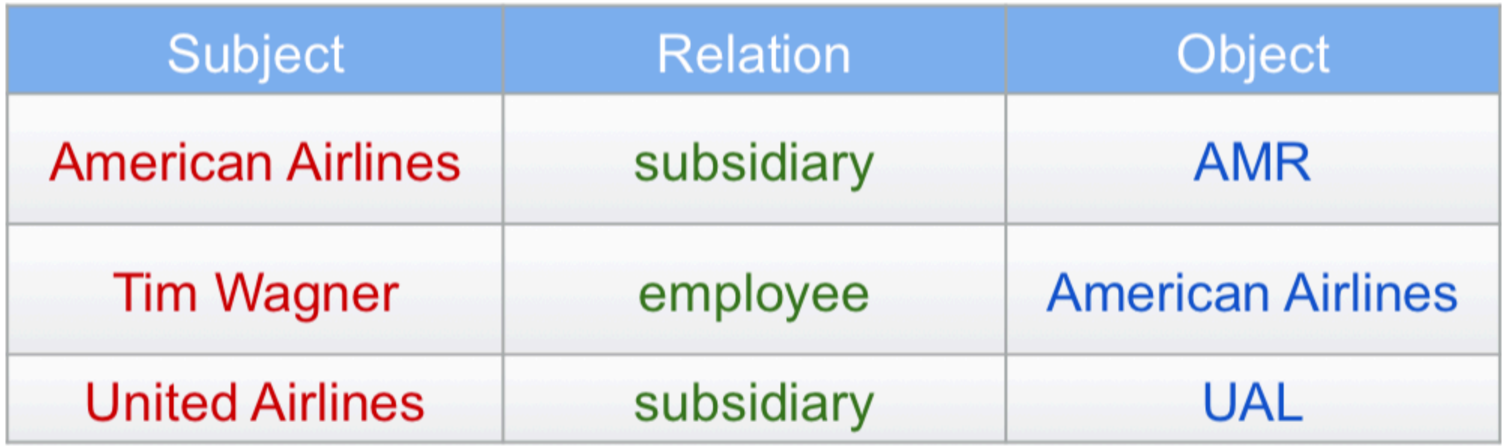
# Relation Extraction

# Relation Extraction

- Find and identify semantic relations among entities
- Binary relations in most cases
- Structure to the entities that constitute the relation
- Used to populate a relational database.

# Example of a Relation

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



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

**Taken from textbook**

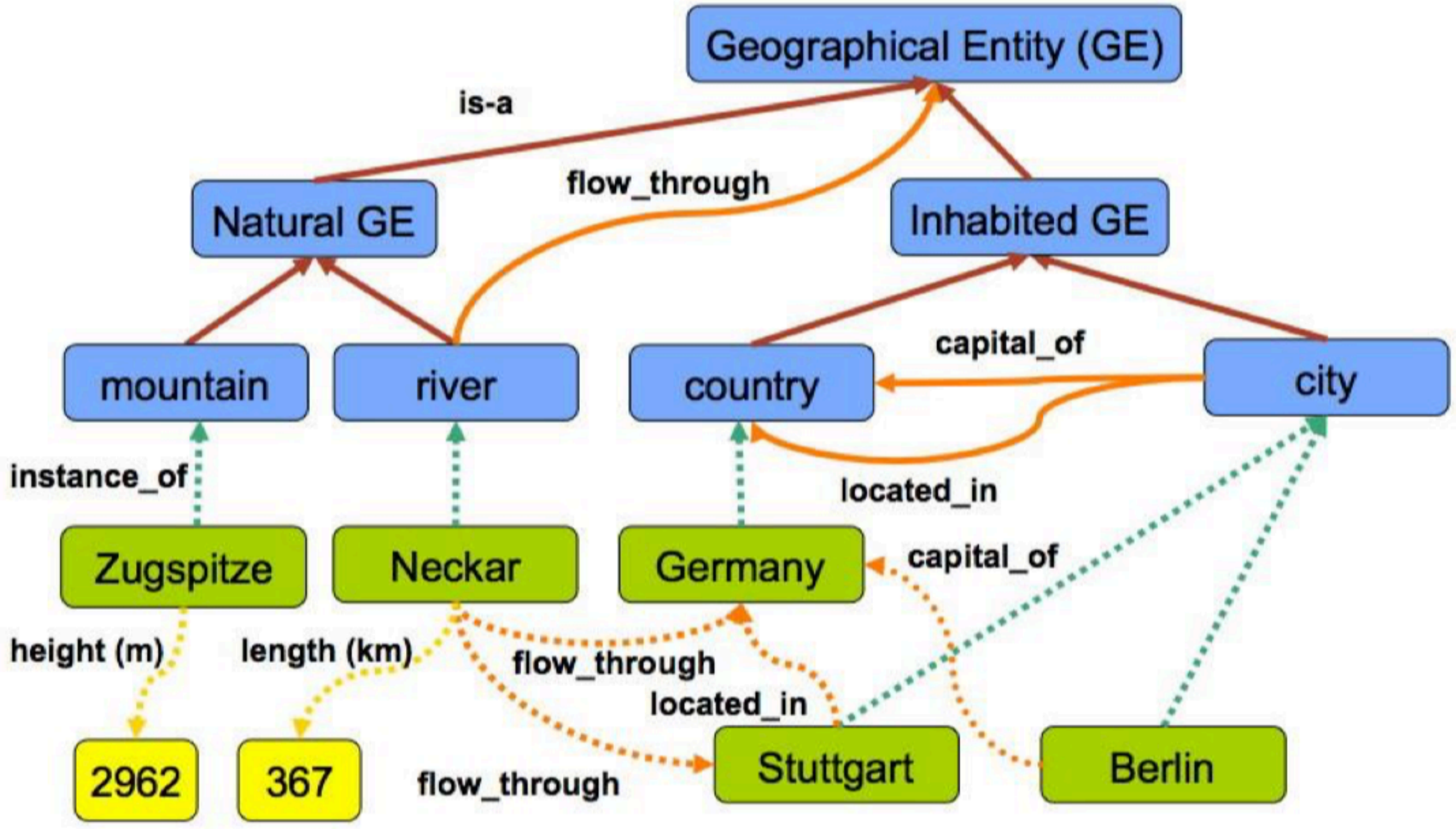
# Other possible examples

- ParentOf(Ned Stark, Jon Snow)
- LocatedIn(Salt Lake City, Utah)
- OccursOn(Christmas, Dec 25)
- Birthyear(Mozart, 1756)

