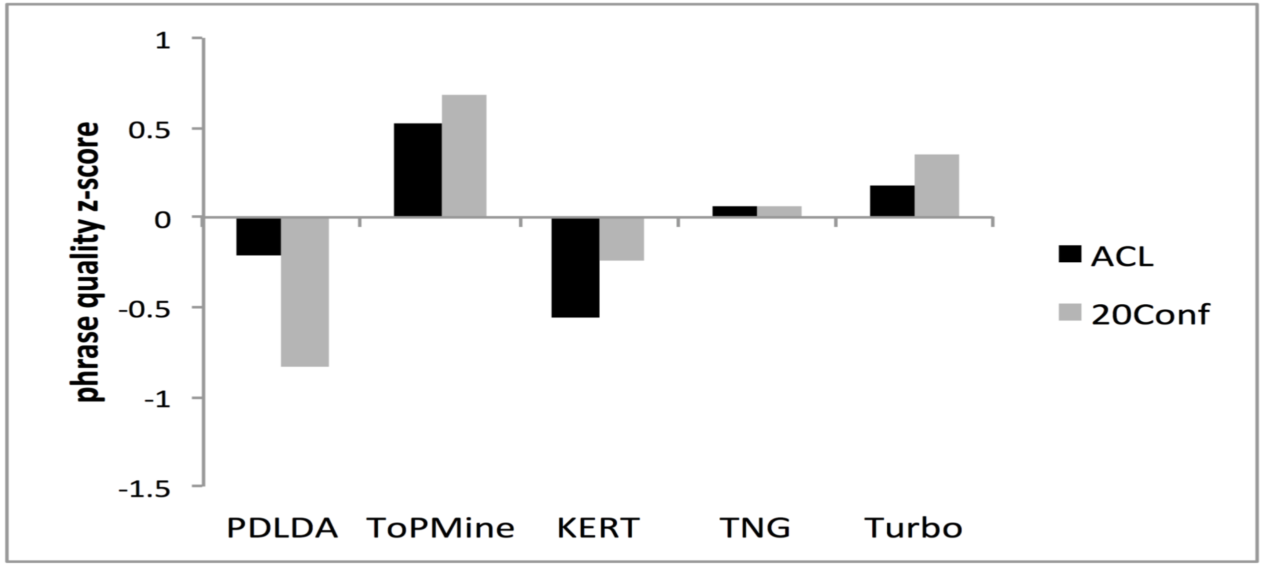
	Topic 1	*Topic 2*	*Topic 3*	*Topic 4*	*Topic 5*
unigrams	plant	church	palestinian	bush	drug
	nuclear	catholic	israeli	house	aid
	environmental	religious	israel	senate	health
	energy	bishop	arab	year	hospital
	year	pope	plo	bill	medical
	waste	roman	army	president	patients
	department	jewish	reported	congress	research
	power	rev	west	tax	test
	state	john	bank	budget	study
	chemical	christian	state	committee	disease
n-grams	energy department	roman catholic	gaza strip	president bush	health care
	environmental protection agency	pope john paul	west bank	white house	medical center
	nuclear weapons	john paul	palestine liberation prganization	bush administration	united states
	acid rain	catholic church	united states	house and senate	aids virus
	nuclear power plant	anti semitism	arab reports	members of congress	drug abuse
	hazardous waste	baptist church	prime minister	defense secretary	food and drug administration
	savannah river	united states	yitzhak shamir	capital gains tax	aids patient
	rocky flats	lutheran church	israel radio	pay raise	centers for disease control
	nuclear power	episcopal church	occupied territories	house members	heart disease
	natural gas	church members	occupied west bank	committee chairman	drug testing

ToPMine Runtime and Phrase Quality

Running time of different algorithms

Method	*sampled dblp titles (k=5)*	*dblp titles (k=30)*	*sampled dblp abstracts*	*dblp abstracts*
PDLDA	3.72(hrs)	~20.44(days)	1.12(days)	~95.9(days)
Turbo Topics	6.68(hrs)	>30(days)*	>10(days)*	>50(days)*
TNG	146(s)	5.57 (hrs)	853(s)	NA†
LDA	**65(s)**	3.04 (hrs)	353(s)	13.84(hours)
KERT	68(s)	3.08(hrs)	1215(s)	NA†
ToP-Mine	67(s)	**2.45(hrs)**	**340(s)**	**10.88(hrs)**

Phrase quality measured by z-score



POS-Constraining ToPMine

- ToPMine divides the topical phrase extraction process into two steps
  1. Segmenting the raw corpus into single and multi-word phrases
  2. Performing phrase-constrained topic modeling
- Since POS-Constrained noun collocation mining also segments the corpus, we can integrate the noun-collocation mining as a first step into ToPMine

This leads to POS-Constrained ToPMine:
Each phrase is a higher-quality phrase because of the part-of-speech constraints!

Improving ToPMine with POS Constraints

- Observing ToPMine on Yelp Reviews, we can see some bad topical phrases can be filtered by enforcing our POS constraints

Topic 1

ToPMine	POS-Constrained TopMine
spring rolls	spring rolls
food was good	fried rice
fried rice	egg rolls
egg rolls	dim sum
pretty good	Thai food
dim sum	Chinese food
Thai food	pad thai

Topic 2

ToPMine	POS-Constrained TopMine
great selection	grocery store
farmer's market	farmer's market
great prices	parking lot
wal mart	shopping center
prices are reasonable	county market
great place	fresh produce
love this place	wal mart supercenter

Three families of methods

A. Supervised/Semi-supervised Entity Mention Detection

B. Unsupervised Entity Mention Detection

C. Weakly and Distantly Supervised Mention Detection

85

Weakly Supervised Methods

Assumptions

1. Unsupervised methods cannot possibly take into consideration the innumerable features, signals, and cues for entity mentions
2. Full supervision can be too expensive (time-wise) to manage

- Use methods that require small numbers of labeled instances (small number of seed entities)

- Rely on entity information from knowledge bases as seed entities

Semi-Supervised Chunker with Structure learning

Insight: Use unlabeled to identify underlying structure of what makes a "good classifier"

1. Learns the concept of a "good classifier" by learning from thousands of automatically generated auxiliary classification on unlabeled data
2. Predictive structure shared by multiple classifiers can be discovered and used to improve performance on target problem

English, all (204K) training examples					
ASO-semi co/self oracle	dev.	**93.15** 90.64	*+2.25* +0.04	*+3.00* +0.20	*+2.62* +0.11
ASO-semi co/self oracle	test	**89.31** 85.40	*+3.20* −0.04	*+4.51* −0.05	*+3.86* −0.05

Exploiting Dictionaries in Mention Detection

Challenges

- Most mention detections sequentially classify words in whether they participate in a candidate mention
- Similarity measures are applied to ***full entity mention candidates***

- **Proposed Method**
  - Semi-Markov extraction, sequentially classifies segments instead of tokens
  - Allows for integration of entity mention detection methods and similarity methods with external data

Exploiting Dictionaries in Mention Detection

Observations

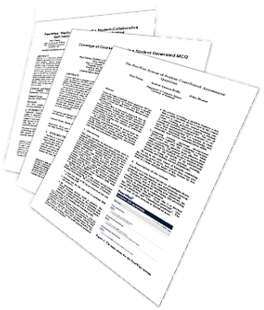
- Semi-Markov Model & HMM implementations with & without dictionary features on NER tasks
- Distance-based incorporation of dictionary values outperforms binary features

		Without dictionary			With dictionary					
					Binary features			Distance features		
		Recall	Prec.	F1	Recall	Prec.	F1	Recall	Prec.	F1
Address-state	lookup				32.2	100.0	48.7			
	HMM-VP$_{(1)}$	5.2	56.8	9.5	19.3	82.6	**31.3**	41.5	87.3	56.3
	HMM-VP$_{(4)}$	8.9	90.7	**16.2**	13.0	97.3	23.0	25.7	100	40.9
	SMM-VP	8.2	62.2	14.6	16.4	82.0	27.3	39.7	97.7	**56.4**
Address-city	lookup				14.8	68.8	24.3			
	HMM-VP$_{(1)}$	60.1	79.3	68.3	68.0	84.2	75.2	70.8	84	76.8
	HMM-VP$_{(4)}$	59.1	87.3	70.5	64.1	91.2	75.2	68.1	90.6	77.7
	SMM-VP	62.8	87.5	**73.1**	70.7	90.0	**79.2**	72.2	89.4	**79.9**
Email-person	lookup				38.7	82.6	57.3			
	HMM-VP$_{(1)}$	60.4	74.9	66.8	73.4	83.7	78.2	79.1	84.6	81.8
	HMM-VP$_{(4)}$	60.9	80.2	69.3	71.1	87.6	78.5	77.1	89.2	82.7
	SMM-VP	64.1	80.3	**71.3**	77.7	88.1	**82.6**	78.9	88.5	**83.4**
Job-company	lookup				14.1	54.8	22.3			
	HMM-VP$_{(1)}$	1.3	34.7	2.5	2.0	28.1	3.8	8.9	79.8	16.1
	HMM-VP$_{(4)}$	3.6	59.8	6.8	11.5	80.6	20.2	18.6	93.4	**31.1**
	SMM-VP	5.2	55.3	**9.6**	13.8	85.4	**23.7**	17.8	95.9	30.0
Job-title	lookup				29.4	29.5	29.4			
	HMM-VP$_{(1)}$	18.4	43.7	25.9	23.9	43.2	30.8	30.9	44.2	36.4
	HMM-VP$_{(4)}$	17.3	51.5	25.9	27.9	48.4	35.4	30.9	45.7	36.8
	SMM-VP	20.9	52.0	**29.8**	34.9	48.8	**40.7**	36.2	47.9	**41.2**

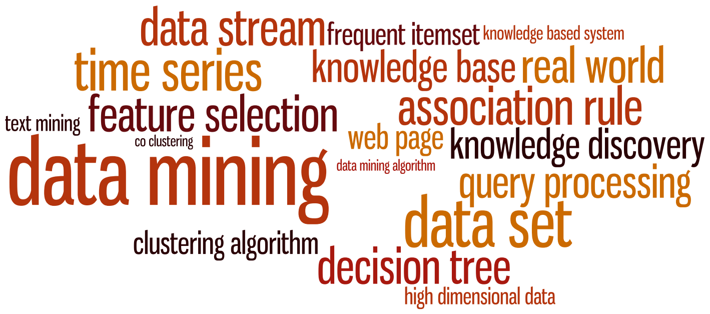
Table 3: Performance of NER methods on five IE tasks under three conditions: with no external dictionary; with an external dictionary and binary features; with an external dictionary and distance features.

SegPhrase: Weakly Supervised Mention Detection

Raw Corpus



Quality Phrases



Segmented Corpus

Document 1
Citation recommendation is an interesting but challenging research problem in data mining area.

Document 2
In this study, we investigate the problem in the context of heterogeneous information networks using data mining technique.

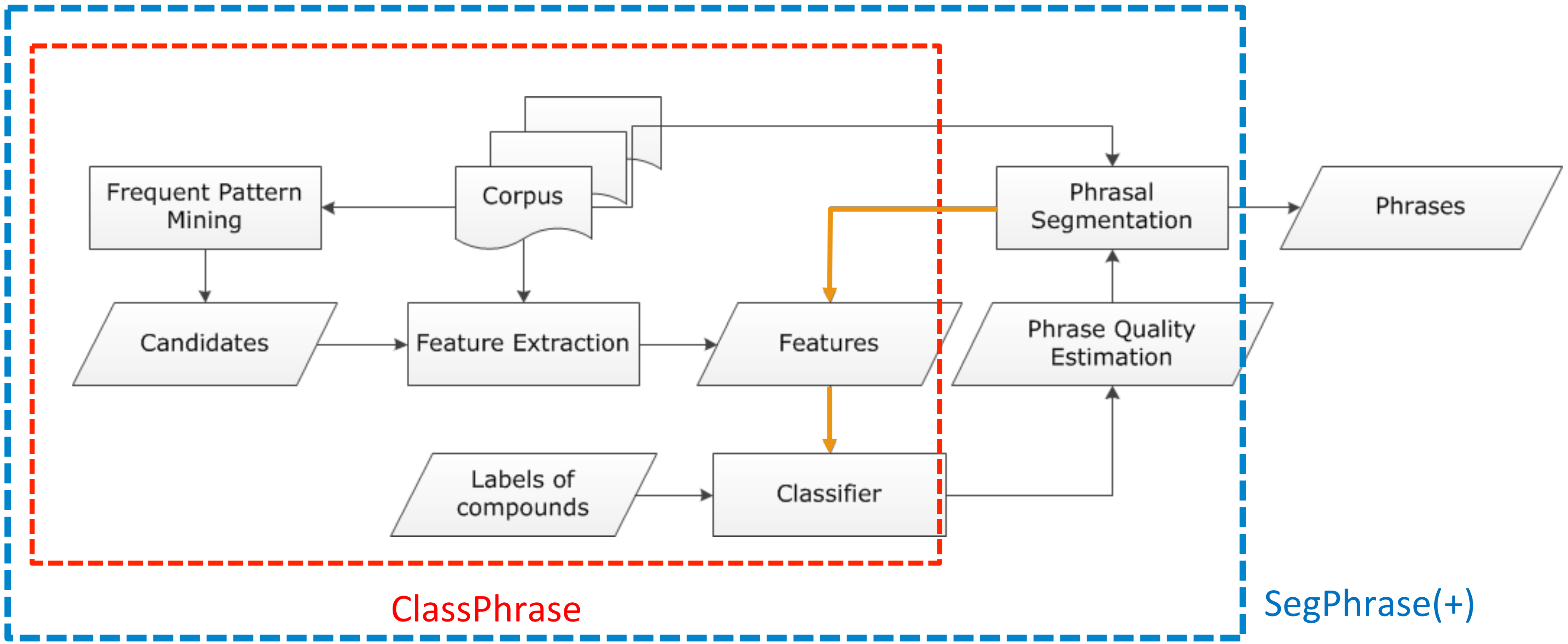
Document 3
Principal Component Analysis is a linear dimensionality reduction technique commonly used in machine learning applications.

Input Raw Corpus → **Quality Phrases** ↔ **Segmented Corpus**

Phrase Mining **Phrasal Segmentation**

SegPhrase: The Overall Framework

- ClassPhrase: Frequent pattern mining, feature extraction, classification
- SegPhrase: Phrasal segmentation and phrase quality estimation
- SegPhrase+: One more round to enhance mined phrase quality



What Kind of Phrases Are of “High Quality”?

- Judging the quality of phrases
  - **Popularity**
    - “information retrieval” vs. “cross-language information retrieval”
  - **Concordance**
    - “powerful tea” vs. “strong tea”
    - “active learning” vs. “learning classification”
  - **Informativeness**
    - “this paper” (frequent but not discriminative, not informative)
  - **Completeness**
    - “vector machine” vs. “support vector machine”

ClassPhrase I: Pattern Mining for Candidate Set

- Build a candidate phrases set by frequent pattern mining
  - Mining frequent *k*-grams
    - *k* is typically small, e.g. 6 in our experiments
- **Popularity** measured by *raw* frequent words and phrases mined from the corpus

ClassPhrase II: Feature Extraction: **Concordance**

- Partition a phrase into two parts to check whether the co-occurrence is significantly higher than pure random
  - $\underbrace{\text{support vector}}_{u_l}\ \underbrace{\text{machine}}_{u_r}$ $\qquad$ $\underbrace{\text{this paper}}_{u_l}\ \underbrace{\text{demonstrates}}_{u_r}$

$$\langle u_l, u_r \rangle = \arg \min_{u_l \oplus u_r = v} \log \frac{p(v)}{p(u_l)p(u_r)}$$

  - Pointwise mutual information:

$$PMI(u_l, u_r) = \log \frac{p(v)}{p(u_l)p(u_r)}$$

  - Pointwise KL divergence:

$$PKL(v \| \langle u_l, u_r \rangle) = p(v) \log \frac{p(v)}{p(u_l)p(u_r)}$$

- The additional *p*(*v*) is multiplied with pointwise mutual information, leading to less bias towards rare-occurred phrases

ClassPhrase II: Feature Extraction: **Informativeness**

- Deriving Informativeness
  - Quality phrases typically start and end with a non-stopword
    - “machine learning is” vs. “machine learning”
  - Use average IDF over words in the phrase to measure the semantics
  - Usually, the probabilities of a quality phrase in quotes, brackets, or connected by dash should be higher (punctuations information)
    - “state-of-the-art”
- We can also incorporate features using some NLP techniques, such as POS tagging, chunking, and semantic parsing

ClassPhrase III: Classifier

- **Limited Training**
  - Labels: Whether a phrase is a quality one or not
    - “support vector machine”: 1
    - “the experiment shows”: 0
  - For ~1GB corpus, only 300 labels
- Random Forest as our classifier
  - Predicted phrase quality scores lie in [0, 1]
  - Bootstrap many different datasets from limited labels

SegPhrase: Why Do We Need Phrasal Segmentation in Corpus?

- Phrasal segmentation can tell which phrase is more appropriate
  - Ex: A standard [feature vector] [machine learning] setup is used to describe...

Not counted towards the rectified frequency

- Rectified phrase frequency (expected influence)
  - Example:

sequence	frequency	phrase?	rectified
support vector machine	100	yes	80
support vector	160	yes	50
vector machine	150	no	6
support	500	N/A	150
vector	1000	N/A	200
machine	1000	N/A	150

SegPhrase: Segmentation of Phrases

- Partition a sequence of word by maximizing the likelihood
  - Considering
    - Phrase quality score
      - ClassPhrase assigns a **quality score** for each phrase
    - Probability in corpus
    - Length penalty
      - **length penalty** α: when $\alpha > 1$, it favors shorter phrases
- Filter out phrases with low rectified frequency
  - Bad phrases are expected to rarely occur in the segmentation results

SegPhrase+: Enhancing Phrasal Segmentation

- SegPhrase+: One more round for enhanced phrasal segmentation
- **Feedback**
  - Using rectified frequency, re-compute those features previously computing based on raw frequency
- Process
  - Classification → Phrasal segmentation // SegPhrase

    → Classification → Phrasal segmentation // SegPhrase+
- **Effects** on computing quality scores
  - np hard in the strong sense ↗
  - ~~np hard in the strong~~ ↘
  - data base management system ↗

Performance Study: Methods to Be Compared

- Other phase mining methods: Methods to be compared
  - NLP chunking based methods
    - Chunks as candidates
    - Sorted by **TF-IDF** and **C-value** (K. Frantzi et al., 2000)
  - Unsupervised raw frequency based methods
    - **ConExtr** (A. Parameswaran et al., VLDB 2010)
    - **ToPMine** (A. El-Kishky et al., VLDB 2015)
  - Supervised method
    - **KEA**, designed for single document keyphrases (O. Medelyan & I. H. Witten, 2006)

