

# **Entity Linking**

**Laura Dietz**

University of New Hampshire

**Alexander Kotov**

Wayne State University

**Edgar Meij**

Bloomberg

# Outline

- Entity Linking
  - introduction
  - methods
  - evaluation

# Introduction



Image taken from Mihalcea and Csomai (2007). Wikify!: linking documents to encyclopedic knowledge. In CIKM '07.

article discussion edit this page history

You're running!

# Plant

From Wikipedia, the free encyclopedia

For other uses, see *Plant* (disambiguation).

**Plants** are a major group of living things including familiar organisms such as trees, flowers, herbs, ferns, and mosses. About 350,000 species of plants, defined as seed plants, bryophytes, ferns and fern allies, have been estimated to exist. As of 2004, some 287,655 species had been identified, of which 258,650 are flowering and 15,000 bryophytes.

Tree

From Wikipedia, the free encyclopedia

For other senses of the word, see tree (disambiguation)

A tree is a large, perennial, woody plant. Though there is no set definition regarding minimum size, the term generally applies to plants at least 6 m (20 ft) high at maturity and, more importantly, having



Fossil range: Middle-Late Ordovician - Recent



Species

From Wikipedia, the free encyclopedia

This article is about biology. For the movie, see Species (film). In biology, a species is one of the basic units of biodiversity. In classification, a species is assigned a two-part name; the genus is listed first (with its leading letter capitalized), followed by the species. For example, humans belong to the genus *Homo*, and species *Homo sapiens*. The name of the species is the whole second part of the name.

# Why do we need entity linking?

- Enable
  - semantic search
  - advanced UI/UX
  - automatic document enrichment; go-read-here
  - inline annotations (microformats, RDFa)
  - ontology learning, KB population
- “Use as feature”
  - to improve
    - end-to-end retrieval, classification, word sense disambiguation, semantic similarity, ...
  - dimensionality reduction (e.g., term vectors)

# Main problem

# Main problem

- Linking free text to entities
  - Any piece of text
    - news documents
    - blog posts
    - tweets
    - queries
    - ...
  - Entities: typically taken from a knowledge base
    - Wikipedia
    - Freebase
    - ...

# Methods



# Common steps

1. Determine “linkable” phrases
  - mention detection – **MD**
2. Identify candidate entities for each mention
  - link generation – **LG**
  - may include NILs (null values, i.e., no target in KB)
3. (Use “context” to disambiguate/filter/improve)
  - disambiguation – **DA**

# **Wikipedia-based methods**

# Wikipedia-based methods

- Basic element: article proper
- But also
  - redirect pages
  - disambiguation pages
  - category/template pages
  - admin pages
- Hyperlinks
  - use “unique identifiers” (URLs)
    - [[United States]] or [[United States|American]]
    - [[United States (TV series)]] or  
[[United States (TV series)|TV show]]



# Wikipedia-based methods

- keyphraseness( $w$ ) **[Mihalcea & Csomai 2007]**

$$\frac{\text{CF}(w_l)}{\text{CF}(w)}$$

# Wikipedia-based methods

- keyphraseness( $w_l$ ) [Mihalcea & Csomai 2007]

$$\frac{\text{CF}(w_l)}{\text{CF}(w)} \longrightarrow \begin{array}{l} \textbf{Collection frequency} \\ \text{phrase } w \text{ as a link to another} \\ \text{Wikipedia article} \end{array}$$


**Collection frequency**  
phrase  $w$

# Wikipedia-based methods

- commonness( $w, c$ ) [Medelyan et al. 2008]