# Why does Relation Extraction matter?

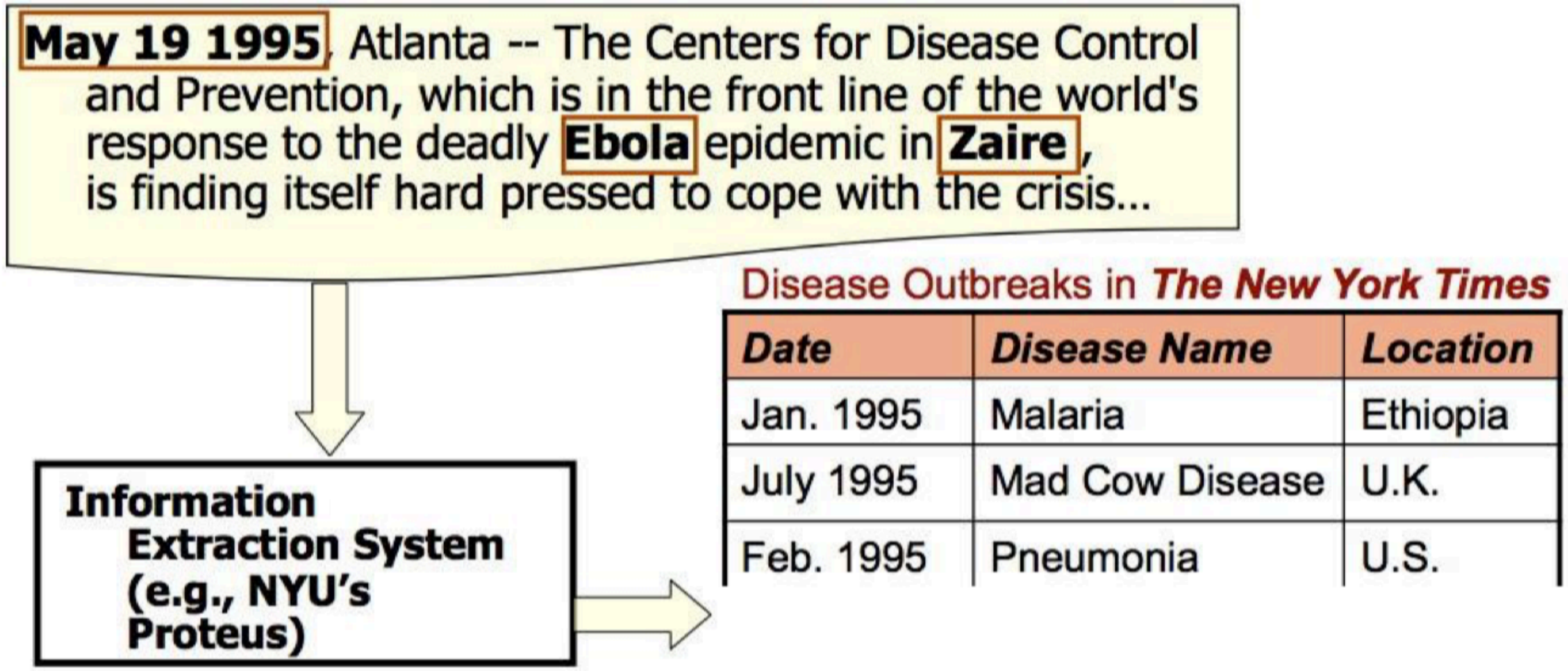
- Create organizational structures and profiles
- Automatically learn to add new entries in databases.
- Can be used for question answering.
- Allows for logical reasoning.

# Example of geographical relations



slide adapted from Paul Buitelaar

# Another Example



**May 19 1995**, Atlanta -- The Centers for Disease Control and Prevention, which is in the front line of the world's response to the deadly **Ebola** epidemic in **Zaire**, is finding itself hard pressed to cope with the crisis...

**Information Extraction System (e.g., NYU's Proteus)**

Disease Outbreaks in ***The New York Times***

| ***Date*** | ***Disease Name*** | ***Location*** |
|---|---|---|
| Jan. 1995 | Malaria | Ethiopia |
| July 1995 | Mad Cow Disease | U.K. |
| Feb. 1995 | Pneumonia | U.S. |

slide adapted from Eugene Agichtein

# Types of relations

- The ACE (Automatic Content Extraction) program conducts community-wide performance evaluations of IE systems.

- Tasks defined and focused on IE: entities, relations, events, within and across documents.

# Types of relations

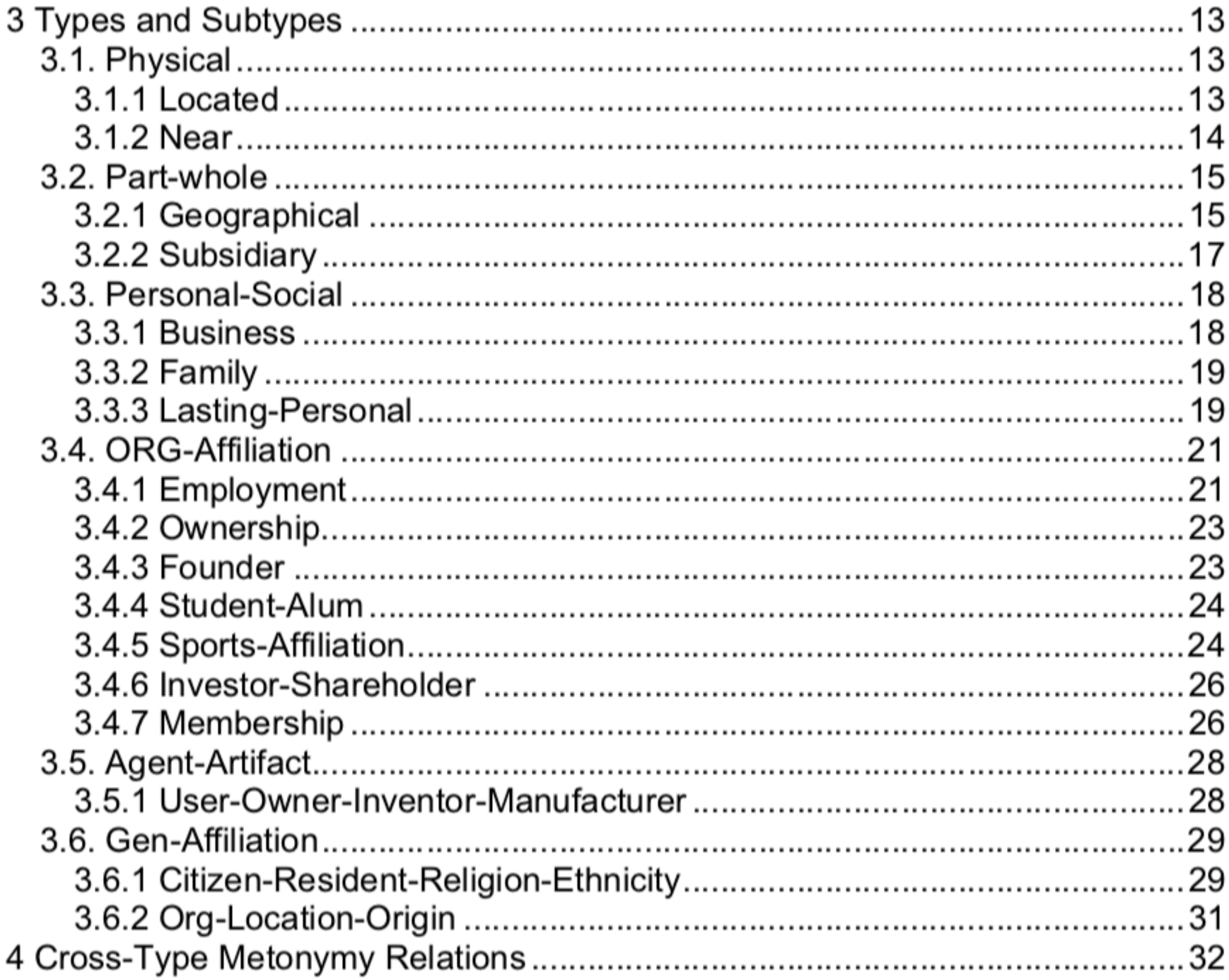
- Mentions: instances irrespective of types
- Geopolitical entities (GPE): geographic area but can act as different entity types.

**the riots in Miami → Location**

**Miami voted against … → Person**

**Miami imposed restrictions … → Organization**

# ACE 2008 Types



# Approaches for Relation Extraction

1. **Handwritten Rules**

2. Bootstrapping approaches

3. Distant Supervision

4. Unsupervised approaches

# Approaches for Relation Extraction

1. **Handwritten Rules**
2. Bootstrapping approaches
3. Supervised approaches
4. Distant Supervision

# Handwritten Rules for Relation Extraction

- Enforcing entity type matching and context matching.
- Example: LocatedIn(<Organization>,<Location>)
- Rules: XYZ based in ABC, ABC headquarters in XYZ, ABC offices in XYZ.

# Handwritten Rules for Relation Extraction

- Example 2: StudiesIn(<Person>,<University>)

- Rules: XYZ attends ABC, XYZ studies in ABC, **XYZ goes to ABC**

- Note: beware of generalizing. Your patterns may start extracting things apart from the relation you are trying to extract.

