Beyond TREC-QA

Ling573
NLP Systems and Applications
May 28, 2013

Roadmap

- Beyond TREC-style Question Answering
  - Watson and Jeopardy!
  - Web-scale relation extraction
    - Distant supervision

Watson & Jeopardy!™ vs QA

- QA vs Jeopardy!
- TREC QA systems on Jeopardy! task
- Design strategies
- Watson components
- DeepQA on TREC

TREC QA vs Jeopardy!

- Both:

TREC QA vs Jeopardy!

- Both:
  - Open domain ‘questions’; factoids
- TREC QA:

TREC QA vs Jeopardy!

- Both:
  - Open domain ‘questions’; factoids
- TREC QA:
  - ‘Small’ fixed doc set evidence, can access Web
  - No timing, no penalty for guessing wrong, no betting

TREC QA vs Jeopardy!

- Both:
  - Open domain ‘questions’; factoids
- TREC QA:
  - ‘Small’ fixed doc set evidence, can access Web
  - No timing, no penalty for guessing wrong, no betting
- Jeopardy!:
  - Timing, confidence key; betting
  - Board; Known question categories; Clues & puzzles
  - No live Web access, no fixed doc set

TREC QA Systems for Jeopardy!

- TREC QA somewhat similar to Jeopardy!

TREC QA Systems for Jeopardy!

- TREC QA somewhat similar to Jeopardy!
- Possible approach: extend existing QA systems
  - IBM's PIQUANT:
    - Closed document set QA, in top 3 at TREC: 30+%
  - CMU's OpenEphyra:
    - Web evidence-based system: 45% on TREC2002

TREC QA Systems for Jeopardy!

- TREC QA somewhat similar to Jeopardy!
- Possible approach: extend existing QA systems
  - IBM's PIQUANT:
    - Closed document set QA, in top 3 at TREC: 30+%
  - CMU's OpenEphyra:
    - Web evidence-based system: 45% on TREC2002
- Applied to 500 random Jeopardy questions
  - Both systems under 15% overall
    - PIQUANT ~45% when 'highly confident'

DeepQA Design Strategies

- Massive parallelism
  - Consider multiple paths and hypotheses

DeepQA Design Strategies

- Massive parallelism
  - Consider multiple paths and hypotheses
- Combine experts
  - Integrate diverse analysis components

DeepQA Design Strategies

- Massive parallelism
  - Consider multiple paths and hypotheses
- Combine experts
  - Integrate diverse analysis components
- Confidence estimation:
  - All components estimate confidence; learn to combine

DeepQA Design Strategies

- Massive parallelism
  - Consider multiple paths and hypotheses
- Combine experts
  - Integrate diverse analysis components
- Confidence estimation:
  - All components estimate confidence; learn to combine
- Integrate shallow/deep processing approaches

Watson Components: Content

- Content acquisition:
  - Corpora: encyclopedias, news articles, thesauri, etc
  - Automatic corpus expansion via web search
  - Knowledge bases: DBs, dbPedia, Yago, WordNet, etc

Watson Components: Question Analysis

- Uses
  - "Shallow & deep parsing, logical forms, semantic role labels, coreference, relations, named entities, etc"

Watson Components: Question Analysis

- Uses
  - "Shallow & deep parsing, logical forms, semantic role labels, coreference, relations, named entities, etc"
- Question analysis: question types, components
- Focus & LAT detection:
  - Finds lexical answer type and part of clue to replace with answer

Watson Components: Question Analysis

- Uses
  - "Shallow & deep parsing, logical forms, semantic role labels, coreference, relations, named entities, etc"
- Question analysis: question types, components
- Focus & LAT detection:
  - Finds lexical answer type and part of clue to replace with answer
- Relation detection: Syntactic or semantic rel's in Q
- Decomposition: Breaks up complex Qs to solve

Watson Components: Hypothesis Generation

- Applies question analysis results to support search in resources and selection of answer candidates