Performance Study: Experimental Setting

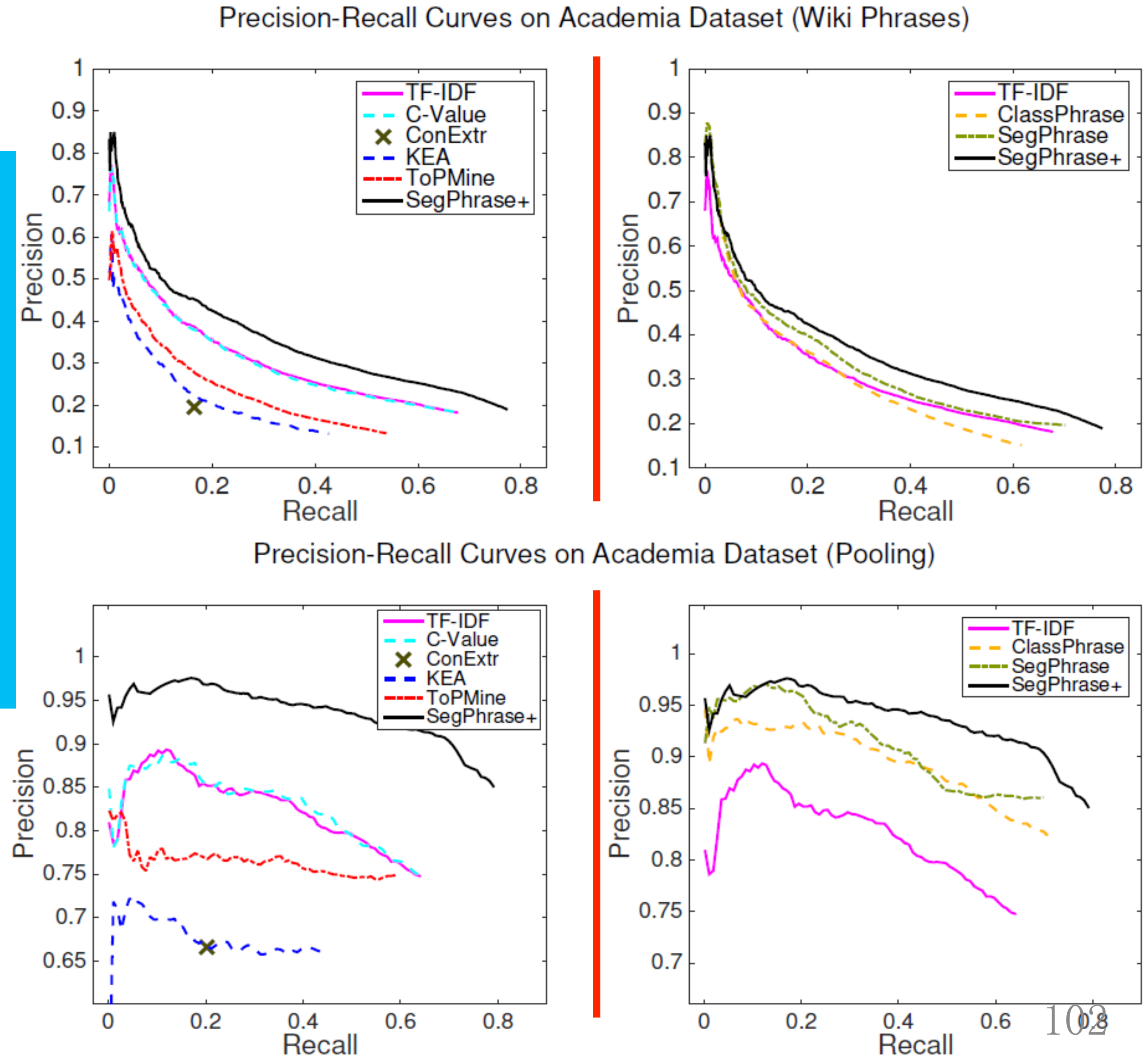
- Datasets

Dataset	#docs	#words	#labels
DBLP	2.77M	91.6M	300
Yelp	4.75M	145.1M	300

- Popular Wiki Phrases
  - Based on internal links
  - ~7K high quality phrases
- Pooling
  - Sampled 500 * 7 **Wiki-uncovered** phrases
  - Evaluated by 3 reviewers independently

Performance: Precision Recall Curves on DBLP

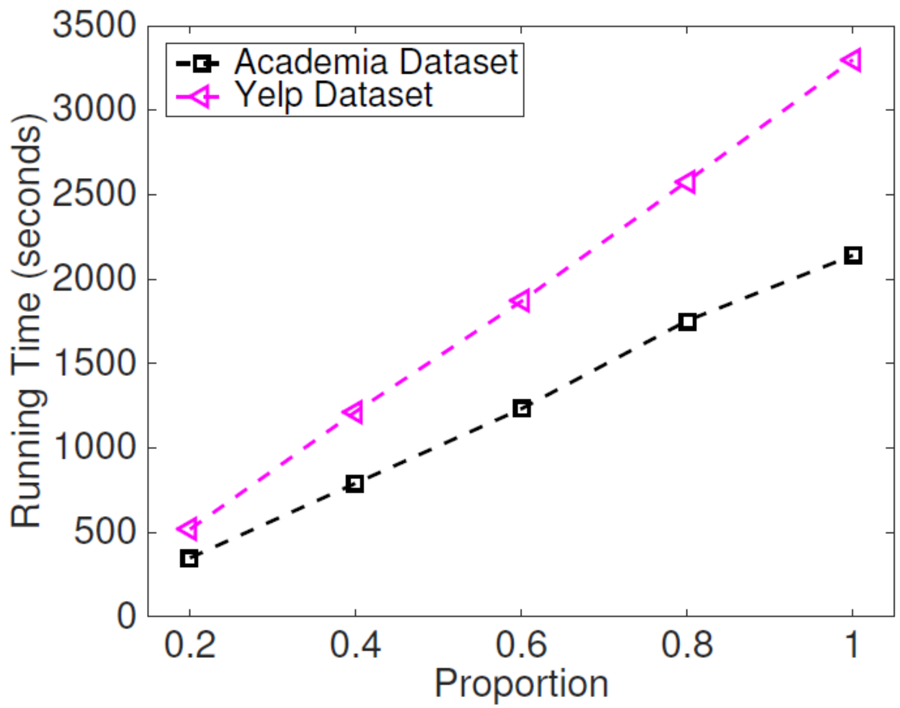
Compare with other baselines
TF-IDF
C-Value
ConExtr
KEA
ToPMine
SegPhrase+



Compare with our 3 variations
TF-IDF
ClassPhrase
SegPhrase
SegPhrase+

Performance Study: Processing Efficiency

- SegPhrase+ is linear to the size of corpus!



dataset	file size	#words	time
Academia	613MB	91.6M	0.595h
Yelp	750MB	145.1M	0.917h
Wikipedia	20.23GB	3.26G	28.08h

Extension to Multiple Languages

- Both ToPMine and SegPhrase+ are extensible to mining quality phrases in multiple languages
- SegPhrase+ on Chinese (From Chinese Wikipedia)
- ToPMine on Arabic (From Quran (Fus7a Arabic)(no preprocessing)
  - Experimental results of Arabic phrases:

→ Those who disbelieve كفروا

→ In the name of بسم الله الرحمن الرحيم God the Gracious and Mercfiul

Rank	Phrase	In English
…	…	…
62	首席_执行官	CEO
63	中间_偏右	Middle-right
…	…	…
84	百度_百科	Baidu Pedia
85	热带_气旋	Tropical cyclone
86	中国科学院_院士	Fellow of Chinese Academy of Sciences
…	…	…
1001	十大_中文_金曲	Top-10 Chinese Songs
1002	全球_资讯网	Global Info Website
1003	天一阁_藏_明代_科举_录_选刊	A Chinese book name
…	…	…
9934	国家_戏剧_院	National Theater
9935	谢谢_你	Thank you
…	…	…

Find “Interesting” Collections of Hotels

- Reported by TripAdvisor

1	club_quarters	0.999620288
2	hampton_inn	0.999542304
3	rush_hour	0.999526829
4	frosted_glass	0.999506234
5	ritz_carlton	0.999476254
6	usa_today	0.999473892
7	jersey_boys	0.999458328
8	holiday_inn_express	0.999450456
9	art_deco	0.9994495
10	gordon_ramsay	0.999448261
11	battery_park	0.999418922
12	grand_central_station	0.999410719
13	naked_cowboy	0.999401511
14	yankee_stadium	0.999390047
15	penn_station	0.999386755
16	columbus_circle	0.999381838
17	charlie_chaplin	0.999381761
18	scrambled_eggs	0.999379073
19	jet_lag	0.999370422
20	affinia_dumont	0.999364144
21	harry_potter	0.999357816
22	les_halles	0.999352377
23	air_conditioning	0.999346666
24	mamma_mia	0.999345891
25	hudson_river	0.999345247
26	pinot_noir	0.999344796
27	woody_allen	0.999337025
28	fairy_tale	0.999306646
29	grand_central	0.999304571
30	radio_city_music_hall	0.999301883

Some interesting collections

The “Catch a Show” collection has phrases like this:

1	at_radio_city_music_hall
2	b'way_shows
3	beacon_theater
4	beacon_theatre
5	broadway_dance_center
6	broadway_play
7	broadway_plays
8	broadway_shows
9	broadway_shows_and_great_restaurants
10	broadway_shows_and_restaurants
11	comedy_shows
12	david_letterman_show
13	easy_walk_to_broadway_shows
14	evening_entertainment
15	great_shows
16	radio_city_hall
17	radio_city_music
18	radio_city_music_hall
19	radio_city_music_hall_and
20	theater_shows
21	theatre_shows
22	walking_distance_to_broadway_shows
23	walking_distance_to_broadway_theaters
24	walking_distance_to_shows
25	walking_distance_to_theatre

My personal favorite when I'm in New York, the “Near The High Line” collection has:

1	chelsea_market_and_high_line
2	chelsea_market_and_the_highline
3	high_line
4	high_line_park
5	highline_park
6	highline_walk
7	highline_walkway
8	the_high_line_park

http://engineering.tripadvisor.com/using-nlp-to-find-interesting-collections-of-hotels/

Experimental Results: Interesting Phrases Generated (From the Titles and Abstracts of SIGMOD)

Query	SIGMOD	
Method	SegPhrase+	Chunking (TF-IDF & C-Value)
1	data base	data base
2	database system	database system
3	relational database	query processing
4	query optimization	query optimization
5	query processing	relational database
…	…	…
51	sql server	database technology
52	**relational data**	database server
53	data structure	**large volume**
54	**join query**	performance study
55	web service	web service
…	…	…
201	**high dimensional data**	**efficient implementation**
202	location based service	sensor network
203	xml schema	**large collection**
204	**two phase locking**	**important issue**
205	deep web	frequent itemset
…	…	…

Only in SegPhrase+

Only in Chunking

Experimental Results: Interesting Phrases Generated (From the Titles and Abstracts of SIGKDD)

Query	SIGKDD	
Method	SegPhrase+	Chunking (TF-IDF & C-Value)
1	data mining	data mining
2	data set	association rule
3	association rule	knowledge discovery
4	knowledge discovery	frequent itemset
5	**time series**	decision tree
…	…	…
51	association rule mining	search space
52	rule set	domain knowledge
53	concept drift	**important problem**
54	knowledge acquisition	concurrency control
55	**gene expression data**	conceptual graph
…	…	…
201	web content	optimal solution
202	**frequent subgraph**	semantic relationship
203	intrusion detection	**effective way**
204	**categorical attribute**	space complexity
205	user preference	**small set**
…	…	…

Only in SegPhrase+

Only in Chunking

Experimental Results: Similarity Search

- Find high-quality similar phrases based on user's phrase query
  - In response to a user's phrase query, SegPhrase+ generates high quality, semantically similar phrases
  - In DBLP, query on "data mining" and "OLAP"
  - In Yelp, query on "blu-ray", "noodle", and "valet parking"

Query	data mining		olap	
Method	SegPhrase+	Chunking	SegPhrase+	Chunking
1	knowledge discovery	driven methodologies	data warehouse	warehouses
2	text mining	text mining	online analytical processing	clustcube
3	web mining	financial investment	data cube	rolap
4	machine learning	knowledge discovery	olap queries	online analytical processing
5	data mining techniques	building knowledge	multidimensional databases	analytical processing

Query	blu-ray		noodle		valet parking	
Method	SegPhrase+	Chunking	SegPhrase+	Chunking	SegPhrase+	Chunking
1	dvd	new microwave	ramen	noodle soup	valet	huge lot
2	vhs	lifetime warranty	noodle soup	asian noodle	self-parking	private lot
3	cd	recliner	rice noodle	beef noodle	valet service	self-parking
4	new release	battery	egg noodle	stir fry	free valet parking	valet
5	sony	new battery	pasta	fish ball	covered parking	front lot

Outline

1. Introduction to entity recognition and typing
2. Entity recognition: An overview and phrase mining approach
3. Entity typing: An overview and network mining approach
4. Trends and research problems

Entity Typing on **General-Domain**, **Formal** Corpora

- Assumptions

1. **Label**: A good amount of label data is available

2. **Feature**: Primitive NLP methods can provide decent & robust features
   (e.g., part-of-speech tags, noun phrases, dependency parse trees, …)

3. **Coverage**: Most mentioned entities can be found in knowledge bases

Entity Typing on General-Domain Corpora

A. Supervised Entity Typing

B. Semi-Supervised Entity Typing

C. Entity linking for typing

D. Weakly-Supervised Entity Typing

E. Distantly-Supervised Entity Typing

Entity Typing on General-Domain Corpora

A. Supervised Entity Typing

- Decision tree
- Support Vector Machine
- Sequence labeling models

B. Semi-Supervised Entity Typing

C. Entity linking for Entity Typing

Supervised Learning for Entity Typing

- **Diagram A**: I-O-B encoding for classification

Steve	Jobs	was	a	co-founder	of	Apple	Inc.
B-PER	I-PER	O	O	O	O	B-ORG	I-ORG

  - **Problem setting**: classify each token into corresponding I-O-B label

- **Diagram B**: detected entity mentions for classification

Steve Jobs	was a	co-founder	of	Apple Inc.
PER		O		ORG

  - **Problem setting**: classify each mention into corresponding type

Workflow of Supervised Entity Typing

- Training
  1. Collect a set of training documents/sentences
  2. Label each token (entity mention) for its entity class or other (O)
  3. Design feature extractors appropriate to the text and classes
  4. Train a classifier to predict the labels from the data

- Testing
  1. Receive testing document (a single document or a batch)
  2. Run trained classifier to label each token (entity mention)
  3. Appropriately output the recognized entities

Features for Classification (word-level)

Features	Examples
Case	- Starts with a capital letter - Word is all uppercased - The word is mixed case (e.g., ProSys, eBay)
Punctuation	- Ends with period, has internal period (e.g., St., I.B.M.) - Internal apostrophe, hyphen or ampersand (e.g., O'Connor)
Digit	- Digit pattern (*see section 3.1.1*) - Cardinal and Ordinal - Roman number - Word with digits (e.g., W3C, 3M)
Character	- Possessive mark, first person pronoun - Greek letters
Morphology	- Prefix, suffix, singular version, stem - Common ending (*see section 3.1.2*)
Part-of-speech	- proper name, verb, noun, foreign word
Function	- Alpha, non-alpha, n-gram (*see section 3.1.3*) - lowercase, uppercase version - pattern, summarized pattern (*see section 3.1.4*) - token length, phrase length

Feature is the king!

[Nadeau & Sekine 07]

Features for Classification (doc/corpus-level)

Features	Examples
Multiple occurrences	- Other entities in the context - Uppercased and lowercased occurrences (see 3.3.1) - Anaphora, coreference (see 3.3.2)
Local syntax	- Enumeration, apposition - Position in sentence, in paragraph, and in document
Meta information	- Uri, Email header, XML section, (see section 3.3.3) - Bulleted/numbered lists, tables, figures
Corpus frequency	- Word and phrase frequency - Co-occurrences - Multiword unit permanency (see 3.3.4)

Feature is the king!

- Distributional features
  - Each word will appear in contexts - induce a distribution over contexts
  - Cluster words based on how similar their distributions are
  - Use cluster IDs as features → great way to combat sparsity

[Nadeau & Sekine 07]

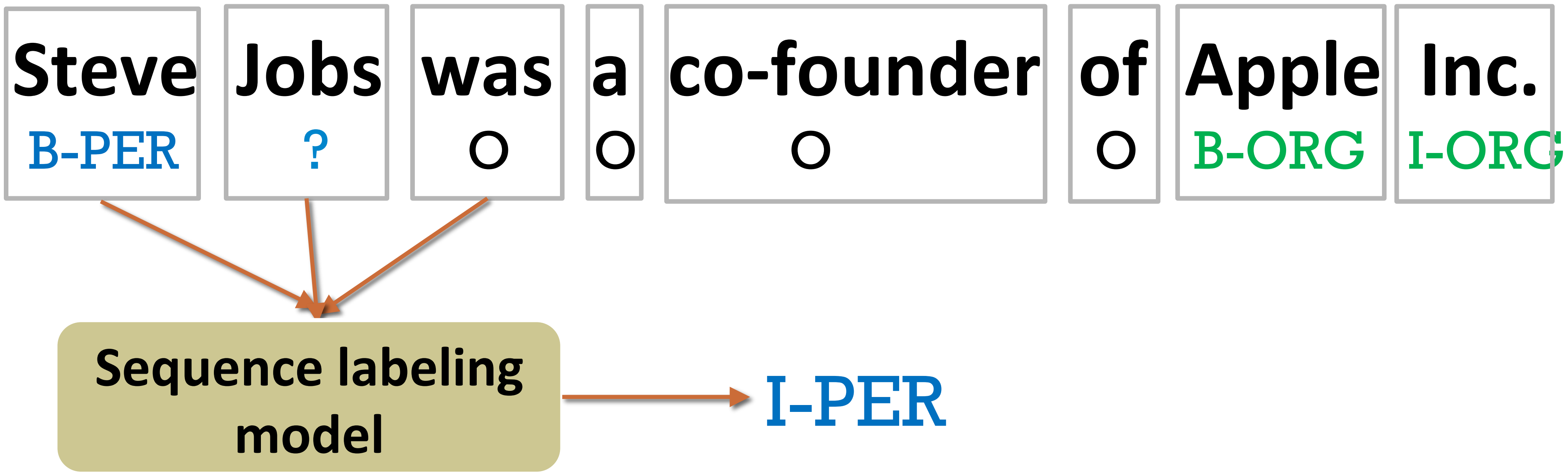
Standard Classification

- **Binary classification following diagram B**
- **Decision tree:**
  - Select feature to test at each node in the tree.
  - Top-down, greedy search through the space of possible decision trees. It picks the best attribute and never looks back to reconsider earlier choices.

- **Support vector machine:**
  - Negative examples are sampled from co-occurring entities which are not of the target types
  - Quadratic kernel gives the best performance

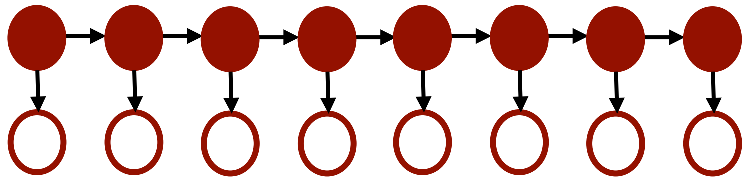
Sequence Labeling Models

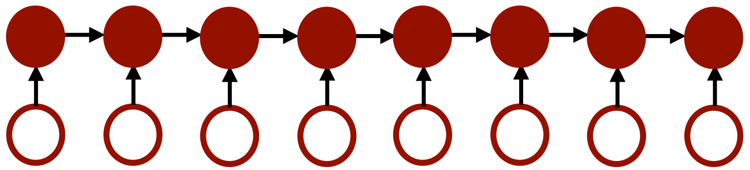
- Insights
  - vs. standard classification:
    - label depends not only on its corresponding observation but also possibly on *other observations* and *other labels* in the sequence

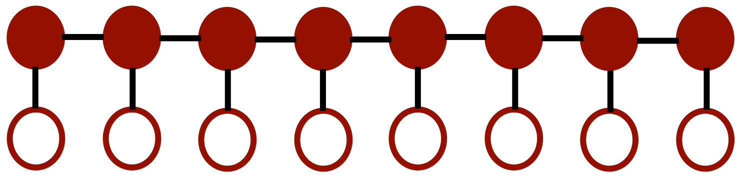


Model Trade-offs and Inference

	Speed	Discrim vs. Generative	Normalization
HMM	very fast	generative	local
MEMM	mid-range	discriminative	local
CRF	kinda slow	discriminative	global







- Greedy inference:
  - Fast; make commit errors
- Viterbi Inference
  - Dynamic programming or memorization
- Beam inference:
  - Fast; inexact (fall off global optimal sequence)

[Chris Manning 07]

Entity Typing on General-Domain Corpora

A. Supervised Entity Typing

B. Semi-Supervised Entity Typing

- Feature-level semi-supervised learning
- Semi-supervised sequence models

C. Entity linking for typing

Semi-Supervised Entity Typing

- **Goal**: leveraging large amount of unannotated corpus in addition to annotated corpus to augment model learning
  - More accurate results using similar amount of labeled data
  - Comparable performance with less amount of labeled data
- **Assumption**:
  - Data (feature) statistics from unannotated corpus can enhance model learning
- **Insights**
  - Features derived from unannotated corpus can be feed into supervised sequence models
  - Standard sequence models can be extended to model unlabeled data jointly

Feature-Level Semi-Supervised Learning

- Insights
  - Unsupervised word feature derived from a large corpus (both annotated and unannotated) can improve performance of existing supervised models
- Feature representations
  - Distributional word representation
    - Words from context windows
  - Clustering-based word representation
    - Brown clusters
  - Distributed word representations (word embedding)

System	Dev	Test
Baseline	94.16	93.79
HLBL, 50-dim	94.63	94.00
C&W, 50-dim	94.66	94.10
Brown, 3200 clusters	**94.67**	**94.11**
Brown+HLBL, 37M	94.62	94.13
C&W+HLBL, 37M	94.68	94.25
Brown+C&W+HLBL, 37M	94.72	94.15
Brown+C&W, 37M	94.76	94.35
Ando and Zhang (2005), 15M	-	94.39
Suzuki and Isozaki (2008), 15M	-	94.67
Suzuki and Isozaki (2008), 1B	-	**95.15**

Turian et al. ACL 2010.