# Exercise

- Form a group of 2-3 people.
- Come up with 3 different handwritten rules to learn the following relationships. Assume that NER has been performed for you.
  - EmployeeOf(<Person>,<Organization>)
  - MemberOf(<Person>,<SportsTeam>)
  - AuthorOf(<Person>,<BookTitle>)

# Approaches for Relation Extraction

1. Handwritten Rules

2. **Bootstrapping approaches**

3. Supervised approaches

4. Distant Supervision

# Bootstrapping Approaches

- “Seed” instances of relations
- Lots of unannotated text
- Semi-supervised approach

# Bootstrapping example

- Target relation: *burial place*
- Seed tuple: [*Mark Twain*, *Elmira*]
- Grep/Google for “Mark Twain” and “Elmira” “Mark Twain is buried in Elmira, NY.”

→ X is buried in Y

“The grave of Mark Twain is in Elmira”

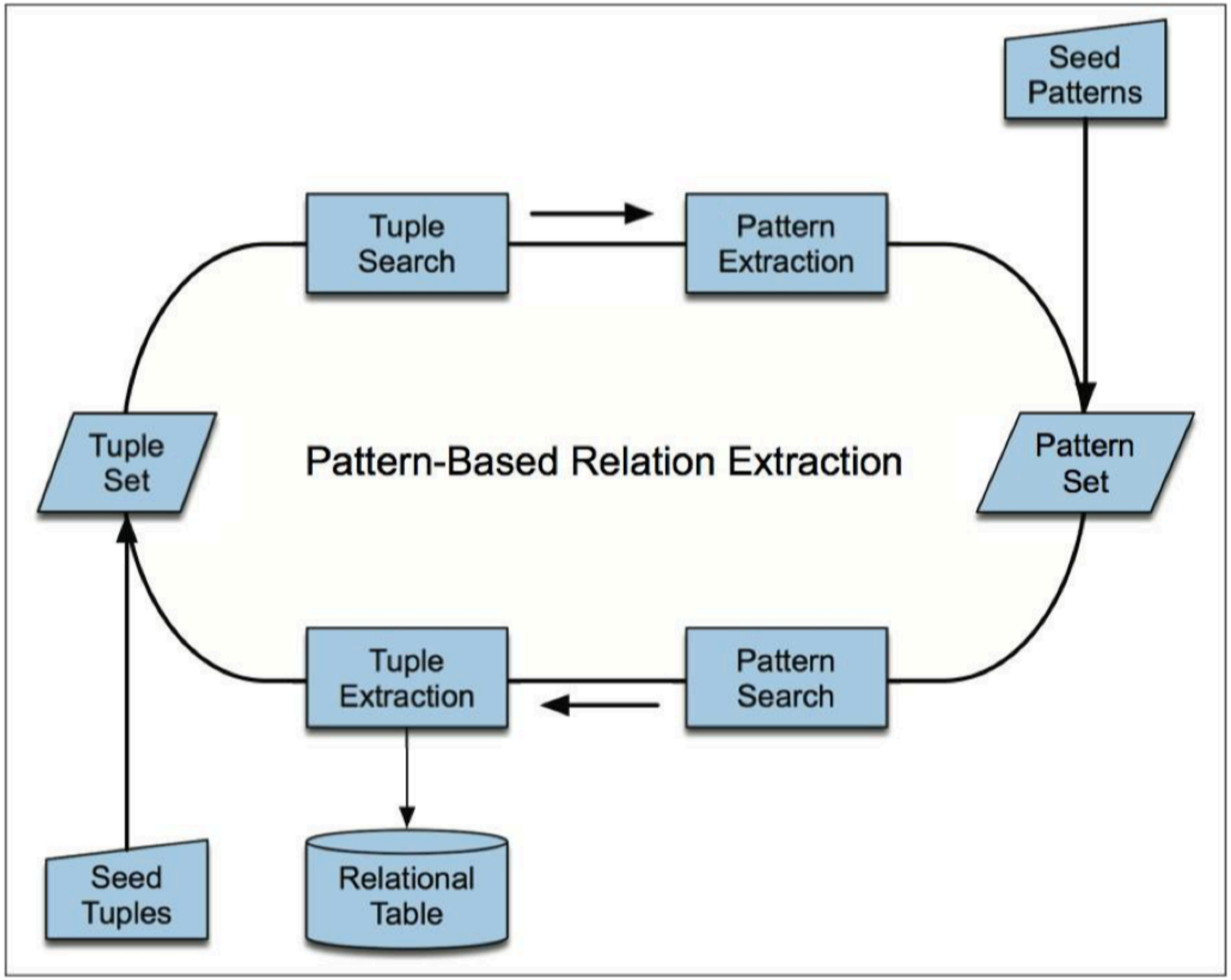
→ The grave of X is in Y

“Elmira is Mark Twain’s final resting place”

→ Y is X’s final resting place

- Use those patterns to search for new tuples

# Bootstrapping process

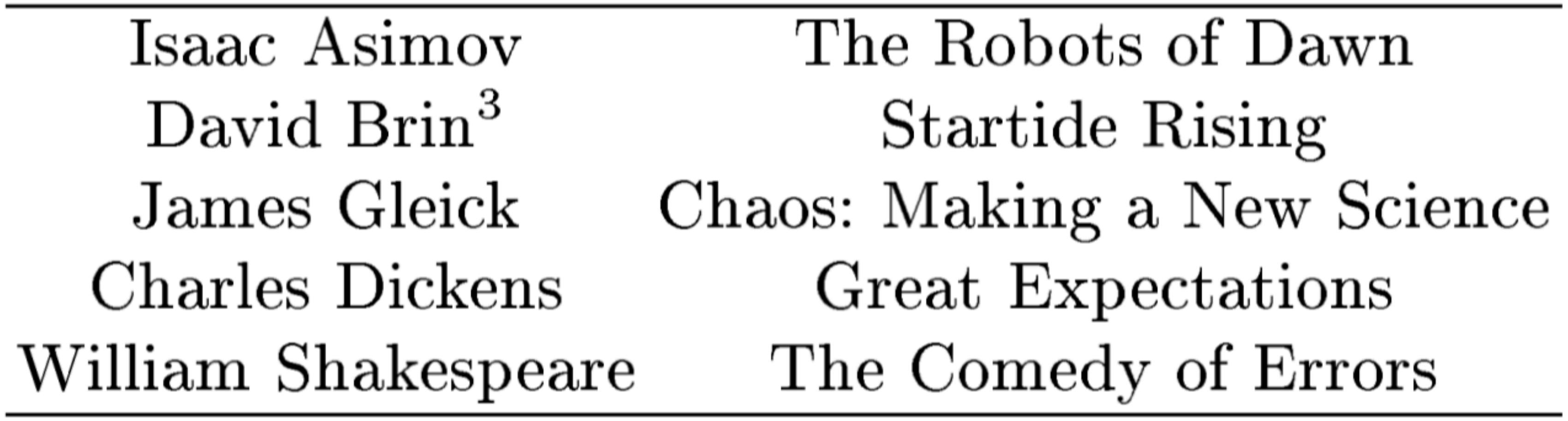


slide adapted from Jim Martin

# DIPRE (Brin 1998)

- Dual Iterative Pattern Relation Expansion
- Extract (author, book) pairs
- 5 Step Approach:
  1. Start with a small sample of target relation
  2. Find occurrences in the data
  3. ***Generate patterns based on the occurrences found***
  4. Search database for new tuples.
  5. If enough tuples found, return. Else repeat.

# DIPRE

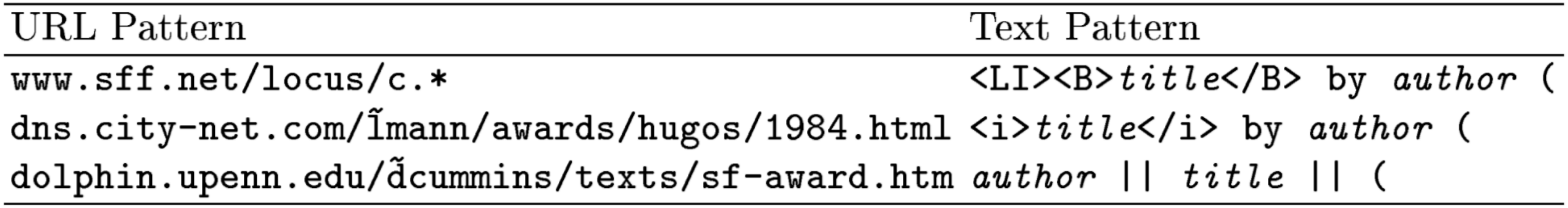


| | |
|---|---|
| Isaac Asimov | The Robots of Dawn |
| David Brin[3] | Startide Rising |
| James Gleick | Chaos: Making a New Science |
| Charles Dickens | Great Expectations |
| William Shakespeare | The Comedy of Errors |

**Fig. 1.** Initial sample of books.

# DIPRE

- Patterns generated by the seed occurrences:
- 1999 occurrences and 3 patterns
- Patterns contain left, middle and right.