Watson Components: Hypothesis Generation

- Applies question analysis results to support search in resources and selection of answer candidates
- 'Primary search':
  - Recall-oriented search returning 250 candidates
  - Document- & passage-retrieval as well as KB search

Watson Components: Hypothesis Generation

- Applies question analysis results to support search in resources and selection of answer candidates
- 'Primary search':
  - Recall-oriented search returning 250 candidates
  - Document- & passage-retrieval as well as KB search
- Candidate answer generation:
  - Recall-oriented extracted of specific answer strings
    - E.g. NER-based extraction from passages

Watson Components: Filtering & Scoring

- Previous stages generated 100s of candidates
  - Need to filter and rank

Watson Components: Filtering & Scoring

- Previous stages generated 100s of candidates
  - Need to filter and rank
- Soft filtering:
  - Lower resource techniques reduce candidates to ~100

Watson Components: Filtering & Scoring

- Previous stages generated 100s of candidates
  - Need to filter and rank
- Soft filtering:
  - Lower resource techniques reduce candidates to ~100
- Hypothesis & Evidence scoring:
  - Find more evidence to support candidate
    - E.g. by passage retrieval augmenting query with candidate
  - Many scoring fns and features, including IDF-weighted overlap, sequence matching, logical form alignment, temporal and spatial reasoning, etc, etc..

Watson Components: Answer Merging and Ranking

- Merging:
  - Uses matching, normalization, and coreference to integrate different forms of same concept
    - e.g., ‘President Lincoln’ with ‘Honest Abe’

Watson Components: Answer Merging and Ranking

- Merging:
  - Uses matching, normalization, and coreference to integrate different forms of same concept
    - e.g., ‘President Lincoln’ with ‘Honest Abe’
- Ranking and Confidence estimation:
  - Trained on large sets of questions and answers
  - Metalearner built over intermediate domain learners
    - Models built for different question classes

Watson Components: Answer Merging and Ranking

- Merging:
  - Uses matching, normalization, and coreference to integrate different forms of same concept
    - e.g., ‘President Lincoln’ with ‘Honest Abe’
- Ranking and Confidence estimation:
  - Trained on large sets of questions and answers
  - Metalearner built over intermediate domain learners
    - Models built for different question classes
- Also tuned for speed, trained for strategy, betting

Retuning to TREC QA

- DeepQA system augmented with TREC-specific:

Retuning to TREC QA

- DeepQA system augmented with TREC-specific:
  - Question analysis and classification
  - Answer extraction

  - Used PIQUANT and OpenEphyra answer typing

Retuning to TREC QA

- DeepQA system augmented with TREC-specific:
  - Question analysis and classification
  - Answer extraction

  - Used PIQUANT and OpenEphyra answer typing

  - 2008: Unadapted: 35% -> Adapted: 60%
  - 2010: Unadapted: 51% -> Adapted: 67%

Summary

- Many components, analyses similar to TREC QA
  - Question analysis →Passage Retrieval → Answer extr.
    - May differ in detail, e.g. complex puzzle questions
- Some additional:
  - Intensive confidence scoring, strategizing, betting
- Some interesting assets:
  - Lots of QA training data, sparring matches
- Interesting approaches:
  - Parallel mixtures of experts; breadth, depth of NLP

Distant Supervision for Web-scale Relation Extraction

- Distant supervision for relation extraction without labeled data
  - Mintz et al, 2009

Distant Supervision for Web-scale Relation Extraction

- Distant supervision for relation extraction without labeled data
  - Mintz et al, 2009
- Approach:
  - Exploit large-scale:
    - Relation database of relation instance examples
    - Unstructured text corpus with entity occurrences
  - To learn new relation patterns for extraction

Motivation

- Goal: Large-scale mining of relations from text

Motivation

- Goal: Large-scale mining of relations from text
  - Example: Knowledge Base Population task
    - Fill in missing relations in a database from text
    - Born_in, Film_director, band_origin
- Challenges:

Motivation

- Goal: Large-scale mining of relations from text
  - Example: Knowledge Base Population task
    - Fill in missing relations in a database from text
    - Born_in, Film_director, band_origin
- Challenges:
  - Many, many relations
  - Many, many ways to express relations

Motivation

- Goal: Large-scale mining of relations from text
  - Example: Knowledge Base Population task
    - Fill in missing relations in a database from text
    - Born_in, Film_director, band_origin
- Challenges:
  - Many, many relations
  - Many, many ways to express relations
  - How can we find them?

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues:

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues: Few relations, examples, documents

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues: Few relations, examples, documents
    - Expensive labeling, domain specificity
- Unsupervised clustering:
  - Issues:

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues: Few relations, examples, documents
    - Expensive labeling, domain specificity
- Unsupervised clustering:
  - Issues: May not extract desired relations
- Bootstrapping: e.g. Ravichandran & Hovy
  - Use small number of seed examples to learn patterns
  - Issues

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues: Few relations, examples, documents
    - Expensive labeling, domain specificity
- Unsupervised clustering:
  - Issues: May not extract desired relations
- Bootstrapping: e.g. Ravichandran & Hovy
  - Use small number of seed examples to learn patterns
  - Issues: Lexical/POS patterns; local patterns

Prior Approaches

- Supervised learning:
  - E.g. ACE: 16.7K relation instances; 30 total relations
  - Issues: Few relations, examples, documents
    - Expensive labeling, domain specificity
- Unsupervised clustering:
  - Issues: May not extract desired relations
- Bootstrapping: e.g. Ravichandran & Hovy
  - Use small number of seed examples to learn patterns
  - Issues: Lexical/POS patterns; local patterns
    - Can't handle long-distance

New Strategy

- Distant Supervision:
  - Supervision (examples) via large semantic database

New Strategy

- Distant Supervision:
  - Supervision (examples) via large semantic database
- Key intuition:
  - If a sentence has two entities from a Freebase relation,
  - they should express that relation in the sentence

New Strategy

- Distant Supervision:
  - Supervision (examples) via large semantic database
- Key intuition:
  - If a sentence has two entities from a Freebase relation,
  - they should express that relation in the sentence
- Secondary intuition:
  - Many witness sentences expressing relation
  - Can jointly contribute to features in relation classifier
- Advantages:

New Strategy

- Distant Supervision:
  - Supervision (examples) via large semantic database
- Key intuition:
  - If a sentence has two entities from a Freebase relation,
  - they should express that relation in the sentence
- Secondary intuition:
  - Many witness sentences expressing relation
  - Can jointly contribute to features in relation classifier
- Advantages: Avoids overfitting, uses named relations

Freebase

- Freely available DB of structured semantic data
  - Compiled from online sources
    - E.g. Wikipedia infoboxes, NNDB, SEC, manual entry

Freebase

- Freely available DB of structured semantic data
  - Compiled from online sources
    - E.g. Wikipedia infoboxes, NNDB, SEC, manual entry
- Unit: Relation
  - Binary relations between ordered entities
    - E.g. person-nationality: <John Steinbeck, US>

Freebase

- Freely available DB of structured semantic data
  - Compiled from online sources
    - E.g. Wikipedia infoboxes, NNDB, SEC, manual entry
- Unit: Relation
  - Binary relations between ordered entities
    - E.g. person-nationality: <John Steinbeck, US>
- Full DB: 116M instances, 7.3K rels, 9M entities

Freebase

- Freely available DB of structured semantic data
  - Compiled from online sources
    - E.g. Wikipedia infoboxes, NNDB, SEC, manual entry
- Unit: Relation
  - Binary relations between ordered entities
    - E.g. person-nationality: <John Steinbeck, US>
- Full DB: 116M instances, 7.3K rels, 9M entities
- Largest relations: 1.8M inst., 102 rels, 940K entities