Semi-Supervised Sequence Models

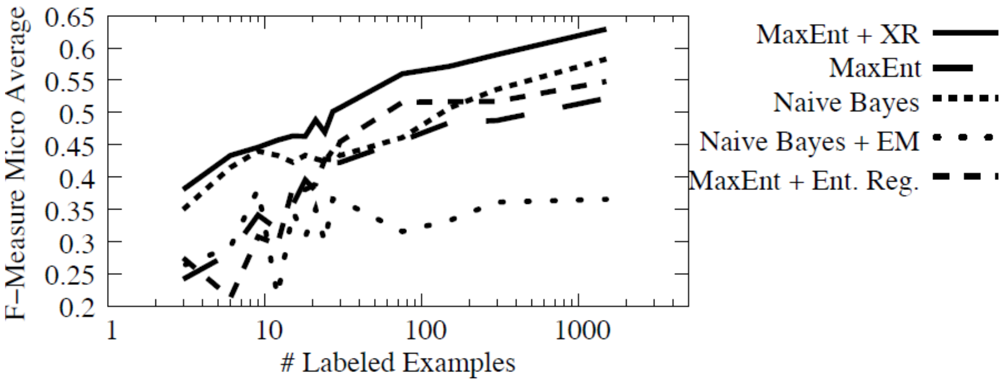
- **Goal**: incorporate unlabeled data into discriminative sequence model training in an effective way
  - Insight 1: semi-supervised CRF with entropy regularization on the unlabeled data

$$
\begin{aligned}
RL(\theta) &= \sum_{i=1}^{N} \log p_\theta(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}) - U(\theta) \qquad (2) \\
&+ \gamma \sum_{i=N+1}^{M} \sum_{\mathbf{y}} p_\theta(\mathbf{y}|\mathbf{x}^{(i)}) \log p_\theta(\mathbf{y}|\mathbf{x}^{(i)})
\end{aligned}
$$

  - Insight 2: use generalized expectation criteria to optimize CRF model

$$p(\mathbf{y}|\mathbf{x};\theta) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_k \theta_k \Psi_k(\mathbf{x},\mathbf{y})\right)$$

$$O(\theta;\mathcal{D}) = \sum_d \log p(\mathbf{y}_d|\mathbf{x}_d;\theta) - \frac{\sum_k \theta_k^2}{2\sigma^2}$$



Entity Typing on General-Domain Corpora

A. Supervised Entity Typing

B. Semi-Supervised Entity Typing

C. Entity linking for typing

Type Entities in Text

- Assumptions
  - Can be found in KB
  - No type ambiguity

Name	Source	# types	# entities	Hierarchy
Dbpedia	Wikipedia infoboxes	529	3M	Tree
YAGO2s	Wiki, WordNet,	350K	10M	Tree
Freebase	Miscellaneous	23K	**23M**	Flat
Probase	Web text	**2M**	5M	DAG

- Insights
  - **Context Similarity**: Contexts of the entity mention provide cues for linking it to the knowledge bases --- [Bunescu & Pascal 06] etc.
  - **Topic Coherence**: Entity mentions in a document/paragraph may share the same topics --- [Cucerzan 07] etc.
  - **Entity Popularity**: popular entity candidate is preferred to be linked to
  - Linking of multiple entity mentions in could be modeled jointly --- [Hoffart et al. 11] etc.

Limitation of Entity Linking

- Low recall of knowledge bases
- Sparse concept descriptors

82 of 900 shoe brands exist in Wiki

Michael Jordan ***won*** *the* ***best*** *paper* ***award***

Can we disambiguate entities without relying on knowledge bases?

- Yes! Exploit the redundancy in the corpus
  - Not relying on knowledge bases: targeted disambiguation of ad-hoc, homogeneous entities [Wang et al. 12]
  - Partially relying on knowledge bases: mining additional evidence in the corpus for disambiguation [Li et al. 13]

Entity Typing on **Domain-Specific**, **Informal Corpora**

- Assumptions

1. Very limited amount of (or no) labeled entity mentions are available for the corpus

2. Primitive NLP methods (e.g., NP chunking, dependency parsing) do not work well on the corpus

3. Only a small portion of entities in the corpus exist in knowledge bases

Entity Typing on Domain-Specific, Informal Corpora

A. Weakly-Supervised Entity Typing

- Pattern-based bootstrapping methods
- Graph-based semi-supervised learning

Unsupervised Entity Typing

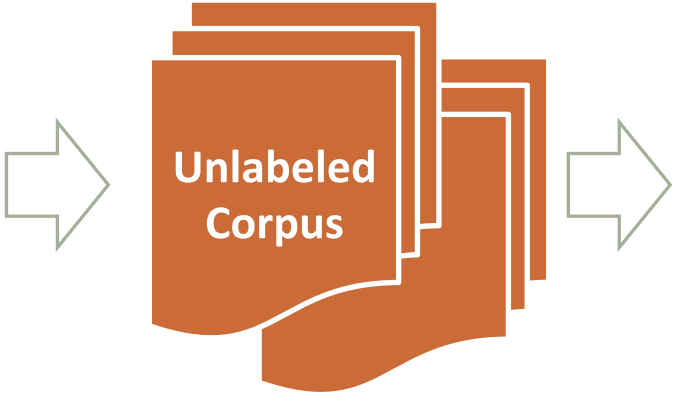
B. Distantly-Supervised Entity Typing

Weakly-Supervised Entity Typing

- Problem setting
  - A large unannotated corpus is available
  - A small set of labeled entity names (seeds) from the corpus are available
- Assumptions on labeled data (seeds)
  - Sufficiently frequent
  - NO type ambiguity
  - Cover all entity types

Seeds

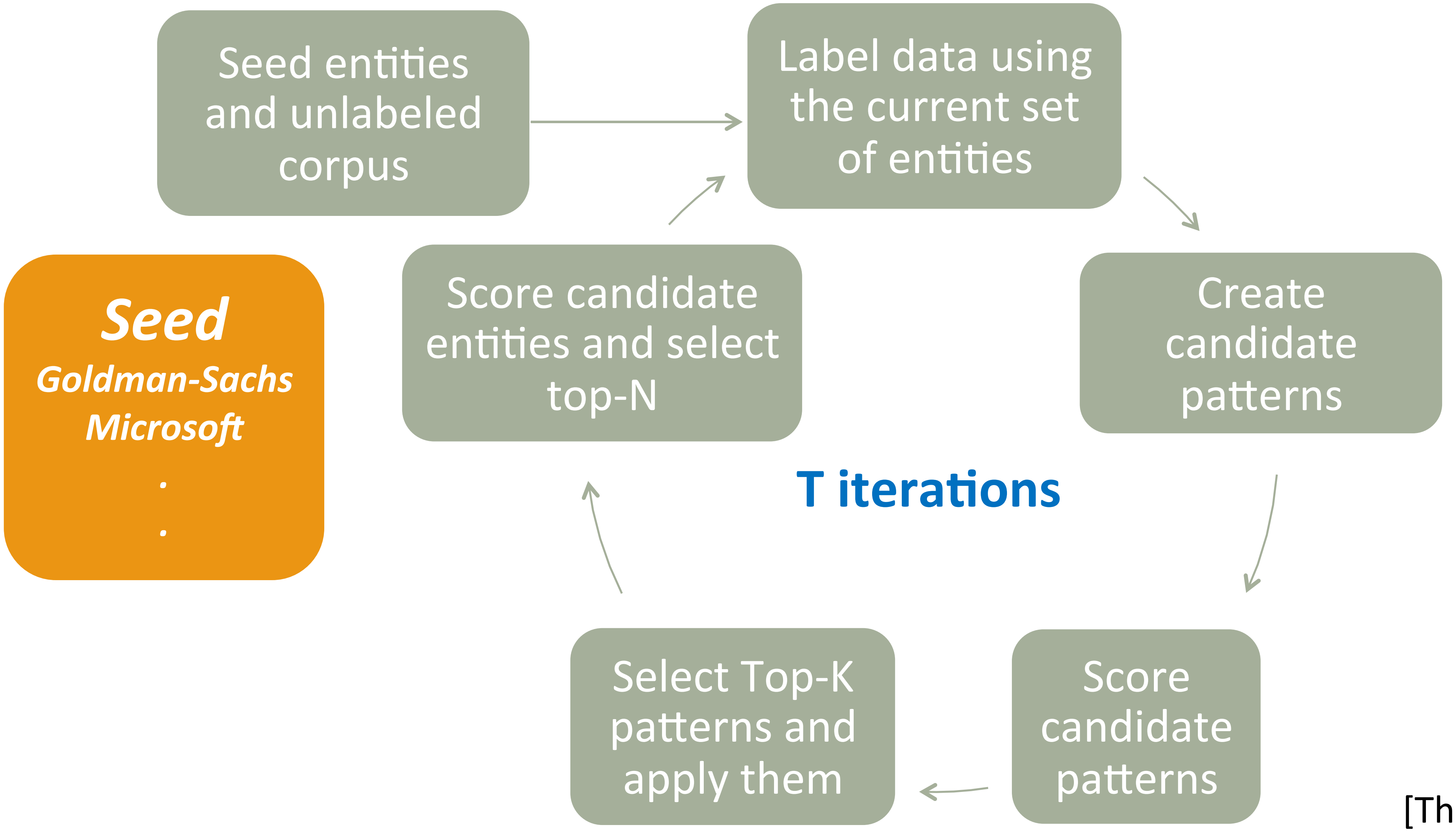
PERSON	LOCATION
Bush Ray Nagin	New Orleans Texas



Annotated entities

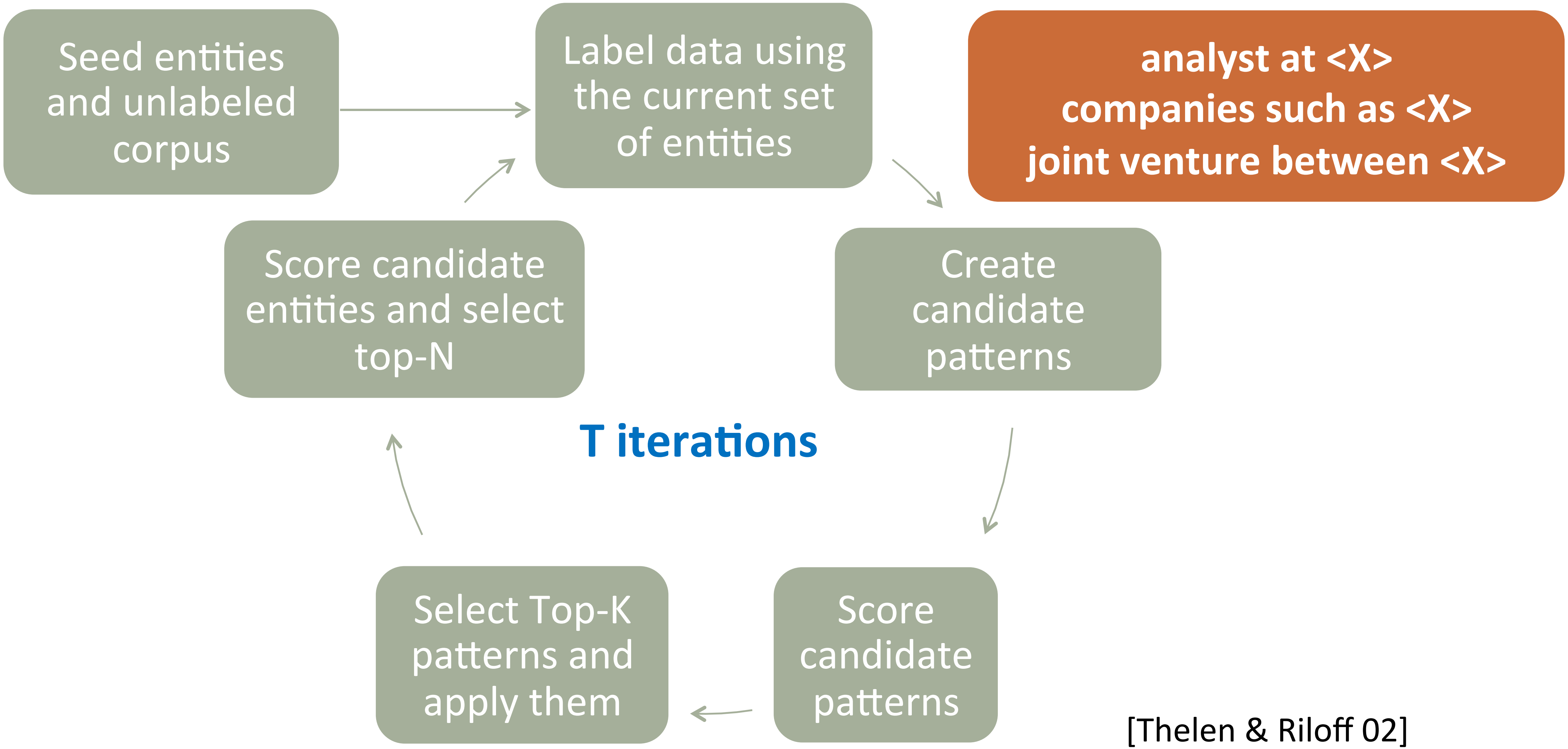
PERSON	LOCATION
Bush Ray Nagin Mayor Nagin Blanco …	New Orleans Texas Louisiana Washington D.C. …

Pattern-Based Bootstrapping Methods



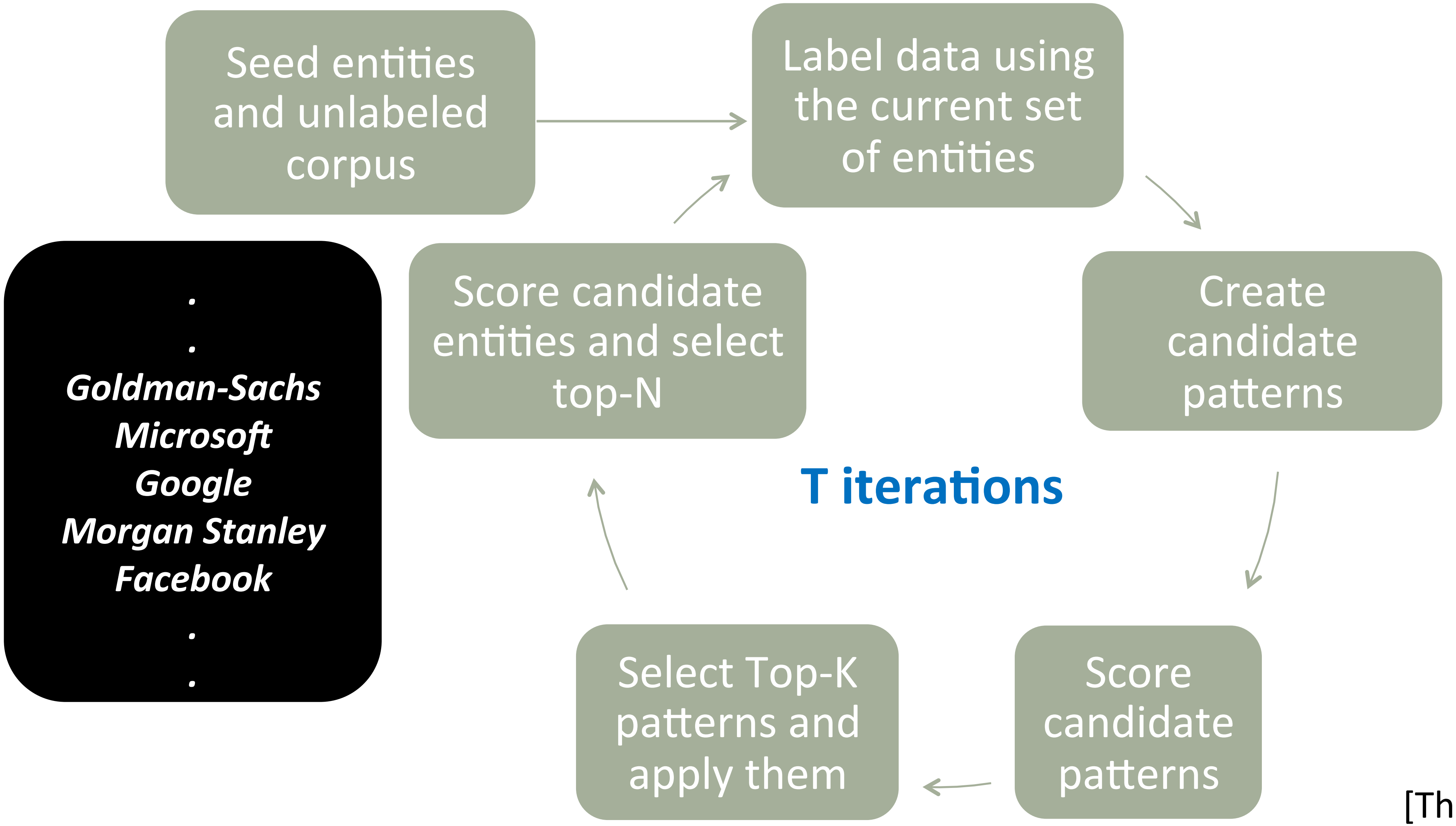
[Thelen & Riloff 02]

Pattern-Based Bootstrapping Methods



[Thelen & Riloff 02]

Pattern-Based Bootstrapping Methods



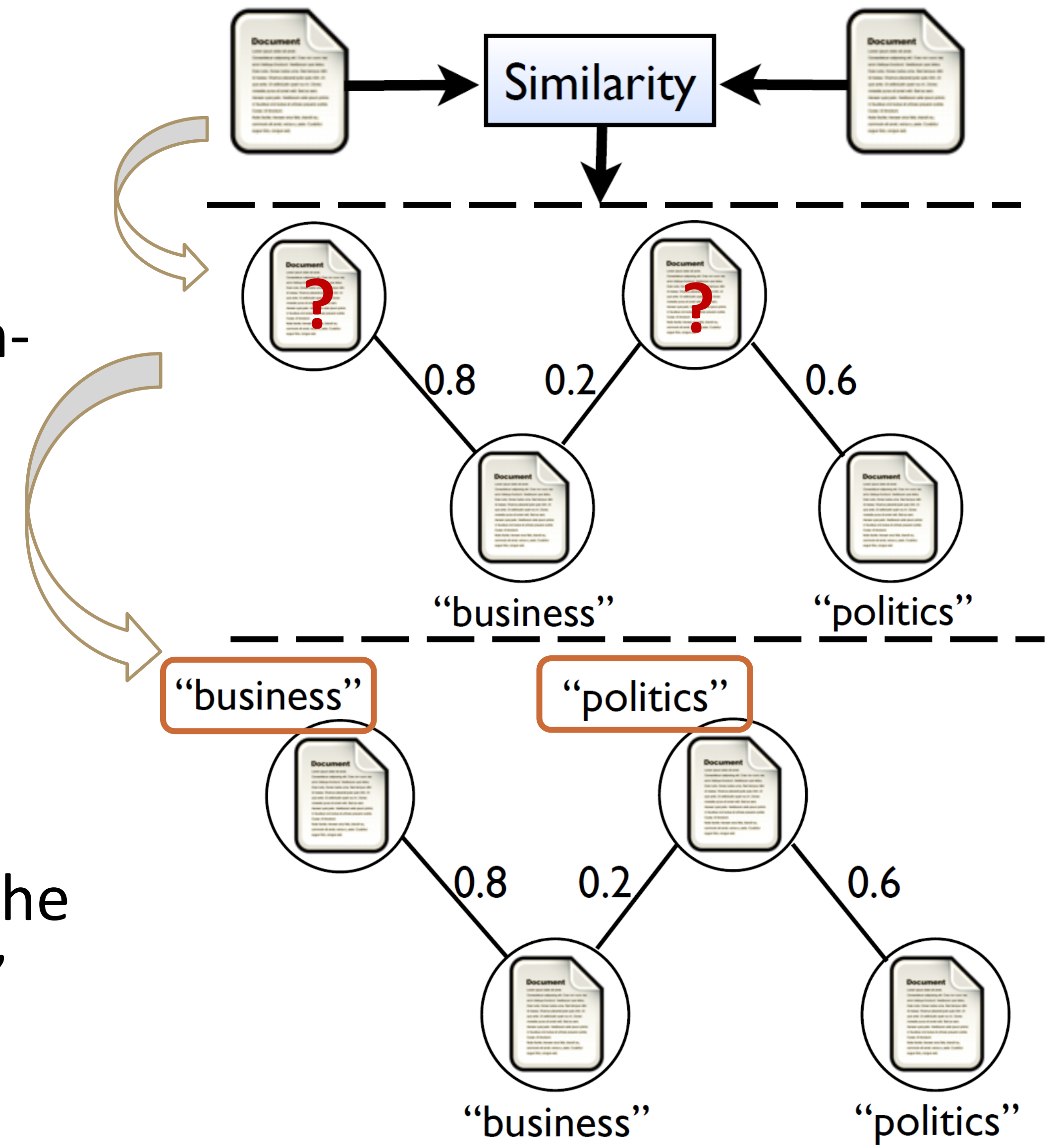
[Thelen & Riloff 02]

Pattern-Based Bootstrapping Methods

- **Assumption**:
  - **Mutual exclusion**: positive examples (i.e., entity names) for one type are negative examples for other types
- **Key questions**:
  - How to induce effective patterns given entities → pattern induction
  - How to evaluate the extracted patterns? → pattern scoring
  - How to evaluate the extracted entities? → entity promotion
- **Limitations**
  - Each entity name is assigned with **only one type**
    - Cannot handle ambiguous names---"*Washington D.C.*"
  - Error aggregation

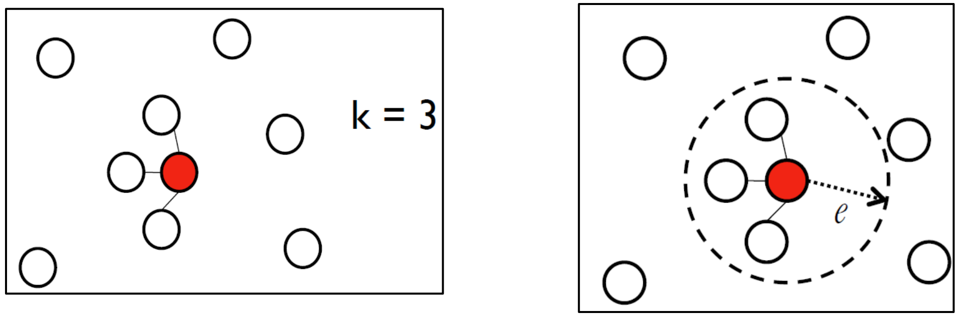
Graph-Based Semi-Supervised Learning

- **Insights**
  - Many text corpus can be naturally and uniformly represented by a graph
  - Entity typing can then be modeled as graph-based semi–supervised learning problem
- **Assumptions**
  - Quality entity candidates are already extracted
  - **[Smoothness Assumption]**
    - "If two instances are similar according to the graph, then their labels should be similar."