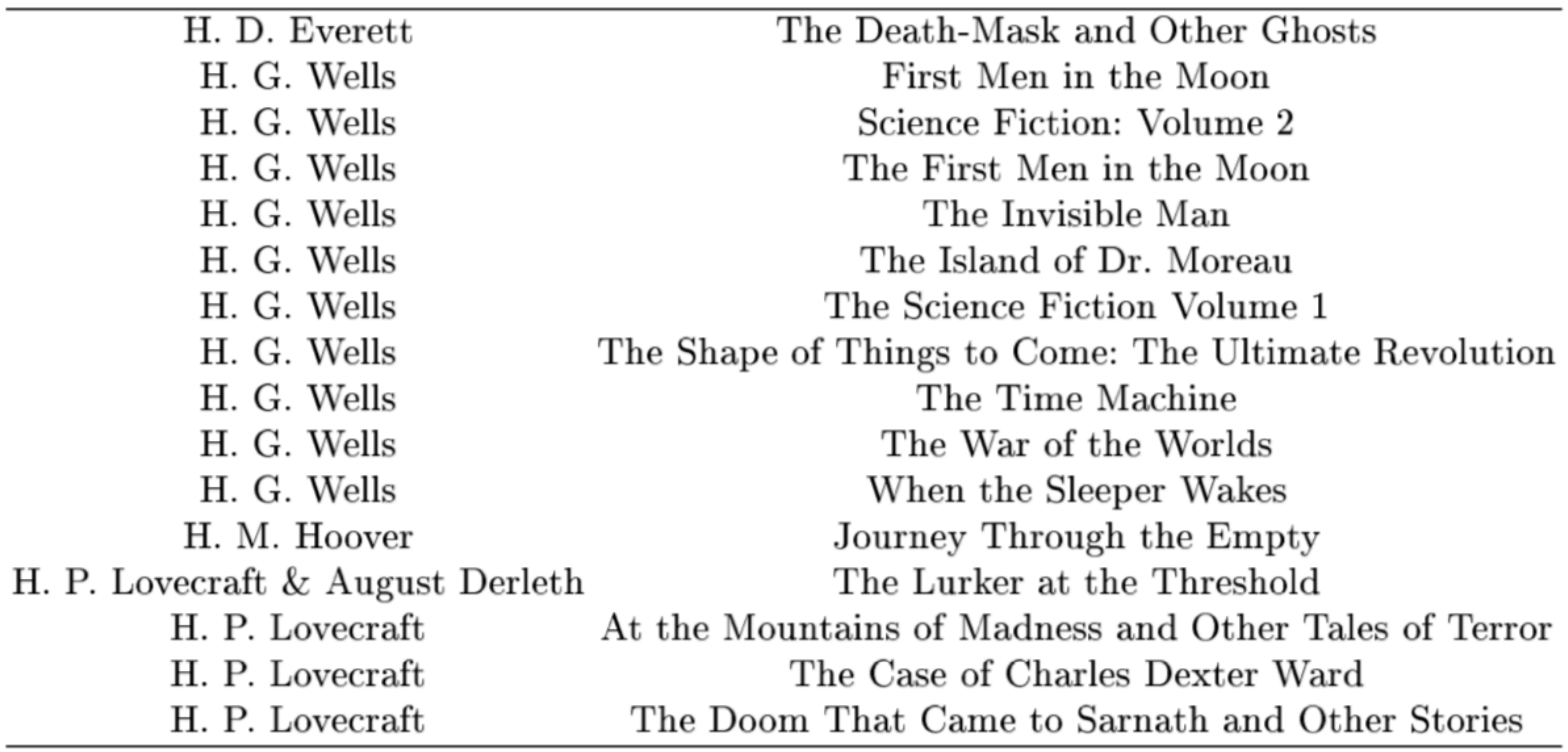


| URL Pattern | Text Pattern |
|---|---|
| www.sff.net/locus/c.* | <LI><B>*title*</B> by *author* ( |
| dns.city-net.com/Ĩmann/awards/hugos/1984.html | <i>*title*</i> by *author* ( |
| dolphin.upenn.edu/d̃cummins/texts/sf-award.htm | *author* \|\| *title* \|\| ( |

**Fig. 2.** Patterns found in first iteration.

# DIPRE

- Produced 4047 book title and author tuples.
- Was done over the same URLS.



| | |
|---|---|
| H. D. Everett | The Death-Mask and Other Ghosts |
| H. G. Wells | First Men in the Moon |
| H. G. Wells | Science Fiction: Volume 2 |
| H. G. Wells | The First Men in the Moon |
| H. G. Wells | The Invisible Man |
| H. G. Wells | The Island of Dr. Moreau |
| H. G. Wells | The Science Fiction Volume 1 |
| H. G. Wells | The Shape of Things to Come: The Ultimate Revolution |
| H. G. Wells | The Time Machine |
| H. G. Wells | The War of the Worlds |
| H. G. Wells | When the Sleeper Wakes |
| H. M. Hoover | Journey Through the Empty |
| H. P. Lovecraft & August Derleth | The Lurker at the Threshold |
| H. P. Lovecraft | At the Mountains of Madness and Other Tales of Terror |
| H. P. Lovecraft | The Case of Charles Dexter Ward |
| H. P. Lovecraft | The Doom That Came to Sarnath and Other Stories |

**Fig. 3.** Sample of books found in first iteration.

# Snowball

- (Agichtein & Gravano, 2000)

- Weakly supervised, bootstrapping method for learning relations between two **named entities**.