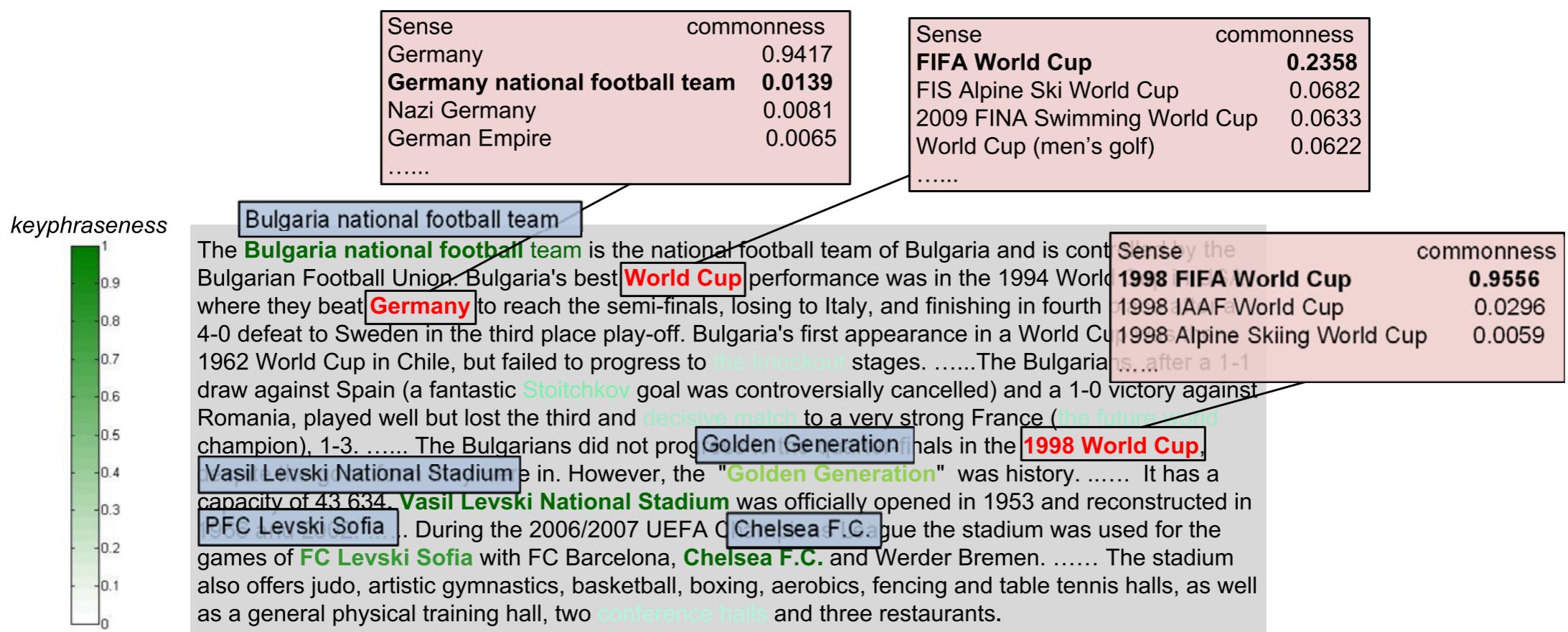
$$\frac{|L_{w,c}|}{\sum_{c'} |L_{w,c'}|}$$

  
**Number of links**  
with target  $c'$  and anchor text  $w$

# Wikipedia-based methods

- Of course, these can also be based on other data, e.g.,
  - (focused) web crawls, with anchor text in the links to Wikipedia articles **[Chisholm & Hachey 2015]**
  - click logs **[Pantel et al. 2011]**

# Commonness and keyphraseness



# Learning to Link with Wikipedia

[Milne & Witten 2008b]

- Key idea: disambiguation informs detection
  - start with unambiguous senses
  - compare each possible sense with its *relatedness* to the context sense candidates
  - So, first LG, then base MD on these results

# Learning to Link with Wikipedia

## [Milne & Witten 2008b]

**Depth-first search**

From Wikipedia, the free encyclopedia



**Depth-first search (DFS)** is an [algorithm](#) for traversing or searching a [tree](#) [tree structure](#) or [graph](#). One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before [backtracking](#).

Formally, DFS is an [uninformed search](#) that progresses by expanding the first child node of the search [tree](#) that appears and thus going deeper and deeper until a goal node is found, or until it hits a node that has no children. Then the search [backtracks](#), returning to the most recent node it hadn't finished exploring. In a non-recursive implementation, all freshly expanded nodes are added to a [LIFO stack](#) for exploration.

| sense                        | commonness   |
|------------------------------|--------------|
| Tree                         | 92.82%       |
| Tree (graph theory)          | 2.94%        |
| <b>Tree (data structure)</b> | <b>2.57%</b> |
| Tree (set theory)            | 0.15%        |
| Phylogenetic tree            | 0.07%        |
| Christmas tree               | 0.07%        |
| Binary tree                  | 0.04%        |
| Family tree                  | 0.04%        |
| ...                          |              |

# Learning to Link with Wikipedia

## [Milne & Witten 2008b]

### Depth-first search

From Wikipedia, the free encyclopedia

**Depth-first search (DFS)** is an algorithm for traversing or searching a tree structure or graph. One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before backtracking.