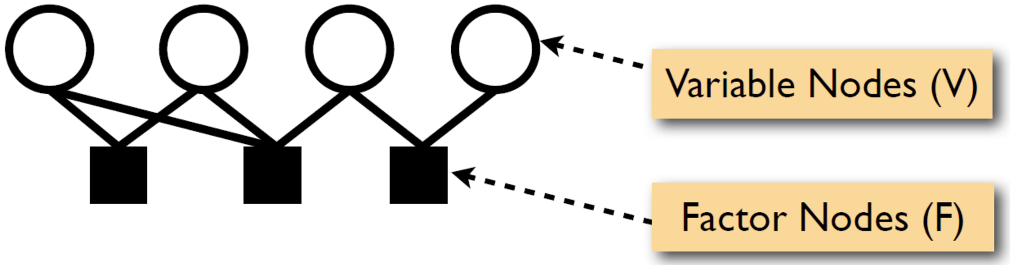


Graph-Based Semi-Supervised Learning

- **Graph construction**
  - Edge formation & weighting



- **Learning Algorithms**
  - Label propagation: Random walk, Graph Laplacian, LP-ZGL [Zhu et al. 03]
  - Factor graph model [Kschischang et al. 01]
  - Manifold regularization [Belkin et al., 2006]



- **Advantage:** Flexible to model various sources and signals uniformly
- **Limitations**
  - Cannot decide the exact type for each entity mention (name ambiguity)
  - Sensitive to seeds

Entity Typing on Domain-Specific, Informal Corpora

A. Weakly-Supervised Entity Typing

Unsupervised Entity Typing

- Structured Generative Model
- Multi-view Embedding

B. Distantly-Supervised Entity Typing

Why Unsupervised Entity Typing?

- Set free from obtaining labeled data
  - Assumptions on unlabeled data:
    - Hidden (cluster) structures reflect the entity types

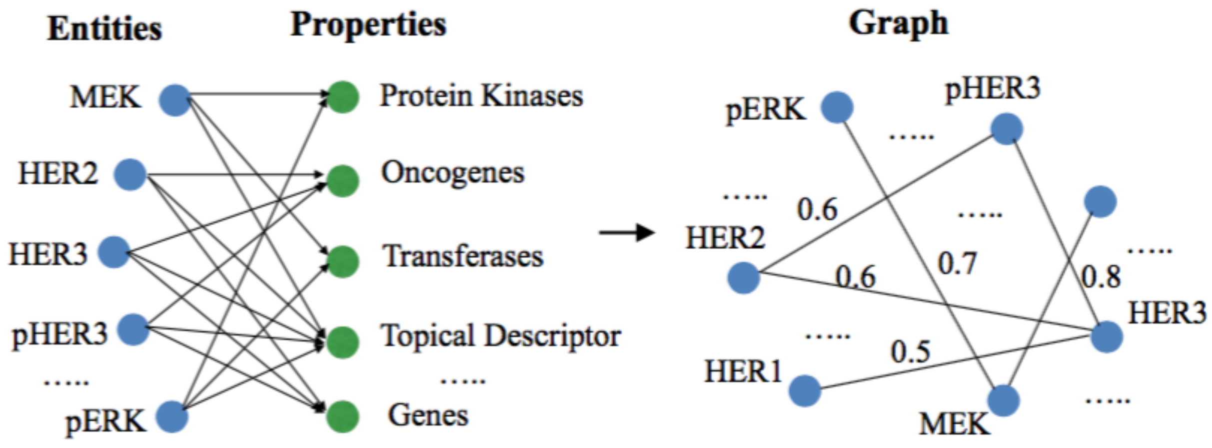
- Methods
  - Structured generative models [Elsner, et. al. 2009]
    - Complex inference algorithm (probabilistic context-free grammar)
  - Multi-view Embedding [Huang et. al., 2016]
    - Cont.

Multi-View Embedding based Entity Typing

- **Heuristic 1**: The types of *common* entities can be effectively captured by their general semantics → Entity Embedding
- **Heuristic 2**: The types of *uncommon* and *polysemantic* entities can be inferred by their specific contexts→ Context-based Embedding
- **Heuristic 3**: The types of *domain specific* entities largely depend on domain-specific knowledge→ knowledge–based embedding

Hierarchical Entity Clustering and Naming:

- Hierarchical X-means Clustering
- Entity linking → type naming



Entity Typing on Domain-Specific, Informal Corpora

A. Weakly-Supervised Entity Typing

Unsupervised Entity Typing

B. Distantly-Supervised Entity Typing

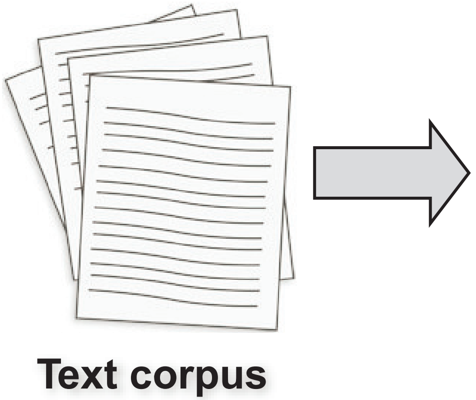
- Multi-label Multi-class classification methods
- Label propagation methods
- **ClusType**: A phrase and network mining approach
- **PLE**: Label Noise Reduction in Entity Typing

Why Distantly-Supervised Entity Typing?

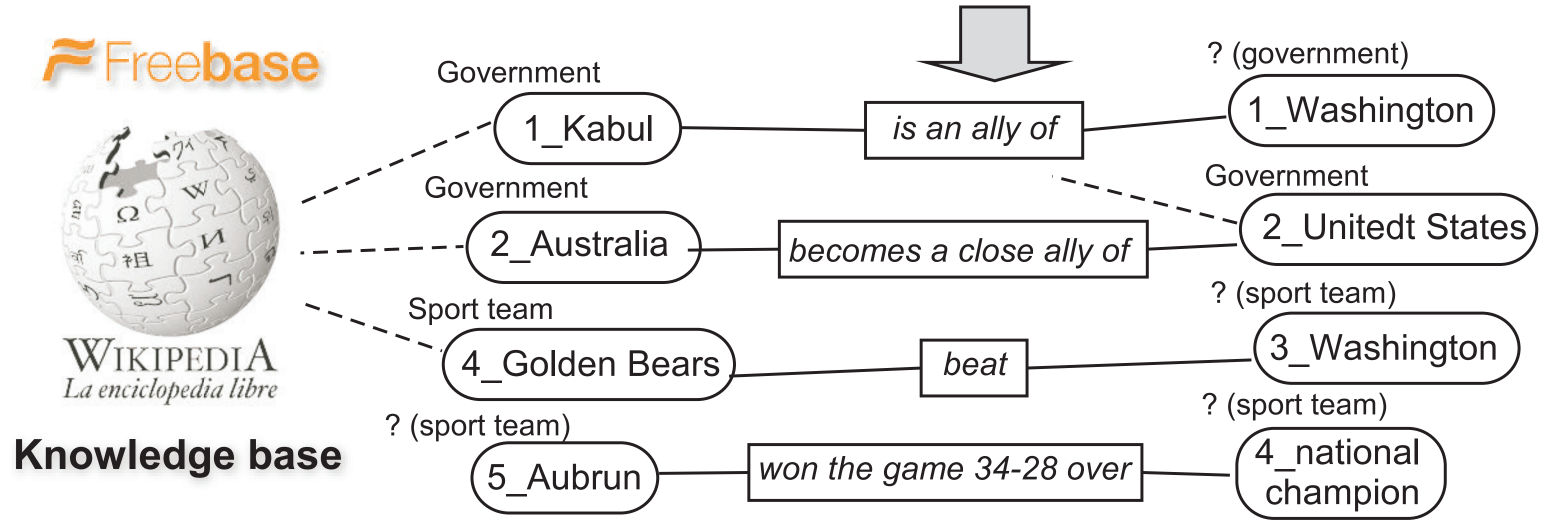
- Weakly-supervised methods still require human annotations
  - Assumptions on labels:
    - Sufficient occurrences in the corpus
    - Semantically unambiguous
    - Cover all entity types
  - Can we get rid of human supervision, and make it fully automatic?
    - Rich entity information in knowledge bases → “distant” supervision for entity typing

Typical Workflow of Distant Supervision

- Detect entity mentions from text
- Map candidate mentions to KB entities of target types
- Use confidently mapped {mention, type} to infer types of remaining candidate mentions

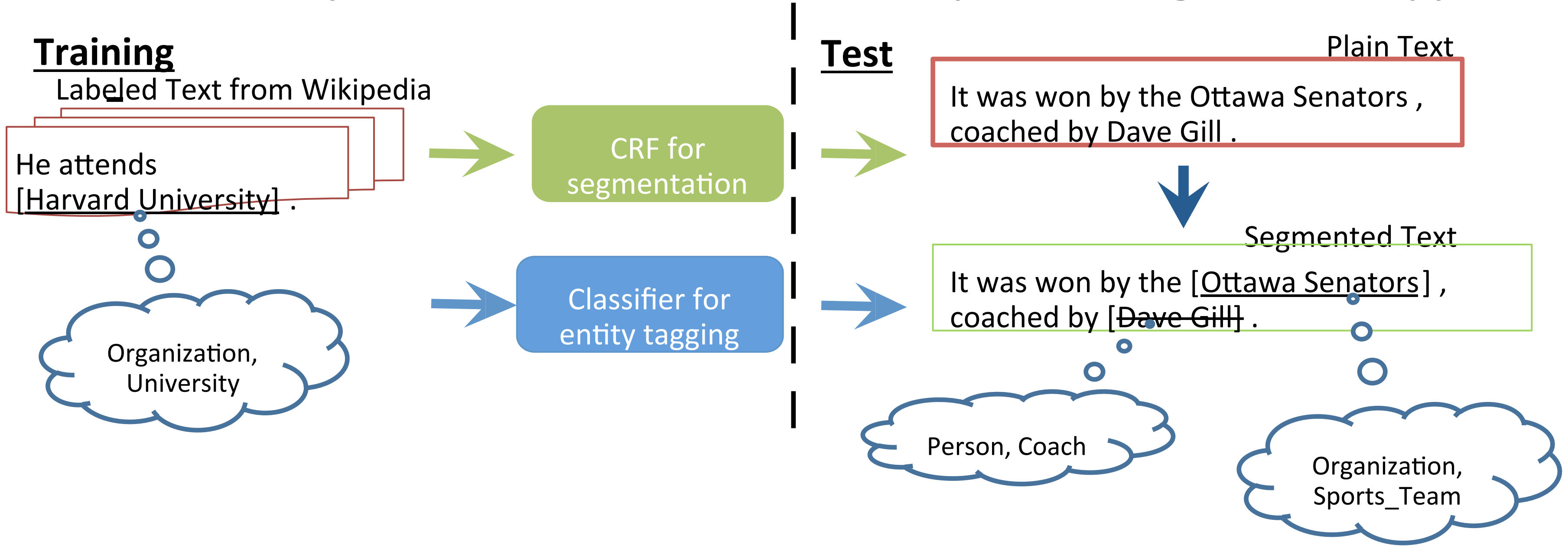


ID	Document Text
1	... has concerns whether **Kabul** *is an ally of* **Washington.**
2	... **Australia** *becomes a close ally of* the **United States**. ...
3	He *has offices in* **Washington, Boston** and **San Francisco**.
4	... The **Cardinal** *will share the title with* **California** if the **Golden Bears** *beat* **Washington** later Saturday. ...
5	... **Auburn** *won the game 34-28 over* the defending **national champions**. ...



Multi-Class Multi-Label Classification

- Assumptions:
  - Entity mentions are already recognized from text
  - Features for classifiers can be robustly computed from the corpus
- Insights:
  - Allow one entity mention to have multiple fine-grained types



Label Propagation Methods

- Assumptions
  - Entity mentions are pre-extracted for the corpus
  - There is no name ambiguity
    - Each entity surface name is assigned with one type
- Insights
  - Linked entities candidates serve as seeds
  - Contextual information (e.g., relation phrases) server as bridges to propagate type information between entity candidates
- Existing work
  - NNPLB [Lin et al. 12]: noun phrase classifier + propagation on OpenIE triples

Challenge I: Domain Restriction

- Most existing work assume entity mentions are already extracted by existing entity detection tools
  - Usually trained on general-domain corpora like news articles (clean, grammatical)
  - Make use of various linguistics features (e.g., semantic parsing structures)
  - Do not work well on specific, dynamic or emerging domains (*e.g.*, tweets, Yelp reviews)
    - E.g., "in-and-out" from Yelp review may not be properly detected

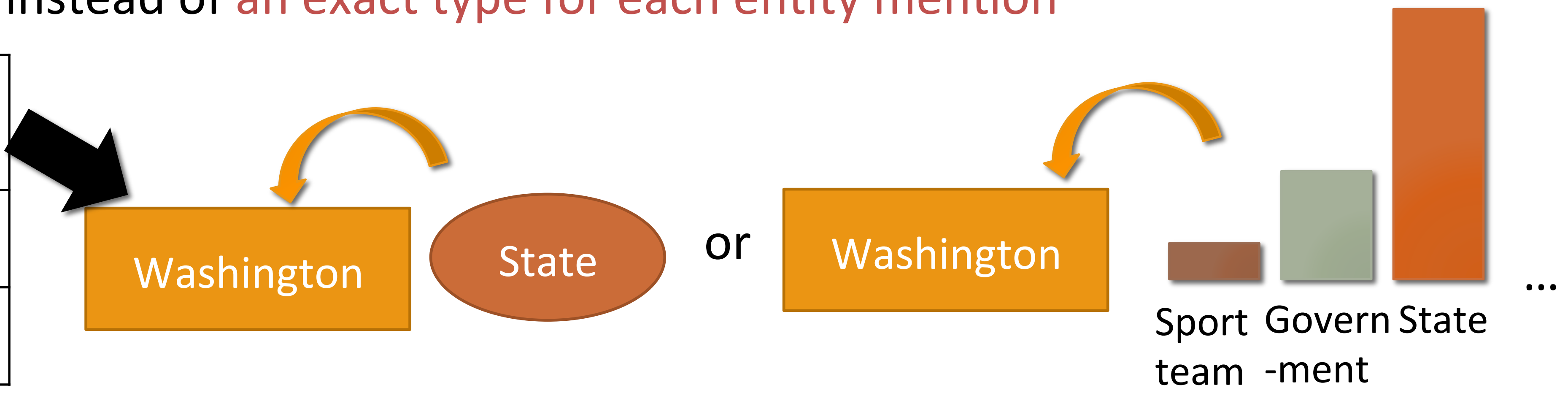
Challenge II: Name Ambiguity

- Multiple entities may share the same surface name

While Griffin is not the part of Washington's plan on Sunday's game, …	Sport team
…has concern that Kabul is an ally of Washington.	U.S. government
He has office in Washington, Boston and San Francisco	U.S. capital city

- Previous methods simply output a single type/type distribution for each surface name, instead of an exact type for each entity mention

While Griffin is not the part of Washington's plan on Sunday's game, …
… news from Washington indicates that the congress is going to…
It is one of the best state parks in Washington.



Challenge III: Context Sparsity

- A variety of contextual clues are leveraged to find sources of shared semantics across different entities
  - Keywords, Wiki concepts, linguistic patterns, textual relations, …
- There are often many ways to describe even the same relation between two entities

ID	Sentence	Freq
1	The magnitude 9.0 quake **caused widespread devastation in [Kesennuma city]**	12
2	… tsunami that **ravaged [northeastern Japan]** last Friday	31
3	The resulting tsunami **devastate [Japan]**’s northeast	244

- Previous methods have difficulties in handling entity mention with sparse (infrequent) context

ClusType: The Solution Ideas

Domain-agnostic phrase mining algorithm

- Extracts candidate entity mentions with minimal linguistic/domain assumption → address domain restriction

Do not simply merge entity mentions with *identical surface names*

- Model each mention based on its surface name and context, in a scalable way → address name ambiguity

Mine synonymous *relation phrase* co-occurring with entity mentions

- Helps form connecting bridges among entities that do not share identical context, but share synonymous relation phrases → **address context sparsity**

Ren et al., SIGKDD 2015

Framework Overview

1. Perform **phrase mining** on a POS-tagged corpus to extract candidate entity mentions and relation phrases
2. **Construct a heterogeneous graph** to encode our insights on modeling the type for each entity mention
3. **Collect seed entity mentions as labels** by linking extracted mentions to the KB
4. Estimate type indicator for unlinkable candidate mentions with the proposed **type propagation integrated with relation phrase clustering** on the constructed graph

Candidate Generation

- An efficient phrase mining algorithm incorporating:
  - **Global significance score**: Filter low-quality candidates;
  - **Generic POS tag patterns**: remove phrases with improper syntactic structure
- Example output of candidate generation on NYT news articles

> Over:RP the weekend the system:EP dropped:RP nearly inches of snow in:RP western Oklahoma:EP and at:RP [Dallas Fort Worth International Airport]:EP sleet and ice caused:RP hundreds of [flight cancellations]:EP and delays. ...... It is forecast:RP to reach:RP [northern Georgia]:EP by:RP [Tuesday afternoon]:EP, Washington:EP and [New York]:EP by:RP [Wednesday afternoon]:EP, meteorologists:EP said:RP.