- Begins with seed tuples

- Iteratively learns relation patterns and new instances

- Relies heavily on the NER to identify context

# Snowball

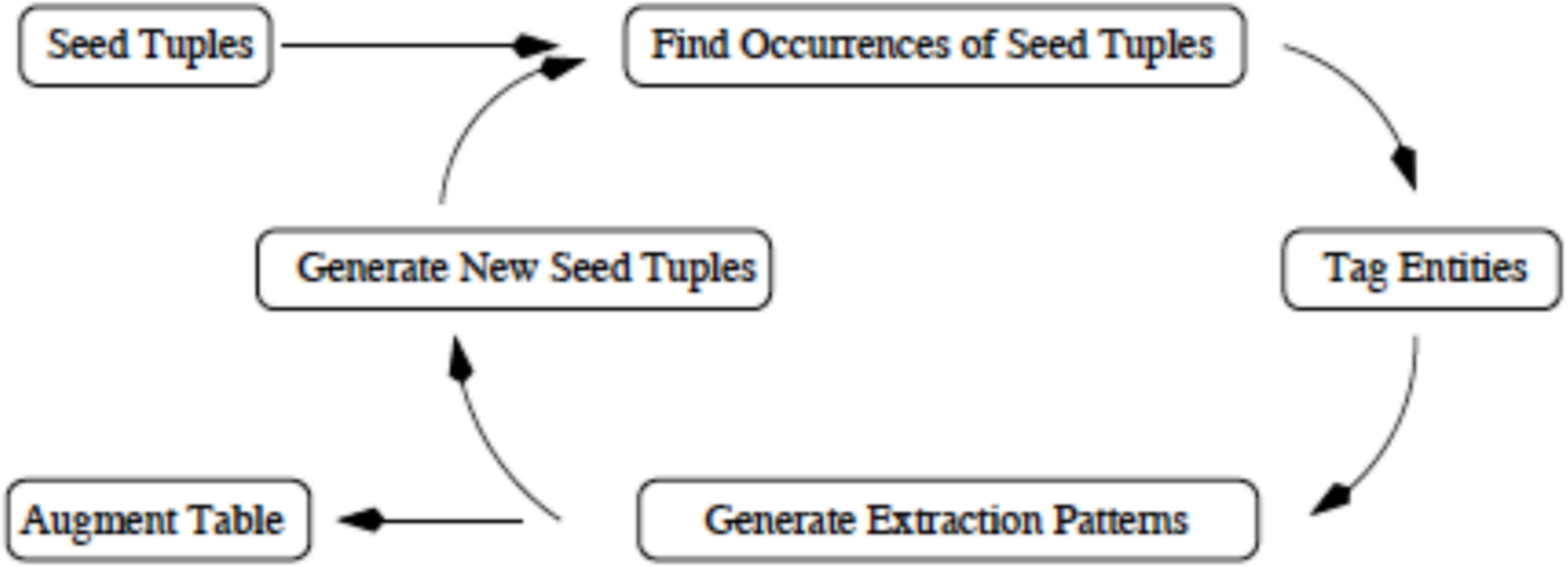


**Figure 2: The main components of *Snowball*.**

# Pattern Representation

- Similar to DIPRE: <Left, Tag1, Middle, Tag2, Right>
- Tag1 and Tag2 are named entity classes.
- Left, Middle, and Right are vectors of terms (with weights) representing different types of context.
- Example: … at the Apple headquarters in California.
- [(<at, 0.3>, <the, 0.2>), <Org>, (<headquarters, 0.2>, <in, 0.3>), <Location>, ( ) ]

# Pattern Matching

- Given two tuples:
  $T_P$ = <$L_P$, $T_1$, $M_P$, $T_2$, $R_P$> $T_S$ = <$L_S$, $T'_1$, $M_S$, $T'_2$, $R_S$>

- The **degree of match**: Match($T_P$,$T_s$) =

  - $L_P \circ L_S + M_P \circ M_S + R_P \circ R_S$ if the tags match

  - 0 otherwise

- The dot indicates the vector dot product.

# Evaluating Patterns

Patterns that extract $< \tau_{SUP}$ seed tuples are filtered, and the rest are assigned a confidence value. Two confidence measures were tried:

$Confidence(P) = P.positive / (P.positive + P.negative )$

$Confidence_{RlogF}(P) = Confidence(P) * \log2(P.positive)$

Since confidence values should range from 0 to 1, $Confidence_{RlogF}$ values are normalized by the largest confidence value of any pattern.

# Discovering new entity pairs

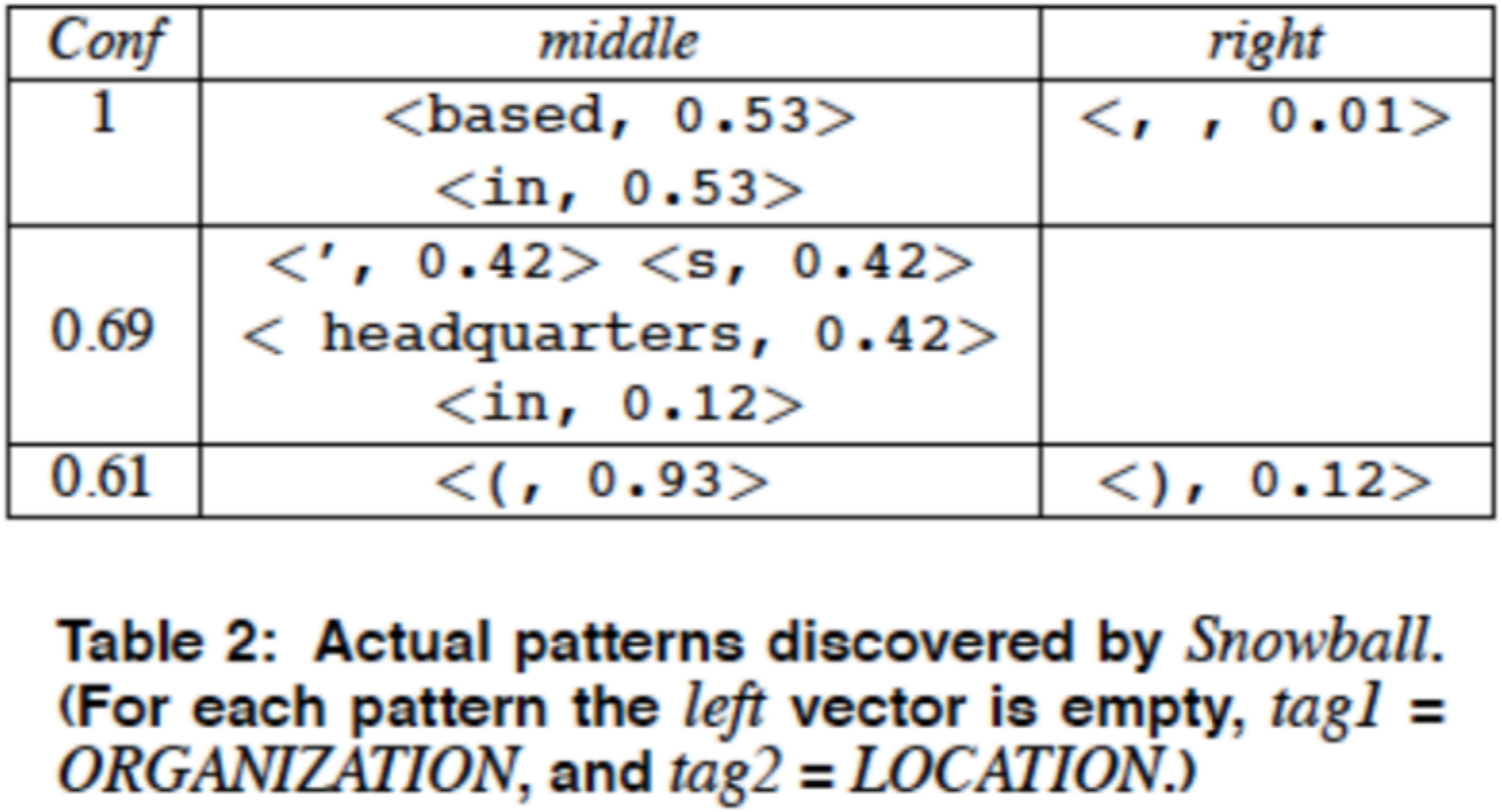
- To discover new *tuples* (entity pairs: <$E_1$, $E_2$>), Snowball first extracts sentences that contain entities of the desired types.

- For each sentence, a 5-tuple is created: T = <$L_P$, $T_1$, $M_P$, $T_2$, $R_P$>, whereT$_1$ is the class of $E_1$ ,and $T_2$ is the class of $E_2$.

- The 5-tuple is matched against the patterns and a candidate tuple (entity pair) is generated for every pattern X such that Match(T, $T_X$) ≥ tao$_{SIM}$

- Each candidate tuple is linked with the set of patterns that generated it and then scored to decide which ones to keep and use for subsequent learning.

# Learned Pattern Examples



| *Conf* | *middle* | *right* |
|---|---|---|
| 1 | <based, 0.53><br><in, 0.53> | <, , 0.01> |
| 0.69 | <', 0.42> <s, 0.42><br>< headquarters, 0.42><br><in, 0.12> | |
| 0.61 | <(, 0.93> | <), 0.12> |

**Table 2: Actual patterns discovered by** *Snowball*. **(For each pattern the** *left* **vector is empty,** *tag1* **=** *ORGANIZATION*, **and** *tag2* **=** *LOCATION*.**)**

# Evaluation

- Snowball was designed to learn LocatedIn(ORG,LOC) relations and produce entity pairs for this relation.

- Snowball's goal was to generate tables of entity pairs from a corpus, as opposed to typical IE systems that want to find every instance of a relation.