Formally, DFS is an uninformed search that progresses by expanding the first child node of the search tree that appears and thus going deeper and deeper until a goal node is found, or until it hits a node that has no children. Then the search backtracks, returning to the most recent node it hadn't finished exploring. In a non-recursive implementation, all freshly expanded nodes are added to a LIFO stack for exploration.



| sense                        | commonness   |
|------------------------------|--------------|
| Tree                         | 92.82%       |
| Tree (graph theory)          | 2.94%        |
| <b>Tree (data structure)</b> | <b>2.57%</b> |
| Tree (set theory)            | 0.15%        |
| Phylogenetic tree            | 0.07%        |
| Christmas tree               | 0.07%        |
| Binary tree                  | 0.04%        |
| Family tree                  | 0.04%        |
| ...                          |              |

# Learning to Link with Wikipedia

## [Milne & Witten 2008b]

### Depth-first search

From Wikipedia, the free encyclopedia

**Depth-first search (DFS)** is an algorithm for traversing or searching a tree structure or graph. One starts at the root (selecting some node as the root in the graph case) and explores as far as possible along each branch before backtracking.

Formally, DFS is an uninformed search that progresses by expanding the first child node of the search tree that appears and thus going deeper and deeper until a goal node is found, or until it hits a node that has no children. Then the search backtracks, returning to the most recent node it hadn't finished exploring. In a non-recursive implementation, all freshly expanded nodes are added to a LIFO stack for exploration.



| sense                        | commonness   | relatedness   |
|------------------------------|--------------|---------------|
| Tree                         | 92.82%       | 15.97%        |
| Tree (graph theory)          | 2.94%        | 59.91%        |
| <b>Tree (data structure)</b> | <b>2.57%</b> | <b>63.26%</b> |
| Tree (set theory)            | 0.15%        | 34.04%        |
| Phylogenetic tree            | 0.07%        | 20.33%        |
| Christmas tree               | 0.07%        | 0.0%          |
| Binary tree                  | 0.04%        | 62.43%        |
| Family tree                  | 0.04%        | 16.31%        |
| ...                          |              |               |

# Entity relatedness

- relatedness( $c, c'$ ) [Milne & Witten 2008a]