EP: entity mention candidate; RP: relation phrase

- Entity detection performance comparison with an NP chunker

Method	NYT		Yelp		Tweet	
	Prec	Recall	Prec	Recall	Prec	Recall
Our method	**0.469**	**0.956**	**0.306**	**0.849**	0.226	**0.751**
NP chunker	0.220	0.609	0.296	0.247	**0.287**	0.181

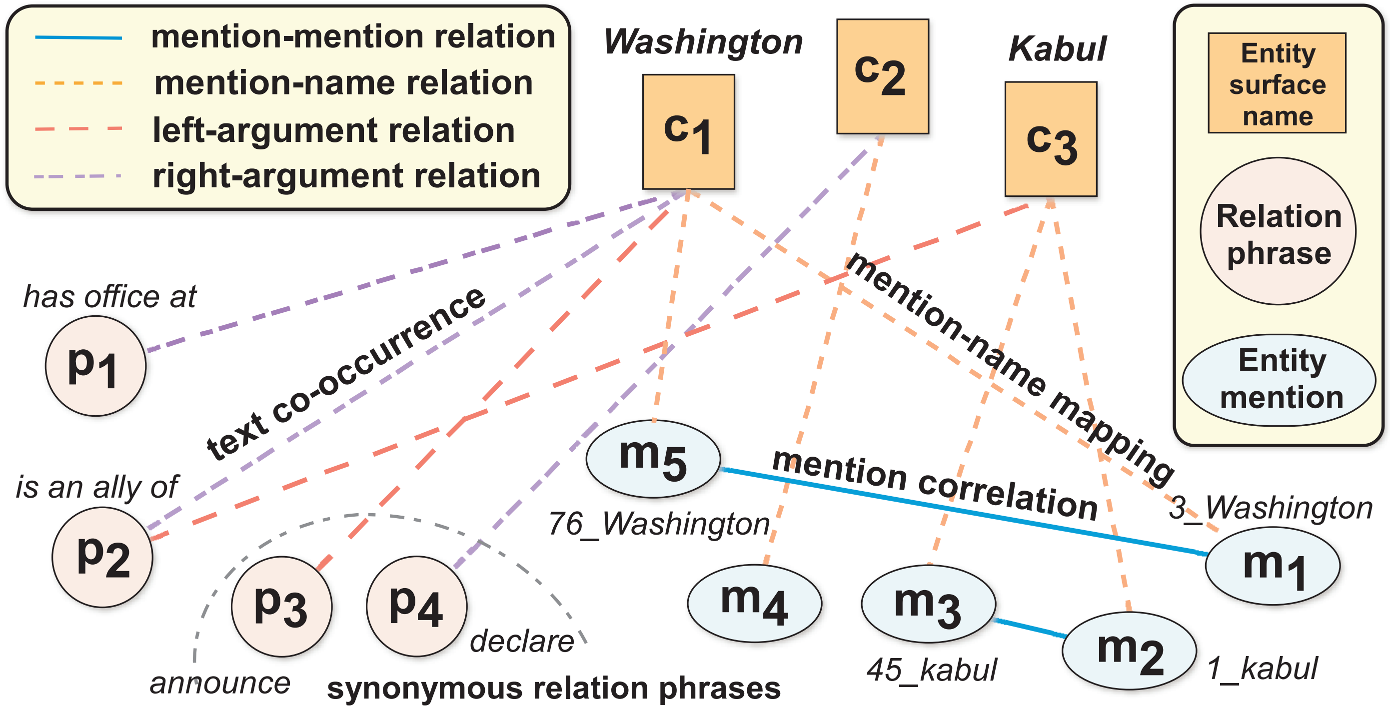
Construction of Heterogeneous Graphs

- With three types of objects extracted from corpus: candidate entity mentions, entity surface names, and relation phrases
  - We can construct a heterogeneous graph to **enforce several hypotheses for modeling type of each entity mention** (introduced in the following slides)

Entity mentions are kept as individual objects **to be disambiguated**

Linked to entity surface names & relation phrases

Basic idea (Smoothness Assumption): the more two objects are likely to share the same label, the larger the weight will be associated with their connecting edge

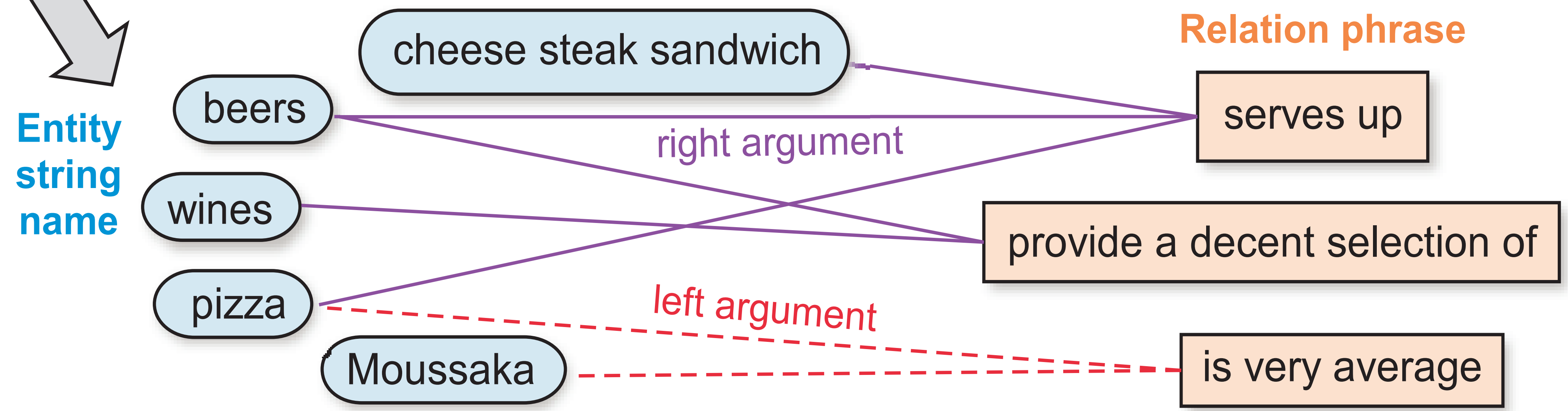


Entity Name-Relation Phrase Subgraph

- Aggregated co-occurrences between entity surface names and relation phrases across corpus
  - → use connected edges as bridges to propagate type information

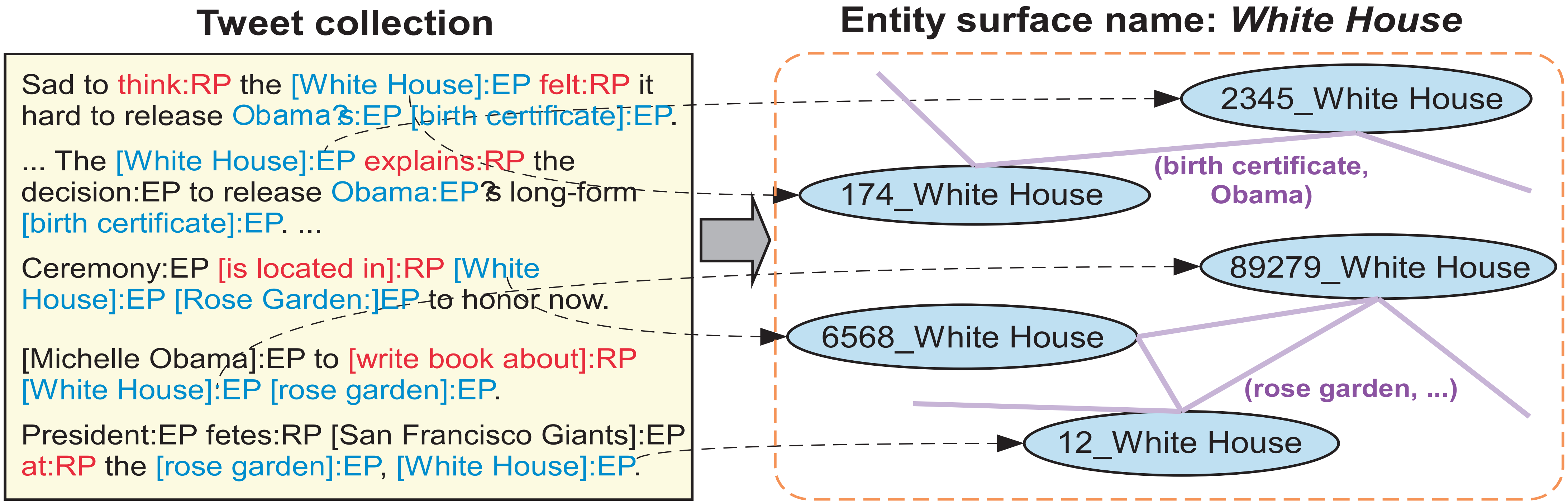
Text Corpus

This place:EP [serves up]:RP the best [cheese steak sandwich]:EP west of:RP the Mississippi:EP. ...... Four Peaks:EP [serves up]:RP some beers:EP and great eats:RP. ...... They [provide a decent selection of]:RP beers:EP and high-end wines:EP. ...... Tons of:RP [places in the valley]:EP, [Jimmy Joes]:EP [serves up]:RP good PIZZA:EP. ..... Pizza:EP [is very average]:RP. ...... The Moussaka:EP [is very average]:RP with:RP no flavor:EP.....



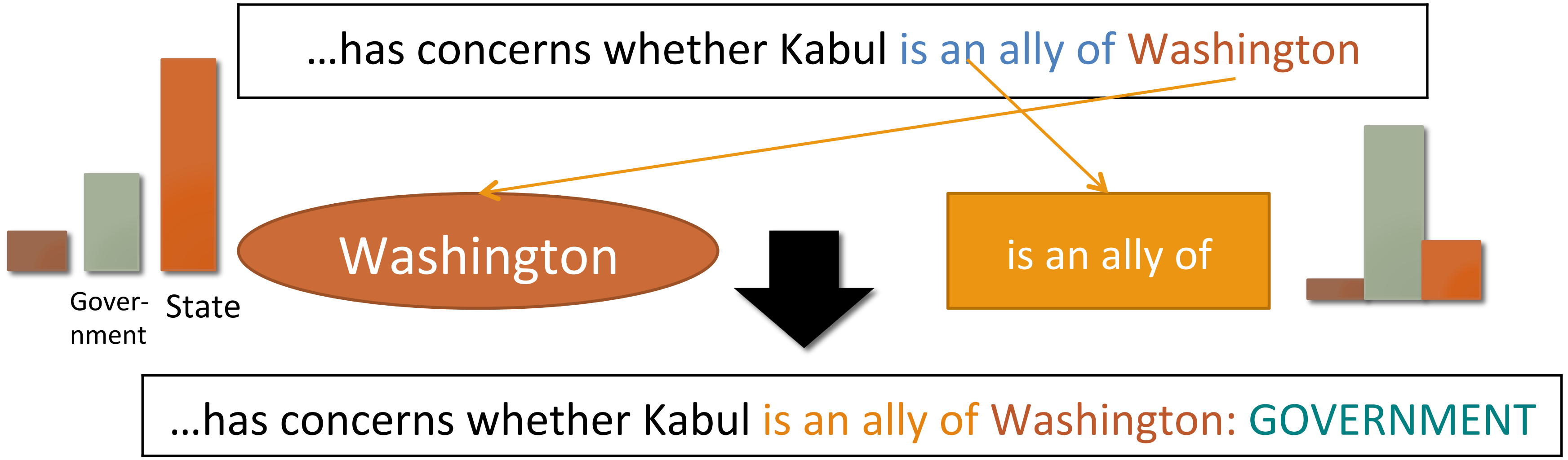
Mention Correlation Subgraph

- An entity mention may have *ambiguous* types and *ambiguous* relation phrases
  - E.g., "*White house*" and "*felt*" in the first sentence in the Figure
- Other co-occurring mentions may provide good hints to the type of an entity mention
  - E.g., "*Obama*" and "*rose garden*" in the Figure



Modeling Type for Entity Mention

- Both the **entity surface name** and the **surrounding relation phrases** provide strong cues on the types of a candidate entity mention
  - → Model by: (1) type indicator of its surface name
  - → (2) the type signatures of its surrounding relation phrases (more details in the following slides)



Relation Phrase Clustering

- Softly clustering synonymous relation phrases:

  → the type signatures of frequent relation phrases can help infer the type signatures of infrequent (sparse) ones that have similar cluster memberships

- **Signals in previous methods**:
  - String similarity & context similarity → may be insufficient to resolve two relation phrases
- **New signal**: Arguments' type information is particular helpful in such case
- **Multi-view clustering** method to incorporate all features

  → further integrated with the graph-based type propagation in a mutually enhancing framework, based on following hypothesis

Two Tasks Mutually Enhance Each Other

Type propagation on heterogeneous graph

Derived entity argument types serve as **good feature** for clustering relation phrases

Multi-view relation phrase clustering

Propagate type information among entities bridges via synonymous relation phrases

Mutually enhancing each other; leads to quality recognition of unlinkable entity mentions

Comparing ClusType with Other Methods and Its Variants

Performance comparison on three datasets

Data sets	NYT			Yelp			Tweet		
Method	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Pattern [9]	0.4576	0.2247	0.3014	0.3790	0.1354	0.1996	0.2107	0.2368	0.2230
FIGER [16]	0.8668	0.8964	0.8814	0.5010	0.1237	0.1983	**0.7354**	0.1951	0.3084
SemTagger [12]	0.8667	0.2658	0.4069	0.3769	0.2440	0.2963	0.4225	0.1632	0.2355
APOLLO [29]	0.9257	0.6972	0.7954	0.3534	0.2366	0.2834	0.1471	0.2635	0.1883
NNPLB [15]	0.7487	0.5538	0.6367	0.4248	0.6397	0.5106	0.3327	0.1951	0.2459
ClusType-NoClus	0.9130	0.8685	0.8902	0.7629	0.7581	0.7605	0.3466	0.4920	0.4067
ClusType-NoWm	0.9244	0.9015	0.9128	0.7812	0.7634	0.7722	0.3539	**0.5434**	0.4286
ClusType-TwoStep	0.9257	0.9033	0.9143	0.8025	0.7629	0.7821	0.3748	0.5230	0.4367
ClusType	**0.9550**	**0.9243**	**0.9394**	**0.8333**	**0.7849**	**0.8084**	0.3956	0.5230	**0.4505**

- Compare with Stanford NER (trained on general-domain) on types **PER, LOC, ORG**

Method	**NYT**	**Yelp**	**Tweet**
Stanford NER [6]	0.6819	0.2403	0.4383
ClusType-NoClus	0.9031	0.4522	0.4167
ClusType	**0.9419**	**0.5943**	**0.4717**

Example Output and Relation Phrase Clusters

Table 7: Example output of ClusType and the compared methods on the Yelp dataset.

ClusType	SemTagger	NNPLB
The best **BBQ:Food** I've tasted in **Phoenix:LOC** ! I had the **[pulled pork sandwich]:Food** with **coleslaw:Food** and **[baked beans]:Food** for lunch. ...	The best BBQ I've tasted in **Phoenix:LOC** ! I had the pulled **[pork sandwich]:LOC** with **coleslaw:Food** and **[baked beans]:LOC** for lunch. ...	The best **BBQ:Loc** I've tasted in **Phoenix:LOC** ! I had the pulled pork **sandwich:Food** with coleslaw and baked **beans:Food** for **lunch:Food**. ...
I only go to **ihop:LOC** for **pancakes:Food** because I don't really like anything else on the menu. Ordered **[chocolate chip pancakes]:Food** and a **[hot chocolate]:Food**.	I only go to ihop for pancakes because I don't really like anything else on the menu. Ordered **[chocolate chip pancakes]:LOC** and a **[hot chocolate]:LOC**.	I only go to ihop for pancakes because I don't really like anything else on the menu. Ordered chocolate chip pancakes and a hot chocolate.

- Extracts more mentions and predicts types with higher accuracy

Table 8: Example relation phrase clusters and their corpus frequency from the NYT dataset.

ID	Relation phrase
1	recruited by (5.1k); employed by (3.4k); want hire by (264)
2	go against (2.4k); struggling so much against (54); run for re-election against (112); campaigned against (1.3k)
3	looking at ways around (105); pitched around (1.9k); echo around (844); present at (5.5k);

- Not only synonymous relation phrases, but also both sparse and frequent relation phrase can be clustered together
- → boosts sparse relation phrases with type information of frequent relation phrases

Fine-Grained Entity Typing

- **Fine-grained Entity Typing**: Type labels for a mention forms a "*type-path*" (not necessarily ending in a leaf node) in a given (tree-structured) type hierarchy

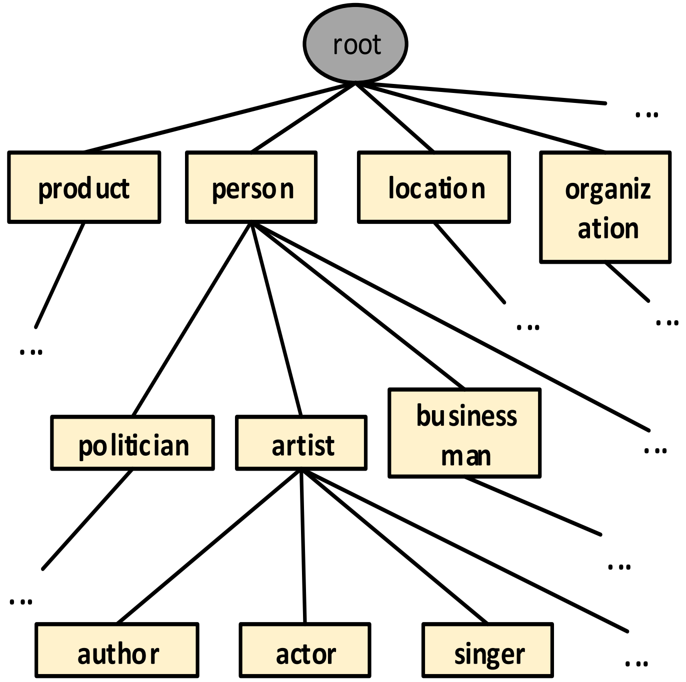
ID	Sentence
S1	Republican presidential candidate ***Donald Trump*** spoke during a campaign event in Rock Hill.
S2	***Donald Trump***'s company has threatened to withhold up to $1 billion of investment if the U.K. government decides to ban his entry into the country.
S3	In ***Trump***'s TV reality show, "The Apprentice", 16 people competed for a job.
…	…

Type-path

Person → politician

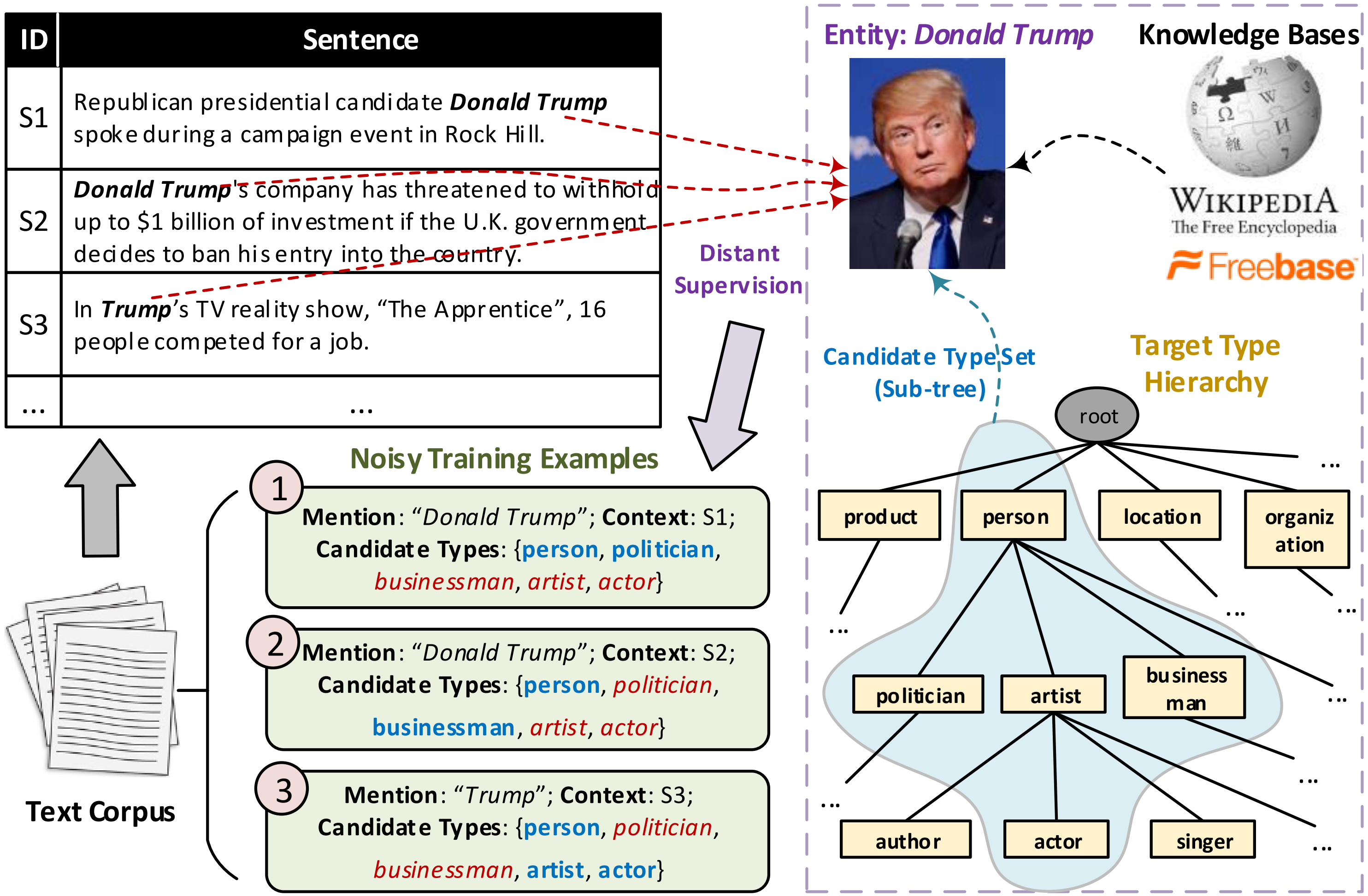
Person → businessman

Person → artist → actor



- Manually annotating training corpora with **100+** entity types
  - Expensive & Error-prone
- **Current practice**: use distant supervision to ***automatically labeled training corpora***

Label Noise Reduction in Distant Supervision



Donald Trump is mentioned in sentences S1-S3.