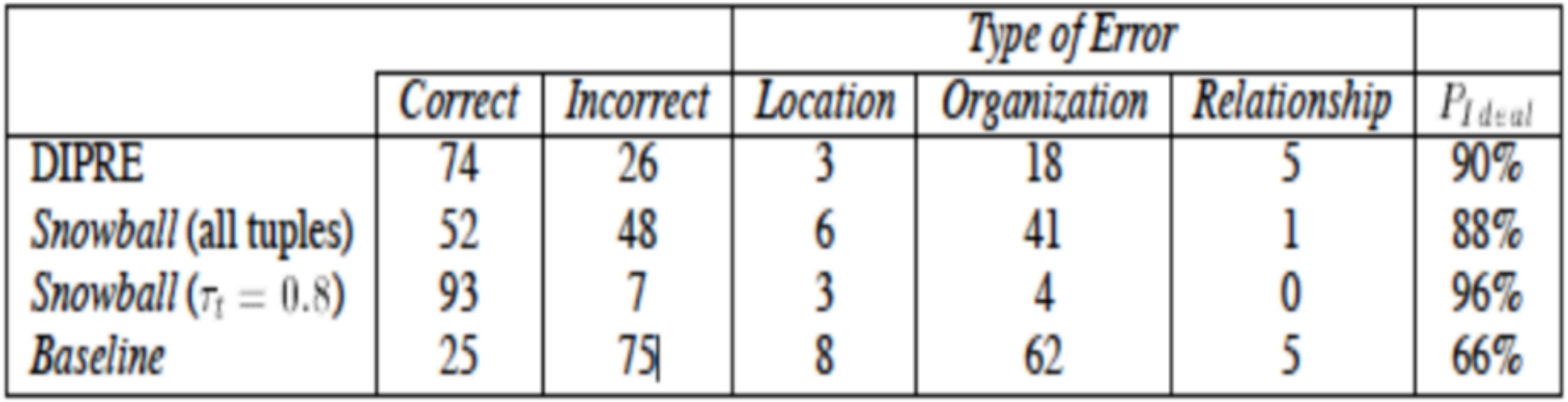
- An "Ideal" set of entity pairs was created by:
  – compiling (ORG,LOC) pairs from "Hoover's Online" web site
  – retained pairs for which the organization name appears in the corpus with its location nearby

- However Hoover's is far from complete. So manual samples of extracted tuples were evaluated by hand.

# Results

100 extracted tuples were evaluated by hand for each system.

Three types of errors were labeled:
Location Errors = mistagging a location (NER error)
Organization Errors = mistagging an organization (NER error)
Relationship Errors = misidentifying the relation



| | | | *Type of Error* | | | |
|---|---|---|---|---|---|---|
| | *Correct* | *Incorrect* | *Location* | *Organization* | *Relationship* | $P_{Ideal}$ |
| DIPRE | 74 | 26 | 3 | 18 | 5 | 90% |
| *Snowball* (all tuples) | 52 | 48 | 6 | 41 | 1 | 88% |
| *Snowball* ($\tau_t = 0.8$) | 93 | 7 | 3 | 4 | 0 | 96% |
| *Baseline* | 25 | 75 | 8 | 62 | 5 | 66% |

# Approaches for Relation Extraction

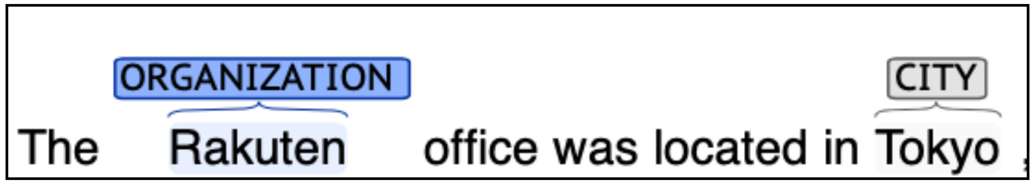
1. Handwritten Rules
2. Bootstrapping approaches
3. **Supervised approaches**
4. Distant Supervision

# Supervised Approaches

- Useful for domain specific data
- Annotate a few instances
- Define features and see which ones produce best results
- Train a classifier to predict a relation given two entities (and features)
- Commonly used in medical relation extractions, etc.

# Basic Features

- Entity Features:
  - the types of the named entities (person, organization, etc)
  - the mention types of the entities (name, nominal, pronoun)
  - the head noun of the entities
  - Indicate entity order

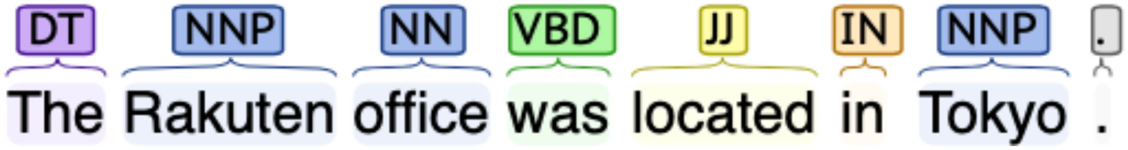


The Rakuten office was located in Tokyo

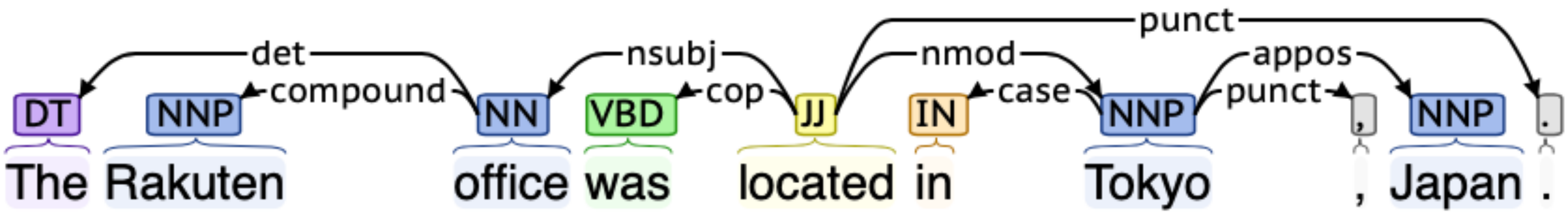
- Lexical Context:
  - Words before, between and after the entities
  - the distance between the entities
  - whether other mentions occur between entities

# Syntactic Features

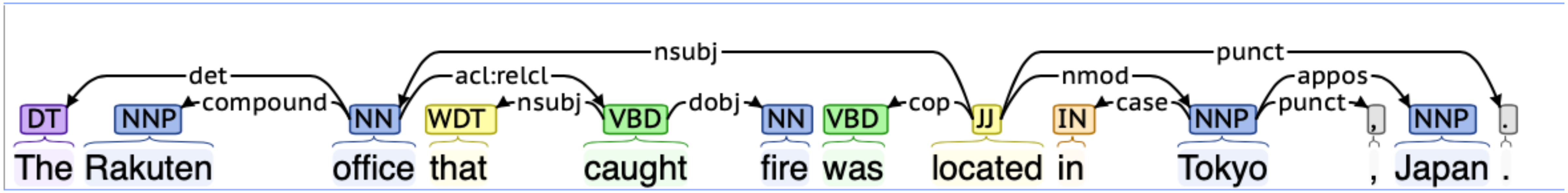
- Chunking-based features
    - phrasal heads containing the entities
    - phrasal heads of before/between/after contexts
- Dependency parsing features
    - dependency relations linked to the entities
    - pairs of heads or entity types and dependent words
- Parse Tree Features
    - POS tags
    - Parse tree paths connecting one entity to the other.



# Dependency Paths



- Very useful feature: traversal path between two entities.
- Example: ← compound ← nsubj → nmod



# Semantic Features

- Semantic class information can be used to distinguish between relation subtypes.

- Semantic knowledge is typically based on WordNet, lists harvested from the Web, or manually defined (e.g., family member terms are a relatively small set).