Relation name	Size	Example
/people/person/nationality	281,107	John Dugard, South Africa
/location/location/contains	253,223	Belgium, Nijlen
/people/person/profession	208,888	Dusa McDuff, Mathematician
/people/person/place_of_birth	105,799	Edwin Hubble, Marshfield
/dining/restaurant/cuisine	86,213	MacAyo's Mexican Kitchen, Mexican
/business/business_chain/location	66,529	Apple Inc., Apple Inc., South Park, NC
/biology/organism_classification_rank	42,806	Scorpaeniformes, Order
/film/film/genre	40,658	Where the Sidewalk Ends, Film noir
/film/film/language	31,103	Enter the Phoenix, Cantonese
/biology/organism_higher_classification	30,052	Calopteryx, Calopterygidae
/film/film/country	27,217	Turtle Diary, United States
/film/writer/film	23,856	Irving Shulman, Rebel Without a Cause
/film/director/film	23,539	Michael Mann, Collateral
/film/producer/film	22,079	Diane Eskenazi, Aladdin

Basic Method

- Training:
  - Identify entities in sentences, using NER

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector
  - Combine features by rel'n across sent. in multiclass LR
- Testing:

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector
  - Combine features by rel'n across sent. in multiclass LR
- Testing:
  - Identify entities with NER
  - If find two entities in sentence together

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector
  - Combine features by rel'n across sent. in multiclass LR
- Testing:
  - Identify entities with NER
  - If find two entities in sentence together
    - Add features to vector

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector
  - Combine features by rel’n across sent. in multiclass LR
- Testing:
  - Identify entities with NER
  - If find two entities in sentence together
    - Add features to vector
  - Predict based on features from all sents
    - Pair appears 10x, 3 features

Basic Method

- Training:
  - Identify entities in sentences, using NER
  - If find two entities participating in Freebase relation,
    - Extract features, add to relation vector
  - Combine features by rel'n across sent. in multiclass LR
- Testing:
  - Identify entities with NER
  - If find two entities in sentence together
    - Add features to vector
  - Predict based on features from all sents
    - Pair appears 10x, 3 features ➔ 30 features

Examples

- Exploiting strong info:

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’
  - Testing: ‘Vienna, the capital of Austria’
- Combining evidence: <Spielberg, Saving Private Ryan>

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’
  - Testing: ‘Vienna, the capital of Austria’
- Combining evidence: <Spielberg, Saving Private Ryan>
    - [Spielberg]’s film, [Saving Private Ryan] is loosely based...

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’
  - Testing: ‘Vienna, the capital of Austria’
- Combining evidence: <Spielberg, Saving Private Ryan>
    - [Spielberg]’s film, [Saving Private Ryan] is loosely based…
      - Director? Writer? Producer?
    - Award winning [Saving Private Ryan] , directed by [Spielberg]

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’
  - Testing: ‘Vienna, the capital of Austria’
- Combining evidence: <Spielberg, Saving Private Ryan>
    - [Spielberg]’s film, [Saving Private Ryan] is loosely based…
      - Director? Writer? Producer?
    - Award winning [Saving Private Ryan] , directed by [Spielberg]
      - CEO? (Film-)Director?
  - If see both

Examples

- Exploiting strong info: Location-contains:
  - Freebase: <Virginia,Richmond>,<France,Nantes>
  - Training sentences: ‘Richmond, the capital of Virginia’
    - ‘Edict of Nantes helped the Protestants of France’
  - Testing: ‘Vienna, the capital of Austria’
- Combining evidence: <Spielberg, Saving Private Ryan>
    - [Spielberg]’s film, [Saving Private Ryan] is loosely based…
      - Director? Writer? Producer?
    - Award winning [Saving Private Ryan] , directed by [Spielberg]
      - CEO? (Film-)Director?
  - If see both ➔ Film-director