- Distant supervision
  - Assign *same* types (blue region) to *all* the mentions
  - Does not consider ***local contexts*** when assigning type labels
  - Introduce ***label noise*** to the mentions

The types assigned to entity Trump include person, artist, actor, politician, businessman, while only {person, politician} are correct types for the **mention *"Trump"* in S1**

Ren et. al., SIGKDD 2016

Label Noise in Entity Typing (cont.)

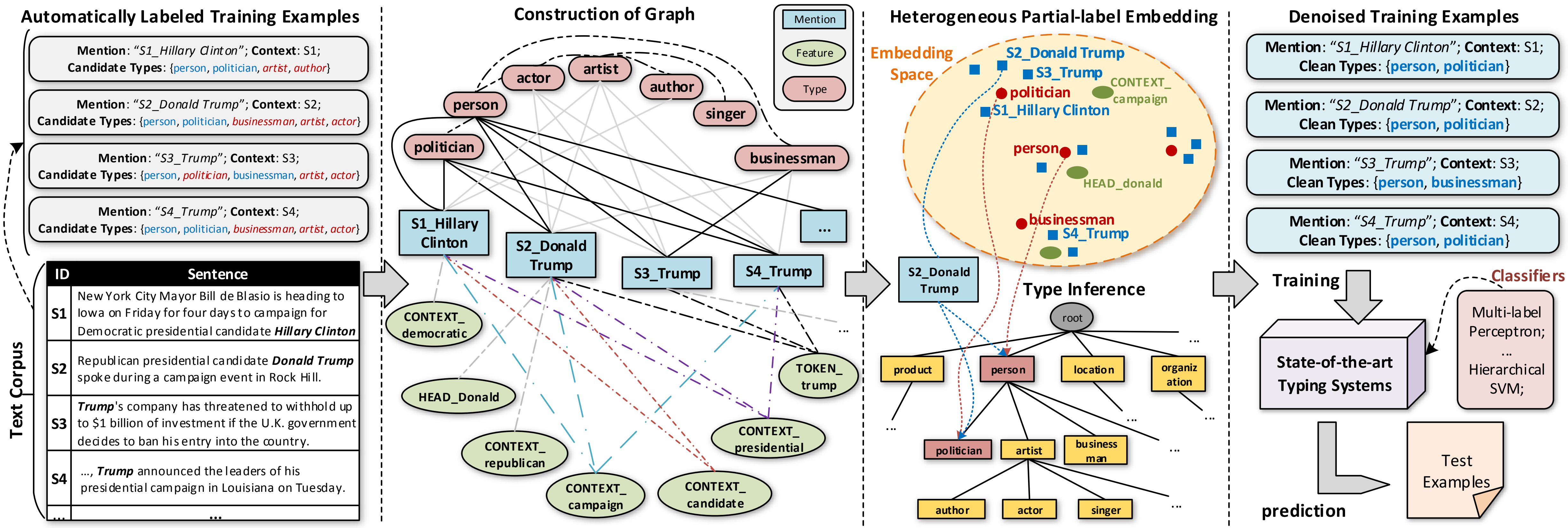
- Current typing systems either **ignore this issue**
  - assume all candidate labels obtained by supervision are “true” labels

Dataset	Wiki	OntoNotes	BBN	NYT
# of target types	113	89	47	446
(1) noisy mentions (%)	27.99	25.94	22.32	51.81
(2a) sibling pruning (%)	23.92	16.09	22.32	39.26
(2b) min. pruning (%)	28.22	8.09	3.27	32.75
(2c) all pruning (%)	45.99	23.45	25.33	61.12

- Or use **simple pruning heuristics** to ***delete*** mentions with conflicting types
  - aggressive deletion of mentions → significant loss of training data

The larger the target type set, the more severe the loss!

Label Noise Reduction by Partial-Label Embedding (PLE)



1. Generate text features and construct a heterogeneous graph
2. Perform joint embedding of the constructed graph G into the same low-dimensional space
3. For each mention, search its candidate type sub-tree in a top-down manner and estimate the true type-path from learned embedding

Example Output

- Example output on news articles

Text	*NASA* says it may decide by tomorrow whether another space walk will be needed ...	... the board of *directors* which are composed of twelve members directly appointed by the ***Queen***.
Wiki Page	https://en.wikipedia.org/wiki/NASA	https://en.wikipedia.org/wiki/Elizabeth_II
Cand. type set	person, artist, location, structure, organization, company, news_company	person, artist, actor, author, person_title, politician
WSABIE	person, artist	person, artist
PTE	organization, company, news_company	person, artist
PLE	organization, company	person, person_title

- PLE predicts fine-grained types with better accuracy (e.g., person_title)
- and avoids from overly-specific predictions (e.g., news_company)

Extrinsic Evaluation on Fine-Grained Entity Typing

- Adopting ***PLE-denoised training corpora*** → **50%+** improvement in accuracy for the two state-of-the-art typing systems (FIGER & HYENA)

Typing System	Noise Reduction Method	Wiki Acc	Wiki Ma-F1	Wiki Mi-F1	OntoNotes Acc	OntoNotes Ma-F1	OntoNotes Mi-F1	BBN Acc	BBN Ma-F1	BBN Mi-F1
N/A	PL-SVM [20]	0.428	0.613	0.571	0.465	0.648	0.582	0.497	0.679	0.677
N/A	CLPL [2]	0.162	0.431	0.411	0.438	0.603	0.536	0.486	0.561	0.582
HYENA [35]	Raw	0.288	0.528	0.506	0.249	0.497	0.446	0.523	0.576	0.587
	Min [7]	0.325	0.566	0.536	0.295	0.523	0.470	0.524	0.582	0.595
	All [7]	0.417	0.591	0.545	0.305	0.552	0.495	0.495	0.563	0.568
	WSABIE-Min [34]	0.199	0.462	0.459	0.400	0.565	0.521	0.524	0.610	0.621
	PTE-Min [28]	0.238	0.542	0.522	0.452	0.626	0.572	0.545	0.639	0.650
	PLE-NoCo	0.517	0.672	0.634	0.496	0.658	0.603	0.650	0.709	0.703
	PLE	**0.543**	**0.695**	**0.681**	**0.546**	**0.692**	**0.625**	**0.692**	**0.731**	**0.732**
FIGER [14]	Raw	0.474	0.692	0.655	0.369	0.578	0.516	0.467	0.672	0.612
	Min	0.453	0.691	0.631	0.373	0.570	0.509	0.444	0.671	0.613
	All	0.453	0.648	0.582	0.400	0.618	0.548	0.461	0.636	0.583
	WSABIE-Min	0.455	0.646	0.601	0.425	0.603	0.546	0.481	0.671	0.618
	PTE-Min	0.476	0.670	0.635	0.494	0.675	0.618	0.513	0.674	0.657
	PLE-NoCo	0.543	0.726	0.705	0.547	0.699	0.639	0.643	0.753	0.721
	PLE	**0.599**	**0.763**	**0.749**	**0.572**	**0.715**	**0.661**	**0.685**	**0.777**	**0.750**

FIGER: Fine-Grained Entity Recognition, AAAI 2012.

HYENA: Hierarchical Type Classification for Entity Names, COLING 2012.

Outline

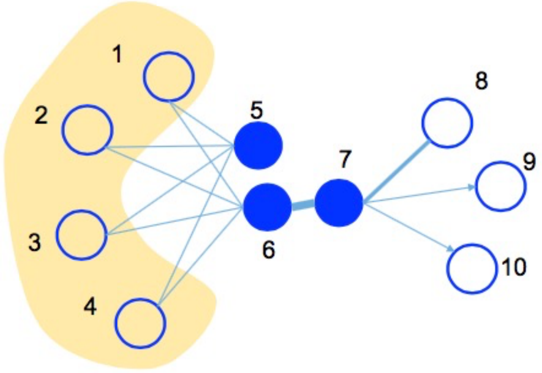
1. Introduction to entity recognition and typing
2. Entity recognition – overview and phrase mining approach
3. Entity typing – overview and network mining approach
4. Trends and research problems

Trends and Research Problems

- Exploration of the Power of Entity Recognition and Typing
  - Mining Hidden Relationship Among Entities
- Mining Attributes and Values for Knowledge Network Construction
  - Mining the Universe of Attributes: The Google Approach
- Construction of Heterogeneous Information Networks from Entities, Attributes and Relationships
- Looking forward to the Future

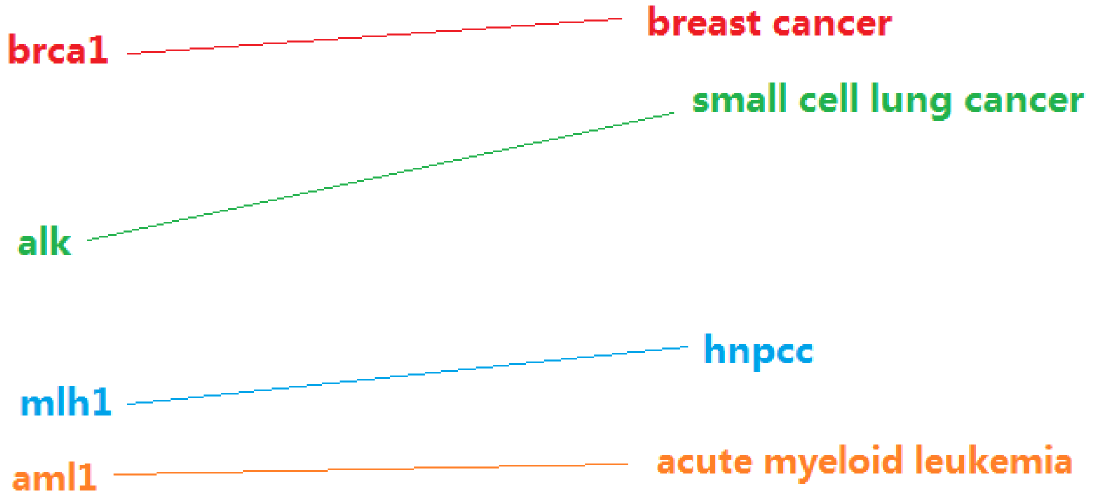
Relationship Discovery for Network Building

- Automatic extraction of relationships between different biological entities from biological research papers (e.g., PubMed)
    - Gene – Disease; Drug - Disease; Drug - Pathway; Drug - Target gene
- Challenges
    - Entity recognition: Most biological entities consist of multiple words
        - E.g., Non-small Cell Lung Cancer, Acute Myeloid Leukemia
    - Sparsity: Most biological entities co-occur only a few times in research papers
        - Most relationships are not explicitly described in papers
    - Few labeled data
- Key ideas
    - Phrase mining
    - Learn phrase-based network embedding from massive data
        - Using: LINE (Tang et al., Large-scale Information Network Embedding, WWW'15)
    - Calculate network embedding



Key Property to Learn Embedding & Experiments

- Key Property to Learn Embedding
  - The lines between genes and diseases are parallel
  - Given a seed pair (A,B) and a query X, we can find an entity Y which satisfies
    - $(A,B) \approx (X, Y)$
    - $Y = Argmax\{sim(\, B - A + X \,, Y \,)\}$



- Experimental Settings
  - Sample 10% Pubmed abstracts
  - Detect phrases by using a 200K phrase list
  - Build a co-occurrence network for all words and phrases
  - Learn entity embedding from the co-occurrence network

Results: Extracted Relations (from 10% PubMed Abstracts)

Relation	Seed Pair	Query Entity	Top Ranked Entities
Gene-Disease	Breast Cancer, BRCA1	Acute Myeloid Leukemia	AML1, E2A-PBX1, NPM1, RUNX1, PBX1
		Acute Lymphocytic Leukemia	E2A-PBX1, NPM1, EVI1, BCL6, ALL1
		HNPCC	MLH1, MSH6, hMSH2, hMLH1, MSH2
	BRCA1, Breast Cancer	ALK	Small Cell Lung Cancer, Non-small Cell Lung Cancer
		AML1	Leukemia, AML, CML
		MLH1	Colorectal Cancer, HNPCC, Colon Cancer
Drug-Disease	Leukemia, Doxorubicin	Small Cell Lung Cancer	Paclitaxel, Gemcitabine, Docetaxel, Cisplatin
		Depressive Disorder	Sertraline, Desvenlafaxine, Duloxetine, Paliperidone
		HIV	Zidovudine, Ritonavir, Lamivudine, Atazanavir
	Doxorubicin, Leukemia	Aspirin	Peptic Ulcer Bleeding, Venous Thromboembolic
		Sertraline	Depressive Disorder, Social Anxiety Disorder
		Penicillin	Bacterial Meningitis, Scabies, Streptococcus

Top Ranked Molecules for Heart Diseases

Disease	Top Ranked Molecules and their scores
Cerebrovascular Accident	Alpha-galactosidase A, Brain-derived Neurotrophic Factor, Tissue-type Plasminogen Activator Methylenetetrahydrofolate Reductase, Matrix Metalloproteinase-9 5.903, 5.595, 4.945, 2.710, 2.680
Ischemic Heart Disease	Cholesteryl Ester Transfer Protein, Apolipoprotein A-I, Adiponectin, Lipoprotein Lipase, Myeloperoxidase 4.597, 3.989, 3.651, 3.302, 3.240
Cardiomyopathy	Interferon Gamma, Interleukin-4, Interleukin-17a, Tumor Necrosis Factor, Titin 3.336, 2.809, 2.729, 2.549, 2.349
Arrhythmia	Methionine Synthase, Ryanodine Receptor 2, Platelet-Activating Factor Acetylhydrolase, Potassium Voltage-gated Channel Subfamily H Member 2, Gap Junction Alpha-1 Protein, 3.799, 3.354, 1.740, 2.730, 1.872
Valve Dysfunction	Mineralocorticoid Receptor, Elastin, Tropomyosin Alpha-1 Chain, Myosin-Binding Protein C Cardiac-type, Platelet-Activating Factor Acetylhydrolase 3.276, 2.380, 2.332, 1.704, 1.611
Congenital Heart Disease	Fibrillin-1, Plakophilin-2, Tyrosine-protein Phosphatase Non-receptor Type 11, Arachidonate 5-Lipoxygenase-activating Protein, Catechol O-methyltransferase 4.920, 3.208, 2.667, 2.036, 1.791

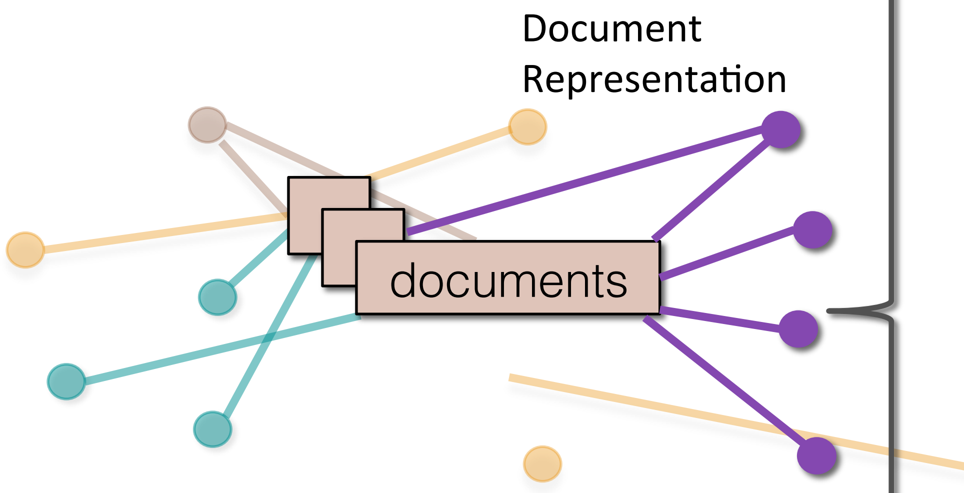
Mining PubMed abstracts (1995-2015) with keyword: “Cardiovascular Diseases”

Mining Disease-* Relations for Heart Diseases

Relation and Seed Pair	Query Entity	Top Ranked Entities and Their Scores
Disease-Drug Heart Disease : Aspirin	Cerebrovascular	Clopidogrel, Anti-platelet, Ticlopidine, Ticagrelor, prasugrel 0.7170, 0.6955, 0.6922, 0.6759, 0.6661
	Ischemic Heart Disease	Anti-platelet, Clopidogrel, Ticlopidine, Aspirin-Clopidogrel, Plavix 0.7785, 0.7732, 0.7532, 0.7481, 0.7473
	Coronary Heart Disease	Clopidogrel, Anti-platelet, Aspirin-Clopidogrel, Prasugrel, Ticagrelor 0.7855, 0.7606, 0.7248, 0.7148, 0.7086
	Dilated Cardiomyopathy	Clopidogrel, Ticlopidine, Prasugrel, Plavix, ACE Inhibitor 0.7649, 0.7436, 0.6968, 0.6765, 0.6586
	Valvular Heart Disease	Anti-platelet, Ticlopidine, Clopidogrel, Aspirin-Clopidogrel, Plavix 0.7750, 0.7749, 0.7668, 0.7529, 0.7260
	Arrhythmia	Clopidogrel, Anti-platelet, Ticlopidine, Thienopyridine, Ticagrelor 0.7589, 0.7411, 0.6958, 0.6838, 0.6788
Disease-Gene Breast Cancer : brca1	Cerebrovascular	tlr9, myh15, abca1, uts2, abcg1 0.6980, 0.6952, 0.6829, 0.6790, 0.6770
	Ischemic Heart Disease	sdc2, mth1, uts2, kcnn4, hspa8 0.7624, 0.7604, 0.7443, 0.7431, 0.7390
	Coronary Heart Disease	apoc2, uts2, apoh, lox1, mth1 0.7911, 0.7765, 0.7754, 0.7718, 0.7615
	Dilated Cardiomyopathy	calm1, actn2, ankrd1, col1a2, fhl2 0.7385, 0.7370, 0.7368, 0.7314, 0.7298
	Valvular Heart Disease	col11a2, ndufs2, kcnn4, ncam1, myl1 0.6938, 0.6815, 0.6765, 0.6750, 0.6717
	Arrhythmia	atp1a2, casq2, ndufs2, gpd1l, kcne4 0.6772, 0.6745, 0.6743, 0.6713, 0.6705

LAKI: Representing Documents via Latent Keyphrase Inference

- Jialu Liu, Xiang Ren, Jingbo Shang, Taylor Cassidy, Clare Voss and Jiawei Han, "Representing Documents via Latent Keyphrase Inference", WWW'16
- **Document Representation**



Words:
dbscan, methods, clustering, process, …

Topics:
[k-means, clustering, clusters, dbscan, …]
[clusters, density, dbscan, clustering, …]
[machine, learning, knowledge, mining, …]

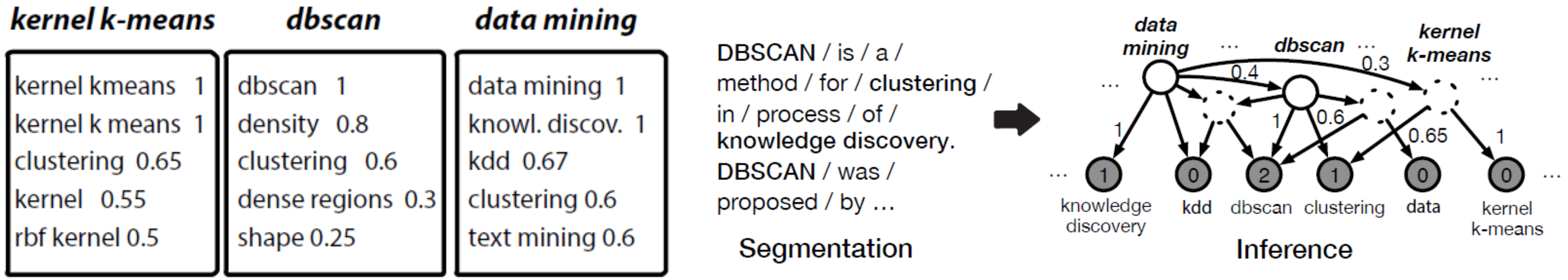
Knowledge base concepts:
data mining: /m/0blvg
clustering analysis: /m/031f5p
dbscan: /m/03cg_k1

Document keyphrase:
dbscan: [dbscan, density, clustering, ...]
clustering: [clustering, clusters, partition, ...]
data mining: [data mining, knowledge, ...]