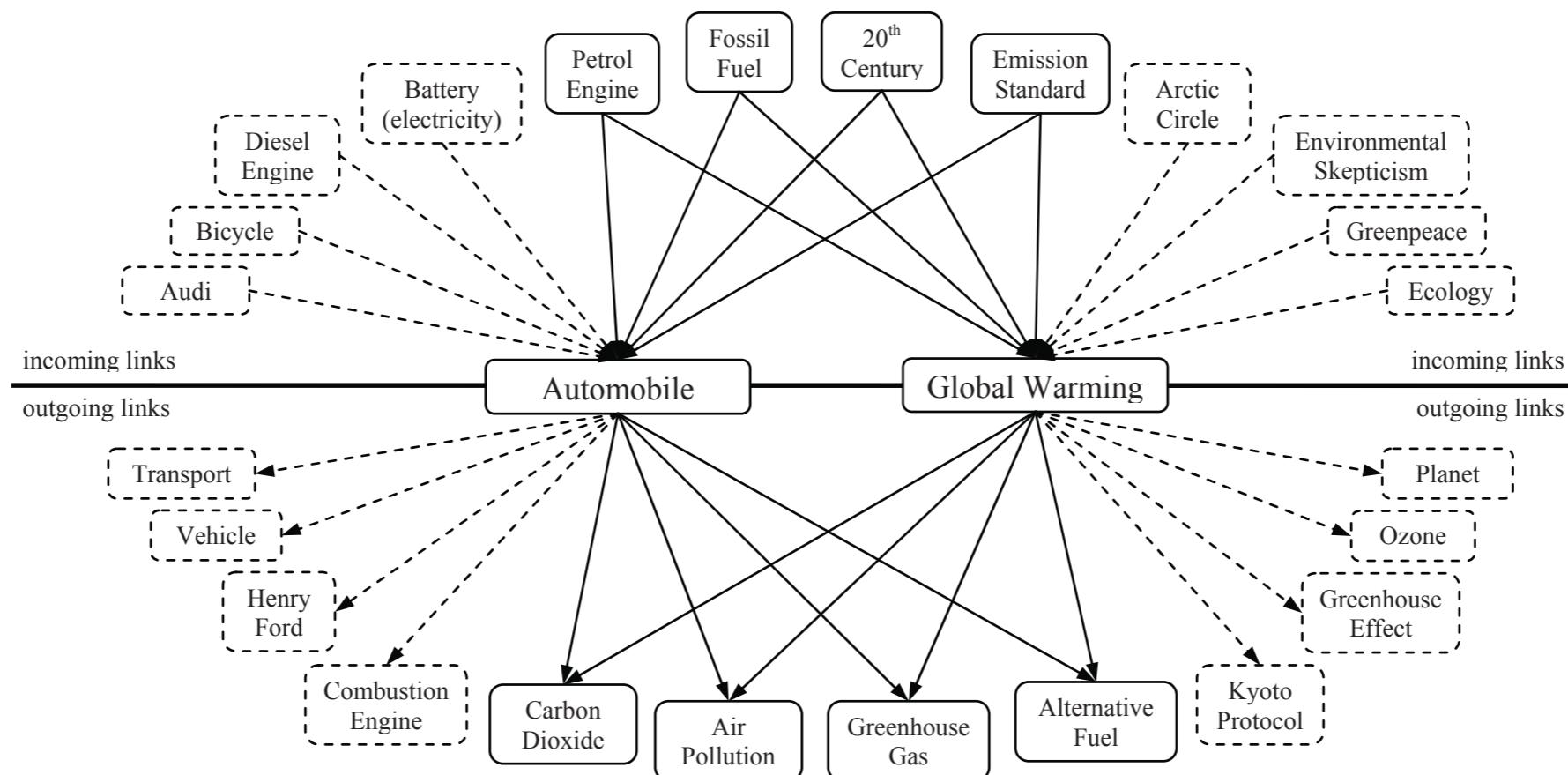


Image taken from Milne and Witten (2008a). An Effective, Low-Cost Measure of Semantic Relatedness Obtained from Wikipedia Links. In AAAI WikiAI Workshop.

# Entity relatedness

- relatedness( $c, c'$ ) [Milne & Witten 2008a]

$$\frac{\log(\max(|L_c|, |L_{c'}|)) - \log(|L_c \cap L_{c'}|)}{\log(|WP|) - \log(\min(|L_c|, |L_{c'}|))}$$