Feature Extraction

- Lexical features: Conjuncts of

Feature Extraction

- Lexical features: Conjuncts of
  - Astronomer Edwin Hubble was born in Marshfield,MO

Feature Extraction

- Lexical features: Conjuncts of
  - Sequence of words between entities
  - POS tags of sequence between entities
  - Flag for entity order
  - k words+POS before 1^{st} entity
  - k words+POS after 2^{nd} entity

  - Astronomer Edwin Hubble was born in Marshfield,MO

Feature Extraction

- Lexical features: Conjuncts of
  - Sequence of words between entities
  - POS tags of sequence between entities
  - Flag for entity order
  - k words+POS before 1st entity
  - k words+POS after 2nd entity

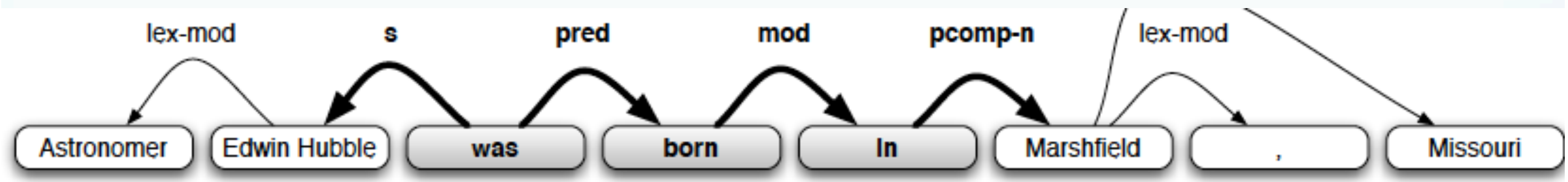
  - Astronomer Edwin Hubble was born in Marshfield,MO

Feature type	Left window	NE1	Middle	NE2	Right window
Lexical	[]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[]
Lexical	[Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[,]
Lexical	[#PAD#, Astronomer]	PER	[was/VERB born/VERB in/CLOSED]	LOC	[, Missouri]

Feature Extraction II

- Syntactic features: Conjuncts of:

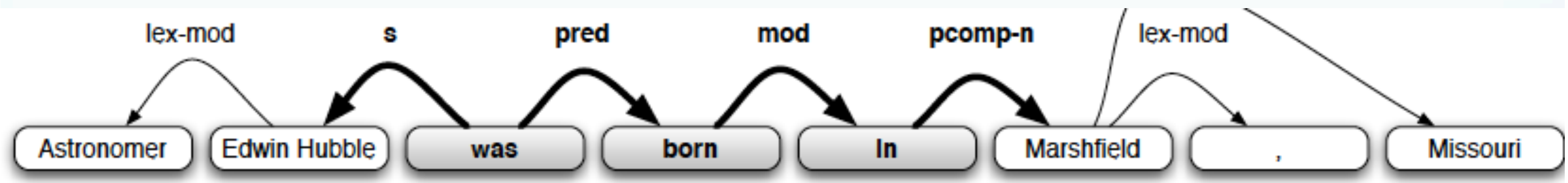
Feature Extraction II



Syntactic	[]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[]
Syntactic	[]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	$[\Uparrow_s \text{ was } \Downarrow_{pred} \text{ born } \Downarrow_{mod} \text{ in } \Downarrow_{pcomp-n}]$	LOC	[$\Downarrow_{lex-mod}$,]

Feature Extraction II

- Syntactic features: Conjuncts of:
  - Dependency path between entities, parsed by Minipar
    - Chunks, dependencies, and directions
  - Window node not on dependency path



Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[]
Syntactic	[]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Edwin Hubble $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]
Syntactic	[Astronomer $\Downarrow_{lex-mod}$]	PER	[$\Uparrow_s$ was $\Downarrow_{pred}$ born $\Downarrow_{mod}$ in $\Downarrow_{pcomp-n}$]	LOC	[$\Downarrow_{lex-mod}$,]

High Weight Features

author_editor	LEX $\curvearrowleft$	ORG	s novel	PER
	SYN	PER	$\Uparrow_{nn}$ series $\Downarrow_{gen}$	PER
founders	LEX	ORG	co - founder	PER
	SYN	ORG	$\Uparrow_{nn}$ owner $\Downarrow_{person}$	PER
place_founded	LEX $\curvearrowleft$	ORG	- based	LOC
	SYN	ORG	$\Uparrow_{s}$ founded $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC

High Weight Features

- Features highly specific: Problem?