# Approaches for Relation Extraction

1. Handwritten Rules

2. Bootstrapping approaches

3. Supervised approaches

4. **Distant Supervision**

# Distant Supervision

- Using a large external knowledge base to provide some positive results and replicate them.
- Hypothesis: If two entities belong to a certain relation, any sentence containing those two entities is likely to express that relation
- Key idea: use a *database* of relations to get lots of *noisy* training examples
  - instead of hand-creating seed tuples (bootstrapping)
  - instead of using hand-labeled corpus (supervised)

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

# Distant Supervision

- Has advantages of supervised approach
  - leverage rich, reliable hand-created knowledge
  - relations have canonical names
  - can use rich features (e.g. syntactic features)
- Has advantages of unsupervised approach
  - leverage unlimited amounts of text data
  - allows for very large number of weak features
  - not sensitive to training corpus: genre-independent
- Bad: data will be “noisy”!!

# Examples of <Microsoft, Redmond>

**Microsoft** Research was founded on the **Microsoft Redmond** campus.

**Microsoft** Corporation Corporate Headquarters One **Microsoft** Way **Redmond**, WA 98052-6399. UNITED STATES

Find buildings in the **Microsoft Redmond** Main Campus in **Redmond**, WA

18617 salaries for 818 jobs at **Microsoft** in **Redmond**

Salaries posted anonymously by **Microsoft** employees in **Redmond.**

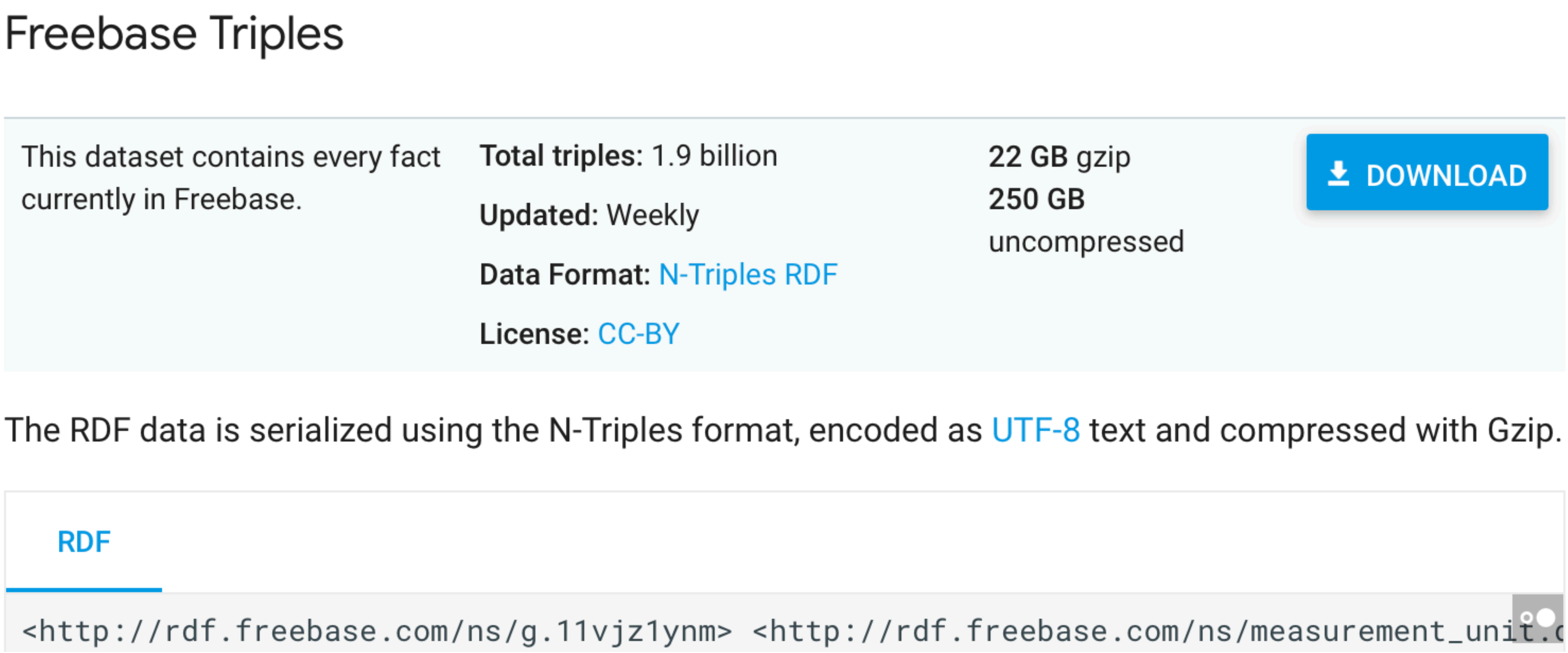
**Microsoft** Jobs available in **Redmond** , WA

**Microsoft** is considering a multibillion-dollar revamp of its headquarters campus in **Redmond**

**Redmond** is commonly recognized as the home of both **Microsoft** and Nintendo.

# Freebase

- Huge database of triples/ relations.
- Now part of Wikidata.



## Freebase Triples

This dataset contains every fact currently in Freebase.

**Total triples:** 1.9 billion

**Updated:** Weekly

**Data Format:** N-Triples RDF

**License:** CC-BY

**22 GB** gzip
**250 GB** uncompressed

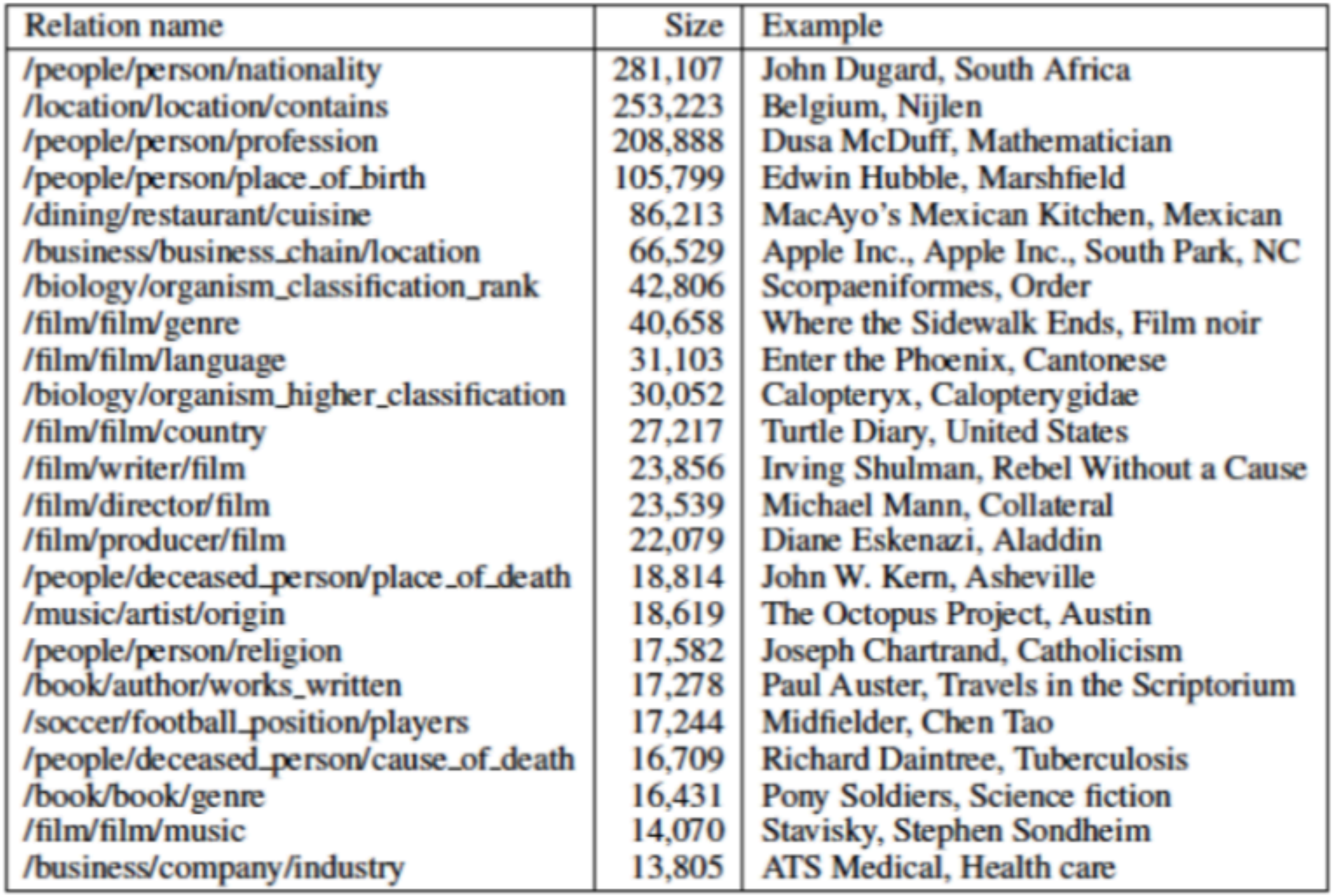
The RDF data is serialized using the N-Triples format, encoded as UTF-8 text and compressed with Gzip.

```
<http://rdf.freebase.com/ns/g.11vjz1ynm> <http://rdf.freebase.com/ns/measurement_unit.
```