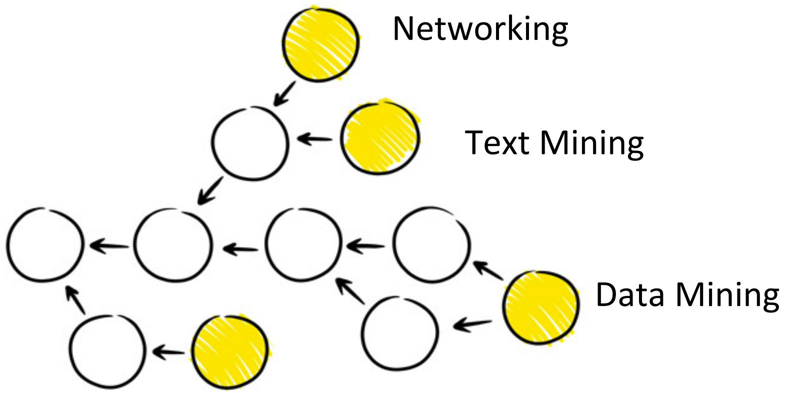
- A document can be represented by
  - A set of works, topics, KB concepts, Keyphrases, …

Document Representation Using Keyphrases: General Ideas

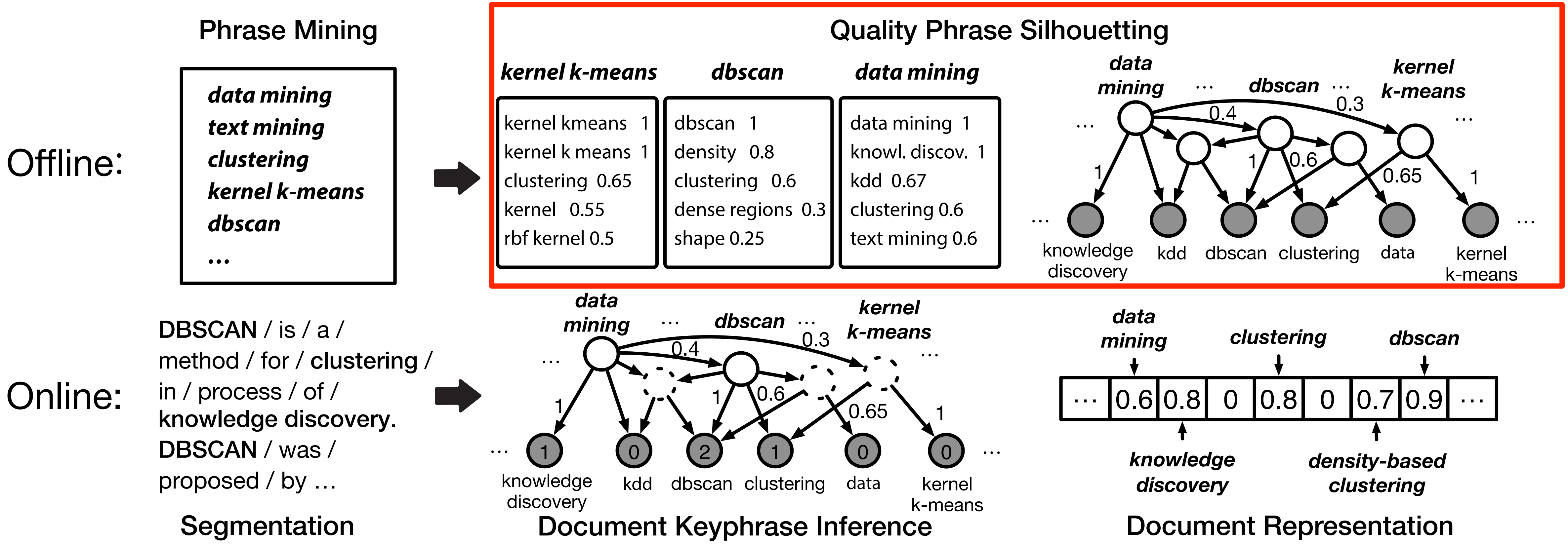
- How to identify document keyphrases?
  - Powered by Bayesian Inference on "Quality Phrase Silhouette"
    - Quality Phrase Silhouette: Topic centered on quality phrase
      - "Reverse" topic models
      - "Pseudo content" for quality phrase



- How to deal with relationship between quality phrases?
  - Phrases are interconnected as a Directed Acyclic Graph



Framework for Latent Keyphrase Inference (LAKI)



LAKI: Experiment Setting

- Two text-related tasks to evaluate document representation quality
  - Phrase relatedness
  - Document classification
- Two datasets:
- Methods:
  - **ESA** (Explicit Semantic Analysis)
  - **KBLink** uses link structure in Wikipedia
  - **BoW** (bag-of-words)
  - **ESA-C:** extends ESA by replacing Wiki with domain corpus
  - **LSA** (Latent Semantic Analysis)
  - **LDA** (Latent Dirichlet Allocation)
  - **Word2Vec** is a neural network computing word embeddings
  - **EKM** uses explicit keyphrase detection

Dataset	#Docs	#Words	Content type
Academia	0.43M	28M	title & abstract
Yelp	0.47M	98M	review

Method	Semantic Space	Input Source
ESA	KB concepts	KB
KBLink	KB concepts	KB
BoW	Words	-
ESA-C	Documents	Corpus
LSA	Topics	Corpus
LDA	Topics	Corpus
Word2Vec	-	Corpus
EKM	Explicit Keyphrases	Corpus
LAKI	Latent Keyphrases	Corpus

LAKI: Experimental Results

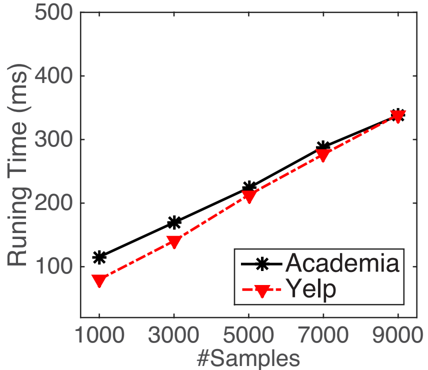
- **Phrase Relatedness Correlation**

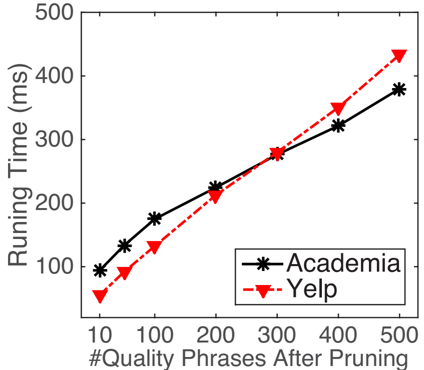
Method	Academia (w/ phrase)	Yelp (w/ phrase)
ESA	37.61 (-)	46.56 (-)
KBLink	36.37 (-)	35.94 (-)
BoW	48.05 (45.60)	51.26 (45.97)
ESA-C	39.75 (42.20)	49.13 (54.51)
LSA	72.50 (79.22)	66.55 (78.57)
LDA	77.27 (80.52)	75.55 (82.65)
EKM	45.46	40.57
LAKI	**84.42**	**90.58**

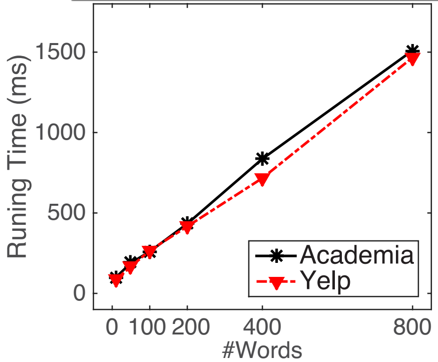
- **Document Classification**

Method	Academia (w/ phrase)	Yelp (w/ phrase)
ESA	0.4320 (-)	0.4567 (-)
KBLink	0.1878 (-)	0.4179 (-)
ESA-C	0.4905 (0.5243)	0.4655 (0.5029)
LSA	0.5877 (0.6383)	0.6700 (0.7229)
LDA	0.3610 (0.5391)	0.3928 (0.5405)
Word2Vec	0.6674 (0.7281)	0.7143 (0.7419)
LAKI	**0.7504**	**0.7609**

- **Time Complexity**







Case Study

- Query on phrases
  - Academia
  - Yelp

- Query on short documents (paper titles or sentences)
  - Academia
  - Yelp

Query	*LDA*	*BOA*
Keyphrases	linear discriminant analysis, latent dirichlet allocation, topic models, topic modeling, face recognition, sda, latent dirichlet, generative model, topic, subspace models, ...	boa steakhouse, bank of america, stripsteak, agnolotti, credit card, santa monica, restaurants, wells fargo, steakhouse, prime rib, bank, vegas, las vegas, cash, cut, dinner, bank, money, ...
Query	*LDA topic*	*BOA steak*
Keyphrases	latent dirichlet allocation, topic, topic models, topic modeling, probabilistic topic models, latent topics, topic discovery, generative model, mixture, text mining, topic distribution, plsi, ...	steak, stripsteak, boa steakhouse, steakhouse, ribeye, craftsteak, santa monica, medium rare, prime, vegas, entrees, potatoes, french fries, filet mignon, mashed potatoes, texas roadhouse, ...
Query	*SVM*	*deep dish pizza*
Keyphrases	support vector machines, svm classifier, multi class, training set, margin, knn, classification problems, kernel function, multi class svm, multi class support vector machine, support vector, ...	deep dish pizza, chicago, deep dish, amore taste of chicago, amore, pizza, oregano, chicago style, chicago style deep dish pizza, thin crust, windy city, slice, pan, oven, pepperoni, hot dog, ...

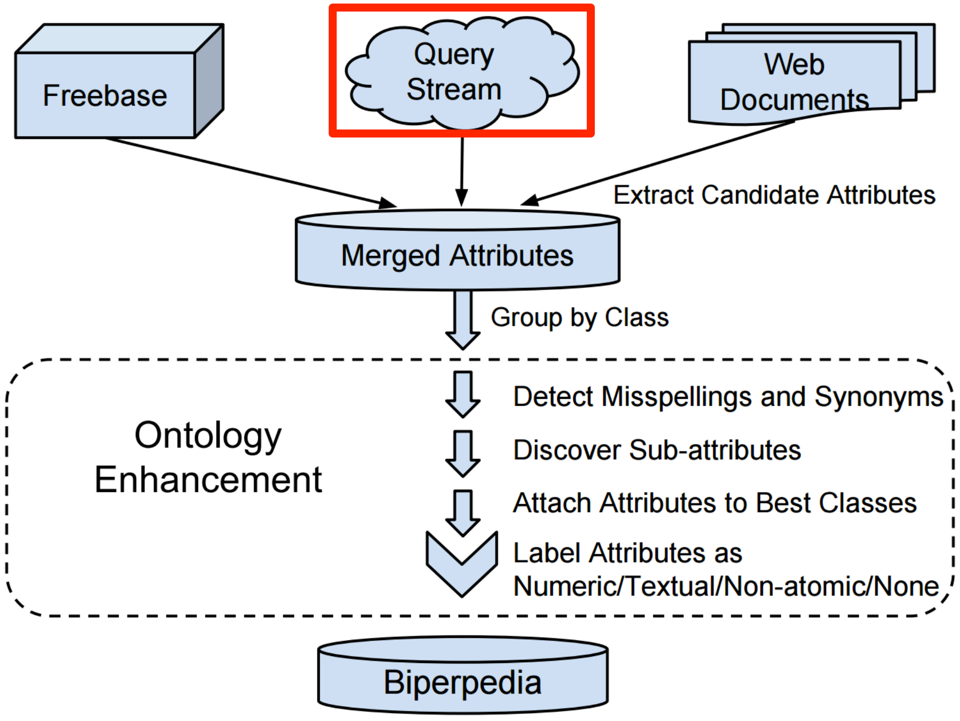
Query	*Mining Frequent Patterns without Candidate Generation*	*I am a huge fan of the All You Can Eat Chinese food buffet.*
Keyphrases	mining frequent patterns, candidate generation, frequent pattern mining, candidate, prune, fp growth, frequent pattern tree, apriori, subtrees, frequent patterns, candidate sets, ...	all you can eat, chinese food, buffet, chinese buffet, dim sum, orange chicken, chinese restaurant, asian food, asian buffet, crab legs, lunch buffet, fan, salad bar, all you can drink, ...
Query	*Text mining, also referred to as text data mining, roughly equivalent to text analytics, refers to the process of deriving high-quality information from text. High-quality information is typically derived through means such as statistical pattern learning.*	*It's the perfect steakhouse for both meat and fish lovers. My table guest was completely delirious about his Kobe Beef and my lobster was perfectly cooked. Good wine list, they have a lovely Sancerre! Professional staff, quick and smooth.*
Keyphrases	text analytics, text mining, patterns, text, textual data, topic, information, text documents, information extraction, machine learning, data mining, knowledge discovery, ...	kobe beef, fish lovers, steakhouse, sancerre, wine list, guests, perfectly cooked, lobster, staff, meat, fillet, fish, lover, seafood, ribeye, filet, sea bass, risotto, starter, scallops, steak, beef, ...
	Academia	Yelp

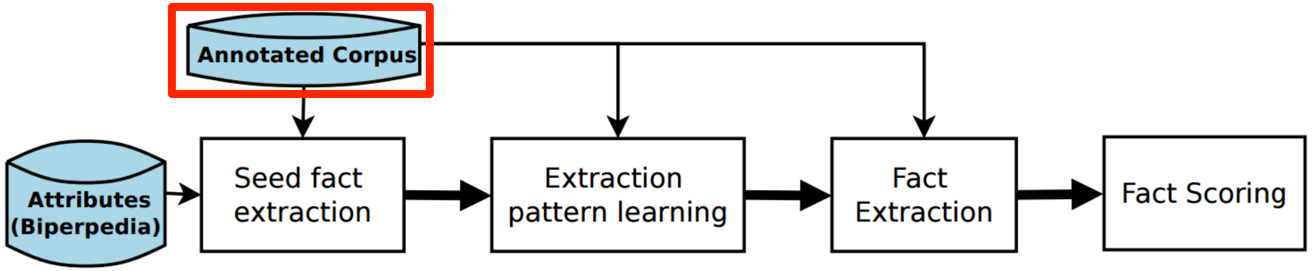
Trends and Research Problems

- Exploration of the Power of Entity Recognition and Typing
  - Mining Hidden Relationship Among Entities
- Mining Attributes and Values for Knowledge Network Construction
  - Mining the Universe of Attributes: The Google Approach
- Construction of Heterogeneous Information Networks from Entities, Attributes and Relationships
- Looking forward to the Future

Google's Approaches on Attribute Extraction

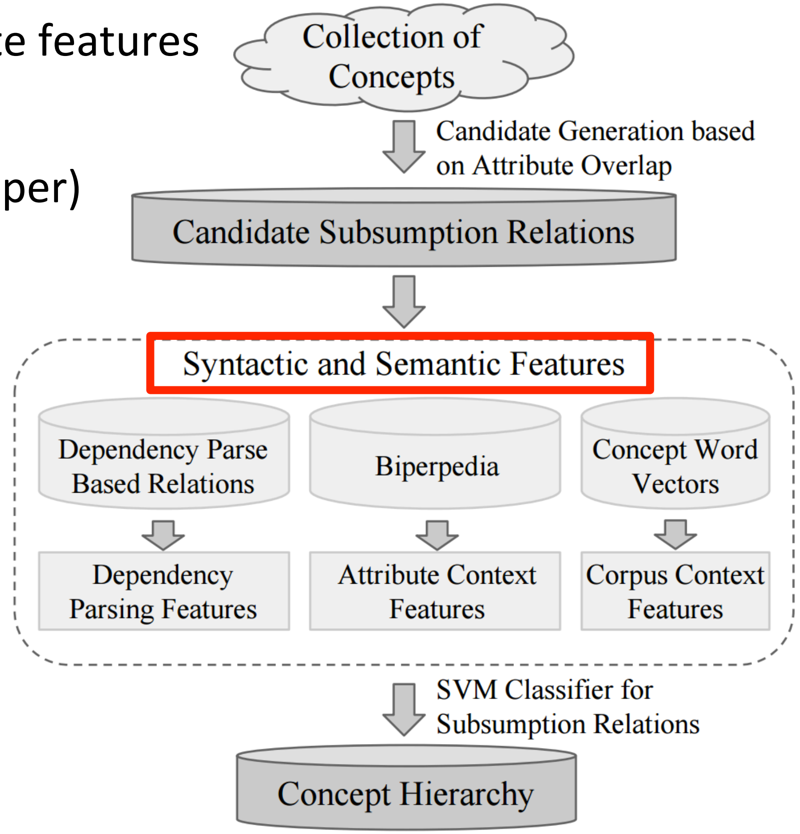
- Given Google's **query log**, web text and knowledge bases
  - "Obama wife name", "Obama daughter name", "Japan asian population", "Brazil female latino population", "Princeton economist"…
  - "Obama's wife, Michelle Obama, is a lawyer and writer.", "Princeton economist Paul Krugman was awarded the Nobel prize in 2008."…
  - Obama: *$Person, $President*; Japan, Brazil: *$Location, $Country*; Princeton: *$Location, $Organization, $University*…
- Biperpedia (VLDB'14): **Attribute Name Extraction** from query log
  - $Person: wife name, daughter name
  - $Country: asian population, female latino population
  - $University: economist
- ReNoun (EMNLP'14): **Fact Extraction for Noun Phrase Attribute**
  - (Obama, wife, Michelle Obama)
  - (Princeton, economist, Paul Krugman)

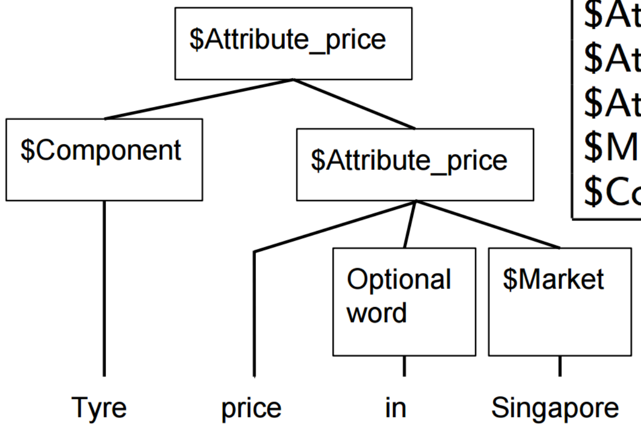




Google's Approaches on Attribute Extraction

- Latte (WebDB'15 Best Paper): **Concept (Type) Hierarchy Extraction** with attribute features
  - {country, address, zip code}: $University (sub) - $Location (super)
  - {online payment, non profit, tax return}: $University (sub) - $Organization (super)
  - {daughter name, wife name, age}: $President (sub) - $Person (super)



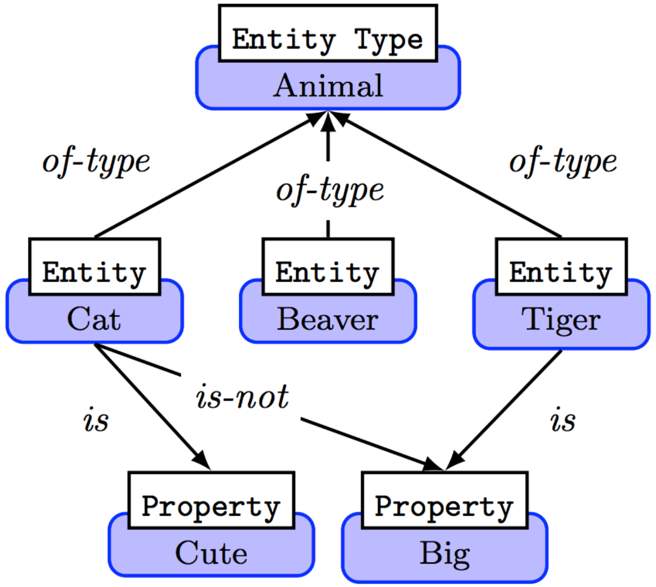



```
$Attribute_price ::= price
$Attribute_price ::= $Market $Attribute_price
$Attribute_price ::= $Attribute_price in $Market
$Attribute_price ::= $Component $Attribute_price
$Market          ::= Singapore | USA | Dubai | London | . . .
$Component       ::= battery | bumper | tyre | door | . . .
```

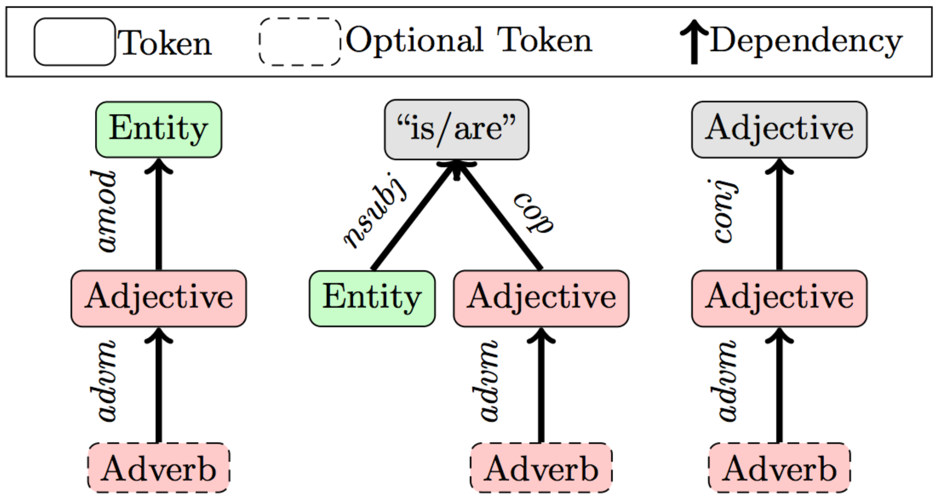
- ARI (WWW'16): **Attribute Name Structure Extraction** with rule-based grammar
  - Long-tail distribution of attribute names
  - $Person: $FamilyMember (name) - daughter, wife, mother, daughter name, wife name
  - $Country: ($Gender) ($Ethnicity) population - asian population, female latino population