'author_editor	LEX$\curvearrowleft$	ORG	s novel	PER
	SYN	PER	$\Uparrow_{nn}$ series $\Downarrow_{gen}$	PER
founders	LEX	ORG	co - founder	PER
	SYN	ORG	$\Uparrow_{nn}$ owner $\Downarrow_{person}$	PER
place_founded	LEX$\curvearrowleft$	ORG	- based	LOC
	SYN	ORG	$\Uparrow_{s}$ founded $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC

High Weight Features

- Features highly specific: Problem?
  - Not really, attested in large text corpus

author_editor	LEX $\curvearrowleft$	ORG	s novel	PER
	SYN	PER	$\Uparrow_{nn}$ series $\Downarrow_{gen}$	PER
founders	LEX	ORG	co - founder	PER
	SYN	ORG	$\Uparrow_{nn}$ owner $\Downarrow_{person}$	PER
place_founded	LEX $\curvearrowleft$	ORG	- based	LOC
	SYN	ORG	$\Uparrow_{s}$ founded $\Downarrow_{mod}$ in $\Downarrow_{pcn}$	LOC

Evaluation Paradigm

Evaluation Paradigm

- Train on subset of data, test on held-out portion

Evaluation Paradigm

- Train on subset of data, test on held-out portion
- Train on all relations, using part of corpus
  - Test on new relations extracted from Wikipedia text
  - How evaluate newly extracted relations?

Evaluation Paradigm

- Train on subset of data, test on held-out portion
- Train on all relations, using part of corpus
  - Test on new relations extracted from Wikipedia text

  - How evaluate newly extracted relations?
    - Send to human assessors
    - Issue:

Evaluation Paradigm

- Train on subset of data, test on held-out portion
- Train on all relations, using part of corpus
  - Test on new relations extracted from Wikipedia text
  - How evaluate newly extracted relations?
    - Send to human assessors
    - Issue: 100s or 1000s of each type of relation

Evaluation Paradigm

- Train on subset of data, test on held-out portion
- Train on all relations, using part of corpus
  - Test on new relations extracted from Wikipedia text

  - How evaluate newly extracted relations?
    - Send to human assessors
    - Issue: 100s or 1000s of each type of relation
      - Crowdsource: Send to Amazon Mechanical Turk

Results

- Overall: on held-out set
    - Best precision combines lexical, syntactic
    - Significant skew in identified relations
        - @100,000: 60% *location-contains*, 13% *person-birthplace*

Results

- Overall: on held-out set
  - Best precision combines lexical, syntactic
  - Significant skew in identified relations
    - @100,000: 60% *location-contains,* 13% *person-birthplace*

  - Syntactic features helpful in ambiguous, long-distance
  - E.g.
    - Back Street is a 1932 film made by Universal Pictures, directed by John M. Stahl,...

Human-Scored Results

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

Human-Scored Results

- @ Recall 100: Combined lexical, syntactic best

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

Human-Scored Results

- @ Recall 100: Combined lexical, syntactic best
  - @1000: mixed

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

Distant Supervision

- Uses large databased as source of true relations
- Exploits co-occurring entities in large text collection
- Scale of corpus, richer syntactic features
  - Overcome limitations of earlier bootstrap approaches
- Yields reasonably good precision
  - Drops somewhat with recall
  - Skewed coverage of categories