# Distant Supervision for Relation Extraction without Labeled Data

- [Mintz et al., ACL-IJCNLP 2009] used distant supervision from Freebase to train a relation extractor.
- They used data for the 102 largest relations, which had 1.8 million instances connecting 940,000 entities.
- “relation” is an ordered, binary relation between entities.
- Example: person-nationality
- “relation instance” is an ordered pair of specific entities that participate in the relation.
- Example: (John Steinbeck, United States)

# Sample of Freebase Relations



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

Table 2: The 23 largest Freebase relations we use, with their size and an instance of each relation.

# Approach

- Apply an NER tagger to identify entities.

- Extract sentences that contain two entities of types that can participate in a relation.

- Group all contexts that correspond to the same relation instance.

- The collective contexts serve as a single positive training example.

Example: Suppose a pair of entities occurs in 10 sentences.If each sentence has 3 features extracted, then the entity pair instance will have 30 features in its feature vector.

- Train a multiclass logistic regression classifier to predict a relation name given a pair of entities (i.e. the feature vector for the pair).

# Why collective contexts?

A key advantage of merging contexts from multiple instances is that some mentions will occur in relation contexts, some in ambiguous contexts, and some in non-relation contexts.

Example:

Steven Spielberg’s film Saving Private Ryan is loosely based on the brothers’ story.

Allison co-produced the Academy Award-winning Saving Private Ryan, directed by Steven Spielberg.

# Lexical Features

Given a context containing two entities, the following information is

extracted.

– the sequence of words between them

– the POS tags for the words between them

– a flag indicating which entity appeared first

– a window of k words to the left of Entity #1 and their POS tags

– a window of k words to the right of Entity #2 and their POS tags

– the named entity tags for the two entities

Each lexical feature is the conjunction of all this information. One conjunctive feature is generated for each k in {0,1 2}

# Syntactic Features

Syntactic features are also generated from a dependency parse of the sentence.

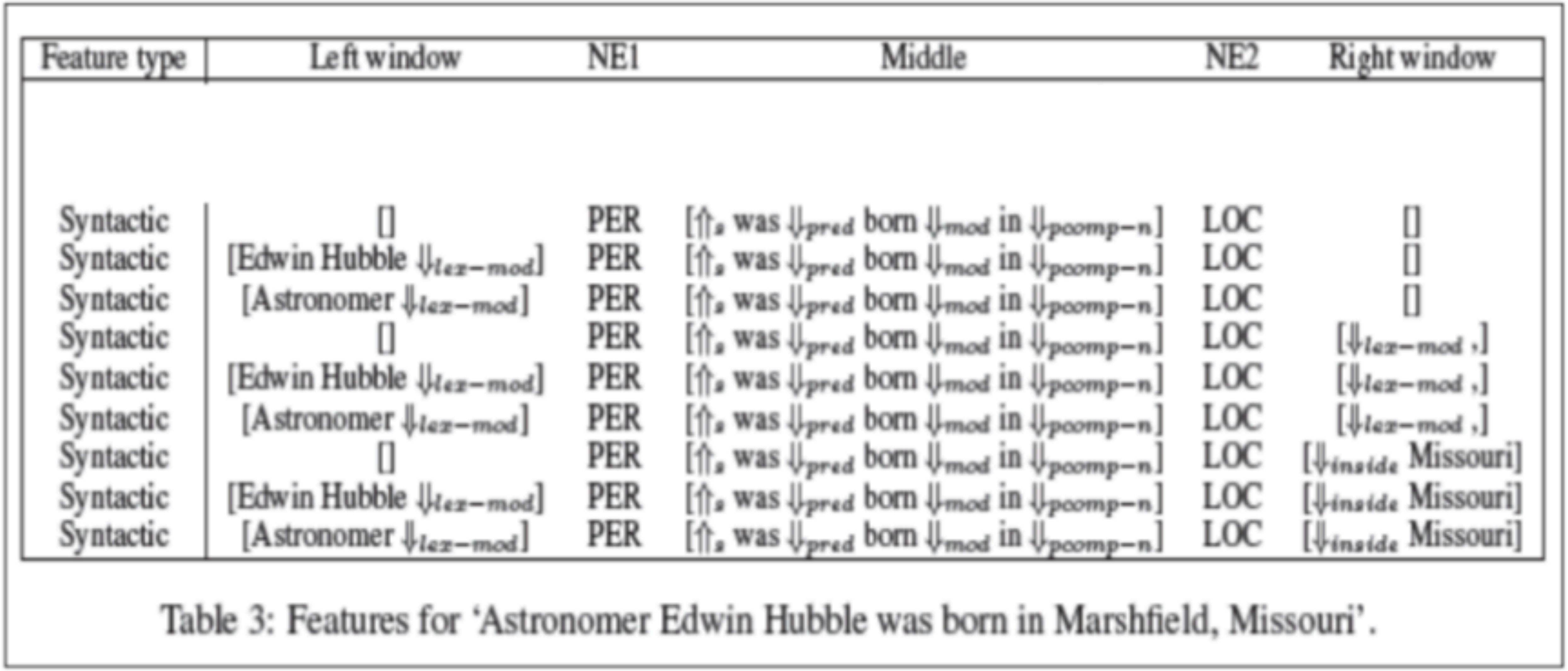
– A dependency path between the two entities, which is a series of dependencies, directions, and words/chunks representing a traversal of the parse.

– For each entity, one window node that is not part of the dependency path.

– the named entity tags for the two entities

Each syntactic feature is a conjunction of this information. One conjunctive feature is generated for each pair of left and right window nodes, as well as features that omit one or both.

# Example of features



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

Table 3: Features for ‘Astronomer Edwin Hubble was born in Marshfield, Missouri’.

# Classifier

As negative training data, random entity pairs that do not participate in a Freebase relation are used to generate feature vectors for an “unrelated” relation. This may produce some noise, but the effect should be small.

They randomly sample 1% of entity pairs that are not in a Freebase relation.

Testing: a multi-class logistic classifier takes an entity pair as input, constructs a feature vector for it, and returns a relation name with a confidence score.

All entity pairs can then be ranked by their confidence scores to identify the N most likely new relation instances.

# Text Corpus

Corpus: full text of all Wikipedia articles.

–1.8 million articles, 14.3 sentences per article on average

–800,000 used for training, 400,000 used for testing

Wikipedia texts chosen because:

–“sentences tend to make explicit many facts that might be omitted in newswire”

–“most of the information in Freebase is derived from tabular data from Wikipedia, meaning that Freebase relations are more likely to appear in sentences in Wikipedia”

# Analysis

The syntactic features showed benefits over just the lexical features, so they inspected examples to understand how they helped.

The syntactic features consistently helped with the director-film and writer-film relations, which are particularly ambiguous.

They observed many examples with a large distance between the director's name and the film, for example: *Back Street is a 1932 film made by Universal Pictures, directed by John M. Stahl, and produced by Carl Laemmle Jr.*

These cases would have long lexical features, but often short dependency paths.

# Conclusion

- Viable option if searching for specific relation
- Need to rely on parsing, semantic labelling, NER
- Choose an approach that works for your domain/needs.