Google's Approaches on Attribute Extraction

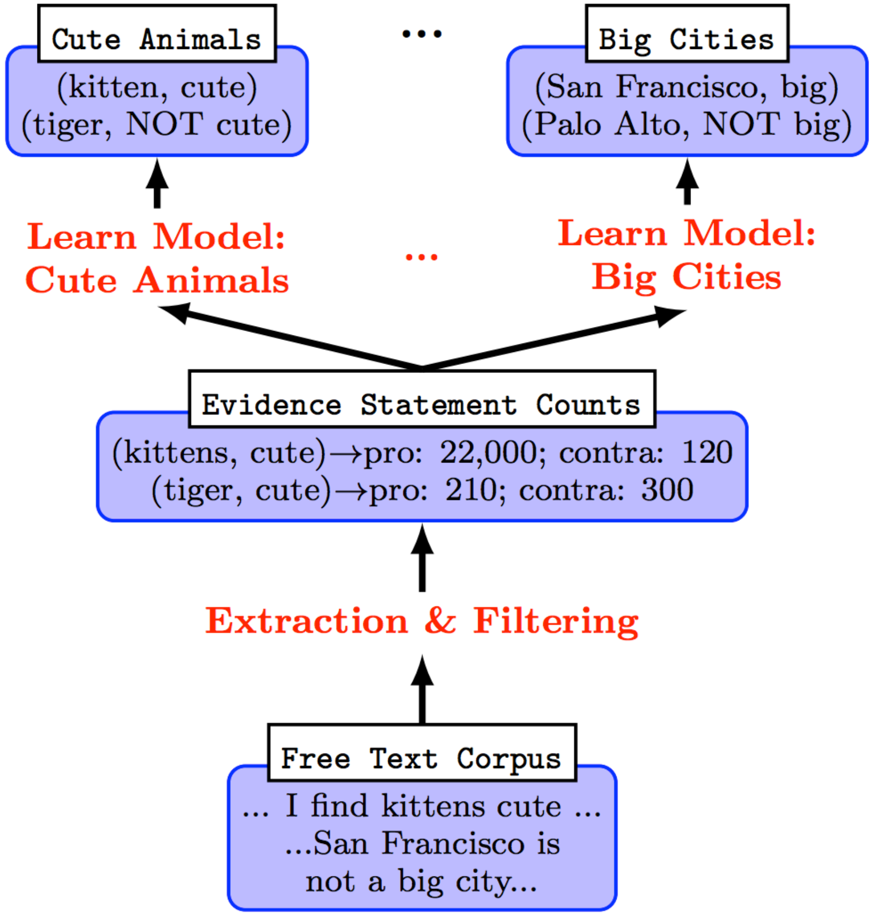
- Surveyor (SIGMOD'15): Learning **Subjective Properties**
  - Probabilistic model as Bayesian network: Learning model parameters
  - At least ρ extractions
  - <entity, property, +/->



Statement	Pattern	Entity	Property
Snakes are dangerous animals	Adjectival modifier	snake	dangerous
Chicago is very big	Adjectival complement	Chicago	very big
Soccer is a fast and exciting sport	Conjunction	soccer	exciting



(a) Adjectival modifier (b) Adjectival complement (c) Conjunction



Trends and Research Problems

- Exploration of the Power of Entity Recognition and Typing
  - Mining Hidden Relationship Among Entities
- Mining Attributes and Values for Knowledge Network Construction
  - Mining the Universe of Attributes: The Google Approach
- Construction of Heterogeneous Information Networks from Entities, Attributes and Relationships
- Looking forward to the Future

Construction of Heterogeneous Networks: Step I

- Scalable *phrase mining* methods for domain-specific corpora
  - Unsupervised approach: **TopMine**
  - → Weakly-supervised approach: **SegPhrase**
  - Easy to be **parallelized**

- A joint *entity recognition* and *relation phrase extraction* method
  - Corpus-level significance + POS tag patterns
  - Works on corpora of various **domains, genres**
  - Can be generalized to different **languages**

SegPhrase: https://github.com/shangjingbo1226/SegPhrase
TopMine: http://web.engr.illinois.edu/~elkishk2/code/ToPMine.zip

Construction of Heterogeneous Networks: Step I

- Distant Training: No need of human labels
  - e.g., Training using anchored phrases in general knowledge bases

- Multi-languages: 10 most popular languages on Wiki
  - Language-independent Tokenization using Lucene
  - Automatic language detection

Construction of Heterogeneous Networks: Step I Extensions of Entity Mention Extraction

- Integrating Part-of-Speech tagging within segmentation module
  - TreeTagger (a multi-lingual POS tagger) as pre-processing
  - Adjust transition probabilities based on the segmentation results of the domain-specific corpus

- Fully Parallel (both time and space efficient)
  - 1GB corpus, 10 threads (2.8GHz Xeon E5-2680)
  - Originally: 5-10GB memory, 1-2 hours
  - Goal: 2-3GB memory, 0.5 hours

Construction of Heterogeneous Networks: Step II

- A fully automatic method, **ClusType**, for entity recognition and typing of larger, domain-specific corpora
  - Leverages **minimal** linguistic/domain assumption
  - Requires **no** human supervision
  - **Efficient** learning compared to traditional NER methods
  - Can be **generalized** to other languages

ClusType: http://shanzhenren.github.io/ClusType

Construction of Heterogeneous Networks: Step II

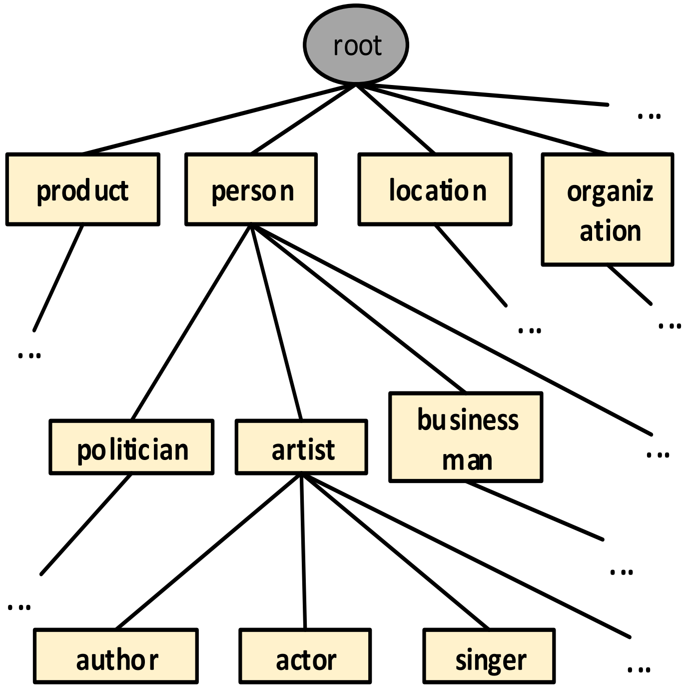
- Propose a novel *relation phrase-based framework* for distantly-supervised entity typing
  - Integrate relation phrase clustering with type propagation
  - Mutually enhance each other via solving a joint optimization problem
- Define the "Label Noise Reduction" task for distantly supervised entity typing
  - Denoise the automatically labeled training data
  - Yields more effective typing models

PLE: https://github.com/shanzhenren/PLE

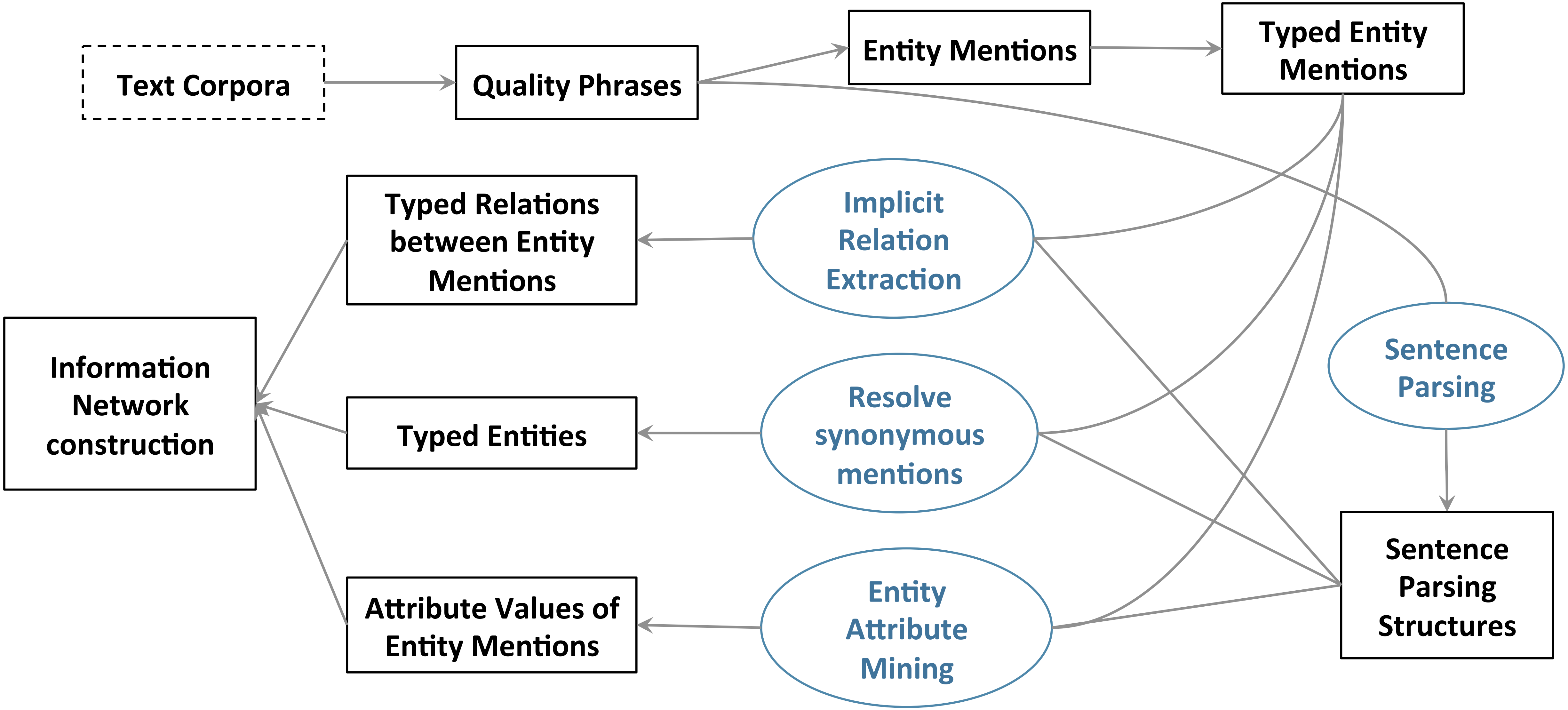
Extensions of Entity Typing

- The relation phrase-based framework can be used for multi-lingual entity typing

- Fine-grained entity typing
  - Current systems: coarse type set (usually < 10)
  - Fine-grained type set (over 100)
  - Relation phrases may be too coarse to distinguish **singer** with **actor**
    - fine-grained text features: dependency structures, …



Looking Forward: Research Problems



Software Packages Released

- Phrase Mining
  - SegPhrase: https://github.com/shangjingbo1226/SegPhrase
  - TopMine: http://web.engr.illinois.edu/~elkishk2/code/ToPMine.zip
- Entity Typing
  - ClusType: http://shanzhenren.github.io/ClusType
- Label Noise Reduction
  - PLE: https://github.com/shanzhenren/PLE
- Checking our research package dissemination portal
  - IlliMine http://illimine.cs.uiuc.edu/

References on Entity Recognition I

1. Ramshaw, Lance A., and Mitchell P. Marcus. "Text chunking using transformation-based learning." arXiv preprint cmp-lg/9505040 (1995)
2. Pla, Ferran, Antonio Molina, and Natividad Prieto. "Tagging and chunking with bigrams." ACL'2000
3. Kudo, Taku, and Yuji Matsumoto. "Chunking with support vector machines." Proc. of the second meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies. Association for Computational Linguistics, 2001
4. Koeling, Rob. "Chunking with maximum entropy models." Proc. workshop on Learning language in logic and the 4th conference on Computational natural language learning-Volume 7. ACL'2000
5. McCallum, Andrew, Dayne Freitag, and Fernando CN Pereira. "Maximum Entropy Markov Models for Information Extraction and Segmentation." ICML'2000
6. Zhou, GuoDong, and Jian Su. "Named entity recognition using an HMM-based chunk tagger." Proc. ACL'2002
7. Sarawagi, Sunita, and William W. Cohen. "Semi-markov conditional random fields for information extraction." NIPS'2004
8. Li, Qi, and Heng Ji. "Incremental joint extraction of entity mentions and relations." ACL'2014

References on Entity Recognition II

9. Ando, Rie Kubota, and Tong Zhang. "A high-performance semi-supervised learning method for text chunking." Proceedings ACL, 2005
10. Collins, Michael. "Ranking algorithms for named-entity extraction: Boosting and the voted perceptron." Proceedings of ACL'2002
11. Sha, Fei, and Fernando Pereira. "Shallow parsing with conditional random fields." Proceedings of the 2003 Conf. North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003.
12. Liu, Jialu, et al. "Mining Quality Phrases from Massive Text Corpora." Proceedings of the 2015 ACM SIGMOD 2015
13. El-Kishky, Ahmed, et al. "Scalable topical phrase mining from text corpora." VLDB'15
14. Ando, Rie Kubota, and Tong Zhang. "A high-performance semi-supervised learning method for text chunking." Proceedings of ACL 2005
15. Cohen, William W., and Sunita Sarawagi. "Exploiting dictionaries in named entity extraction: combining semi-markov extraction processes and data integration methods." Proceedings of KDD, 2004

References on Entity Recognition III

16. Ando, Rie Kubota, and Tong Zhang. "A high-performance semi-supervised learning method for text chunking." Proceedings ACL, 2005
17. Huang, Zhiheng, Wei Xu, and Kai Yu. "Bidirectional LSTM-CRF models for sequence tagging." arXiv preprint arXiv:1508.01991 (2015).
18. Hammerton, James. "Named entity recognition with long short-term memory." Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4. Association for Computational Linguistics, 2003.

References on Entity Typing I

1. *D. Nadeau and S. Sekine. A survey of named entity recognition and classification. Lingvisticae Investigationes, 30(1): 3–26, 2007.*
2. *A. Fader, S. Soderland, and O. Etzioni. Identifying relations for open information extraction. EMNLP, 2011.*
3. *J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. ACL, 2005.*
4. *L. Gal´arraga, G. Heitz, K. Murphy, and F. M. Suchanek. Canonicalizing open knowledge bases. CIKM, 2014.*
5. *S. Gupta and C. D. Manning. Improved pattern learning for bootstrapped entity extraction. CONLL, 2014.*
6. *X. He and P. Niyogi. Locality preserving projections. NIPS, 2004.*
7. *R. Huang and E. Riloff. Inducing domain-specific semantic class taggers from (almost) nothing. ACL, 2010.*
8. *T. Lin, O. Etzioni, et al. No noun phrase left behind: detecting and typing unlinkable entities. EMNLP, 2012.*
9. *X. Ling and D. S. Weld. Fine-grained entity recognition. AAAI, 2012.*
10. *B. Min, S. Shi, R. Grishman, and C.-Y. Lin. Ensemble semantics for large-scale unsupervised relation extraction. EMNLP, 2012.*
11. *N. Nakashole, T. Tylenda, and G. Weikum. Fine-grained semantic typing of emerging entities. ACL, 2013.*
12. *L. Ratinov and D. Roth. Design challenges and misconceptions in named entity recognition. ACL, 2009.*

References on Entity Typing II

13. *W. Shen, J. Wang, and J. Han. Entity linking with a knowledge base: Issues, techniques, and solutions. TKDE, (99):1–20, 2014.*
14. *P. P. Talukdar and F. Pereira. Experiments in graph-based semi-supervised learning methods for class-instance acquisition. ACL, 2010.*
15. Isozaki, Hideki, and Hideto Kazawa. "Efficient support vector classifiers for named entity recognition." COLING, 2002.
16. Chris Manning. Information Extraction and Named Entity Recognition. 2007. https://web.stanford.edu/class/cs124/lec/Information_Extraction_and_Named_Entity_Recognition.pdf
17. Turian, Joseph, Lev Ratinov, and Yoshua Bengio. "Word representations: a simple and general method for semi-supervised learning." ACL, 2010.
18. Mann, Gideon S., and Andrew McCallum. "Generalized expectation criteria for semi-supervised learning of conditional random fields." ACL-HLT, 2008
19. Jiao, Feng, Shaojun Wang, Chi-Hoon Lee, Russell Greiner, and Dale Schuurmans. "Semi-supervised conditional random fields for improved sequence segmentation and labeling." ACL, 2006.
20. R. Bunescu, M. Pasca. Using encyclopedic knowledge for named entity disambiguation, EACL, 2006.
21. S. Cucerzan. Large-scale named entity disambiguation based on wikipedia data, EMNLP-CoNLL, 2007.

References on Entity Typing III

22. Michael Thelen and Ellen Rilof. n A Bootstrapping Method for Learning Semantic Lexicons using Extraction Pattern Contexts, EMNLP 2002
23. Talukdar, Partha Pratim, Thorsten Brants, Mark Liberman, and Fernando Pereira. "A context pattern induction method for named entity extraction." CONLL, 2006.
24. Lin, Winston, Roman Yangarber, and Ralph Grishman. "Bootstrapped learning of semantic classes from positive and negative examples." IProceedings of ICML-2003 Workshop on The Continuum from Labeled to Unlabeled Data, 2003.
25. J. Hoffart, M. Yosef, I. Bordino, H. Furstenau, M. Pinkal, M. Spaniol, B. Taneva, S. Thater, G. Weikum. Robust disambiguation of named entities in text, EMNLP, 2011.
26. C. Wang, K. Chakrabarti, T. Cheng, S. Chaudhuri. Targeted Disambiguation of Ad-hoc, Homogeneous Sets of Named Entities, WWW, 2012.
27. Y. Li, C. Wang, F. Han, J. Han, D. Roth, and X. Yan. Mining Evidences for Named Entity Disambiguation, KDD, 2013.
28. Thelen, Michael, and Ellen Riloff. "A bootstrapping method for learning semantic lexicons using extraction pattern contexts." ACL, 2002.
29. Andrew Carlson, Justin Betteridge, Richard C. Wang, Estevam R. Hruschka, Jr., and Tom M. Mitchell. Coupled semi-supervised learning for information extraction. WSDM, 2010.
30. J. Davis, B. Kulis, P. Jain, S. Sra, and I. Dhillon. Information-theoretic metric learning. ICML, 2007.

References on Entity Typing IV

31. D. Downey, O. Etzioni, S. Soderland, and D. S. Weld. 2004. Learning Text Patterns for Web Information Extraction and Assessment. AAAI Workshop on Adaptive Text Extraction and Mining, 2004.
32. Roman Yangarber, Winston Lin, and Ralph Grishman. 2002. Unsupervised learning of generalized names. COLING, 2002.
33. P. Dhillon, P. Talukdar, and K. Crammer. Inference-driven metric learning for graph construction. Technical report, MS-CIS-10-18, University of Pennsylvania, 2010.
34. S. Daitch, J. Kelner, and D. Spielman. Fitting a graph to vector data. ICML, 2009.
35. Frank R. Kschischang, Brendan J. Frey, and Hans-Andrea Loeliger. Factor Graphs and the Sum-Product Algorithm. IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 47, NO. 2, FEBRUARY 2001
36. Belkin, Mikhail, Partha Niyogi, and Vikas Sindhwani. "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples." JMLR 7 (2006): 2399-2434.
37. Nakashole, Ndapandula, Gerhard Weikum, and Fabian Suchanek. "PATTY: a taxonomy of relational patterns with semantic types." EMNLP, 2012.
38. Xiang Ren, Ahmed El-Kishky, Chi Wang, Fangbo Tao, Clare R. Voss, Heng Ji, and Jiawei Han, ClusType: Effective Entity Recognition and Typing by Relation Phrase-Based Clustering, KDD 2015
39. Ahmed El-Kishky, Yanglei Song, Chi Wang, Clare R Voss, and Jiawei Han, Scalable Topical Phrase Mining from Text Corpora, VLDB 2015