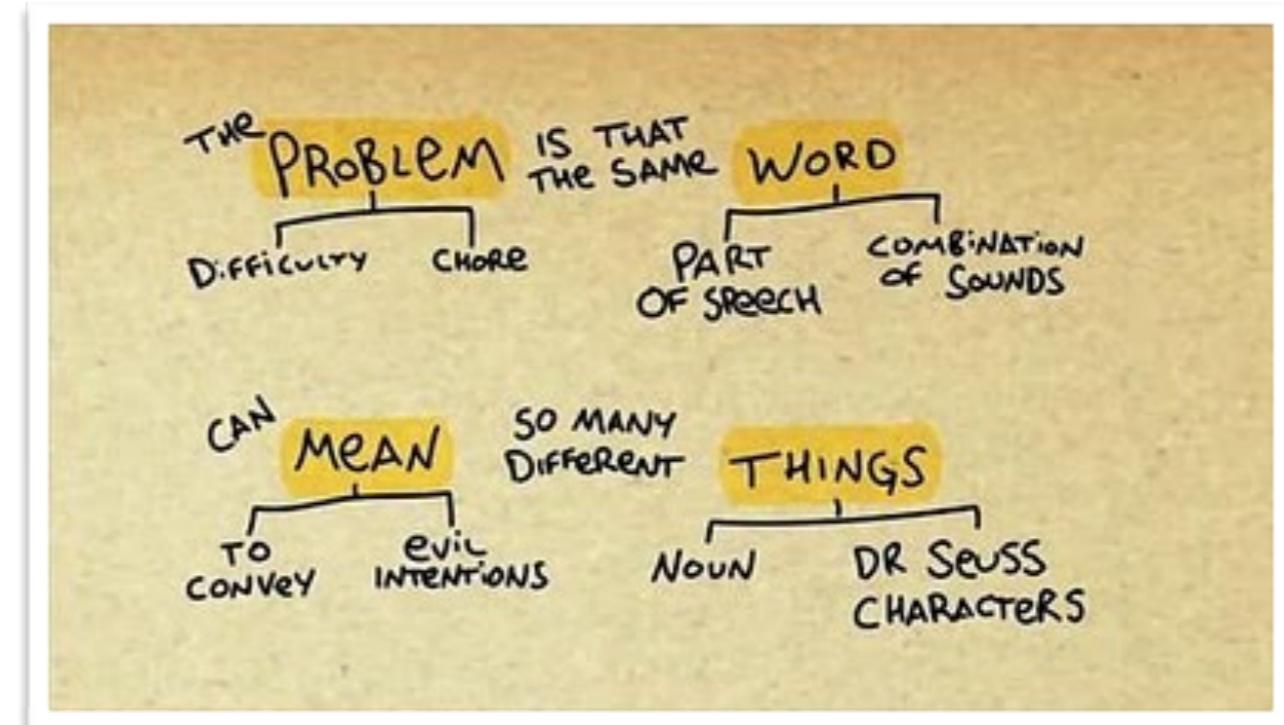
Number of links  
with target  $c$

Intersection of inlinks  
with target  $c$  and  $c'$

Total number of  
Wikipedia articles

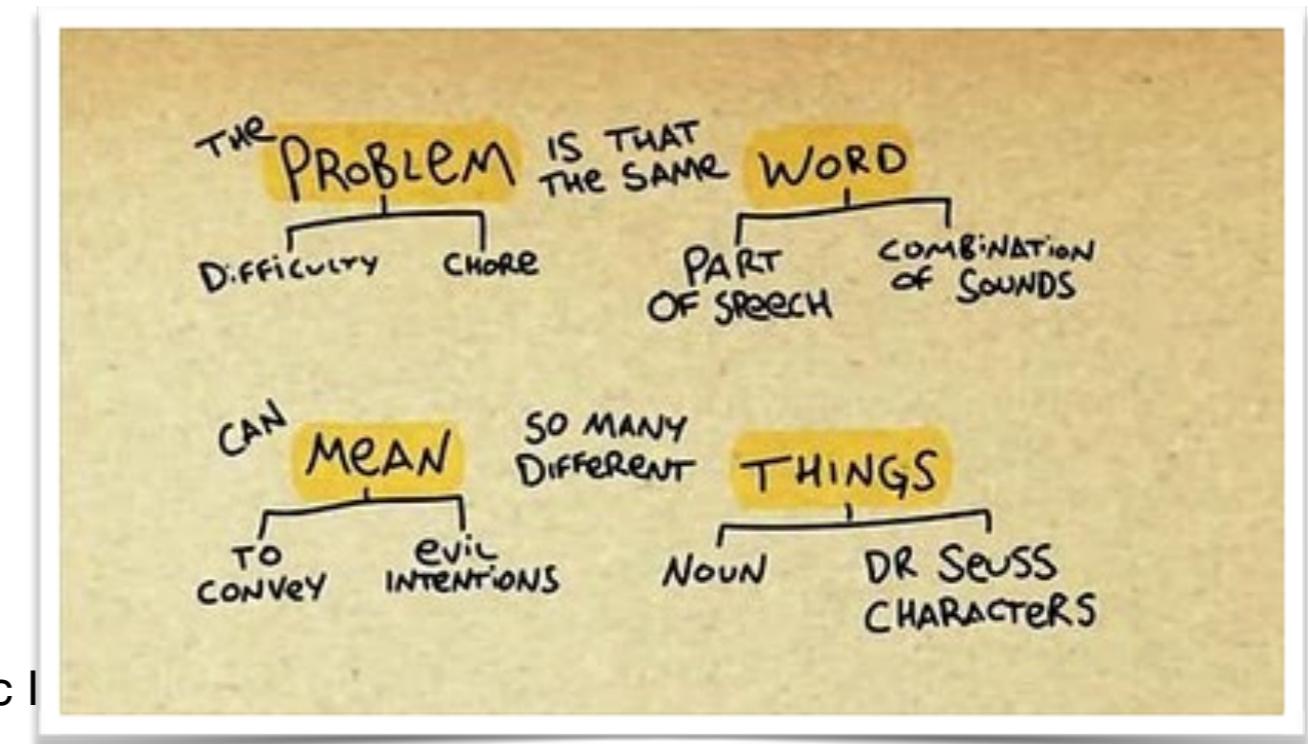
```
graph TD; E[entity relatedness] --> N1["Number of links  
with target c"]; E --> N2["Intersection of inlinks  
with target c and c'"]; N2 --- B[ ]; E --> T["Total number of  
Wikipedia articles"]
```

# Context



# Main intuition

- Leverage “context” as disambiguation signal
  - query: history, session, interests; ...
  - phrase: sentence, paragraph, document, ...
  - But also candidate entity context
    - e.g., KG, candidate entity graph, relatedness, ...



# Local versus global context

- “Global”
  - i.e., disambiguation of the whole candidate entity graph
  - NP-hard
- Optimization
  - reduce the search space to a “disambiguation context”
    - all plausible (reciprocal) disambiguations **[Cucerzan 2007]**
    - unambiguous surface forms, pair-wise comparisons, and/or averages **[Milne & Witten 2008b]**
    - hill-climbing, integer linear programs **[Kulkarni et al. 2009]**
    - hybrid + ML **[Ratinov et al. 2011, Ferragina & Scaiella 2010]**

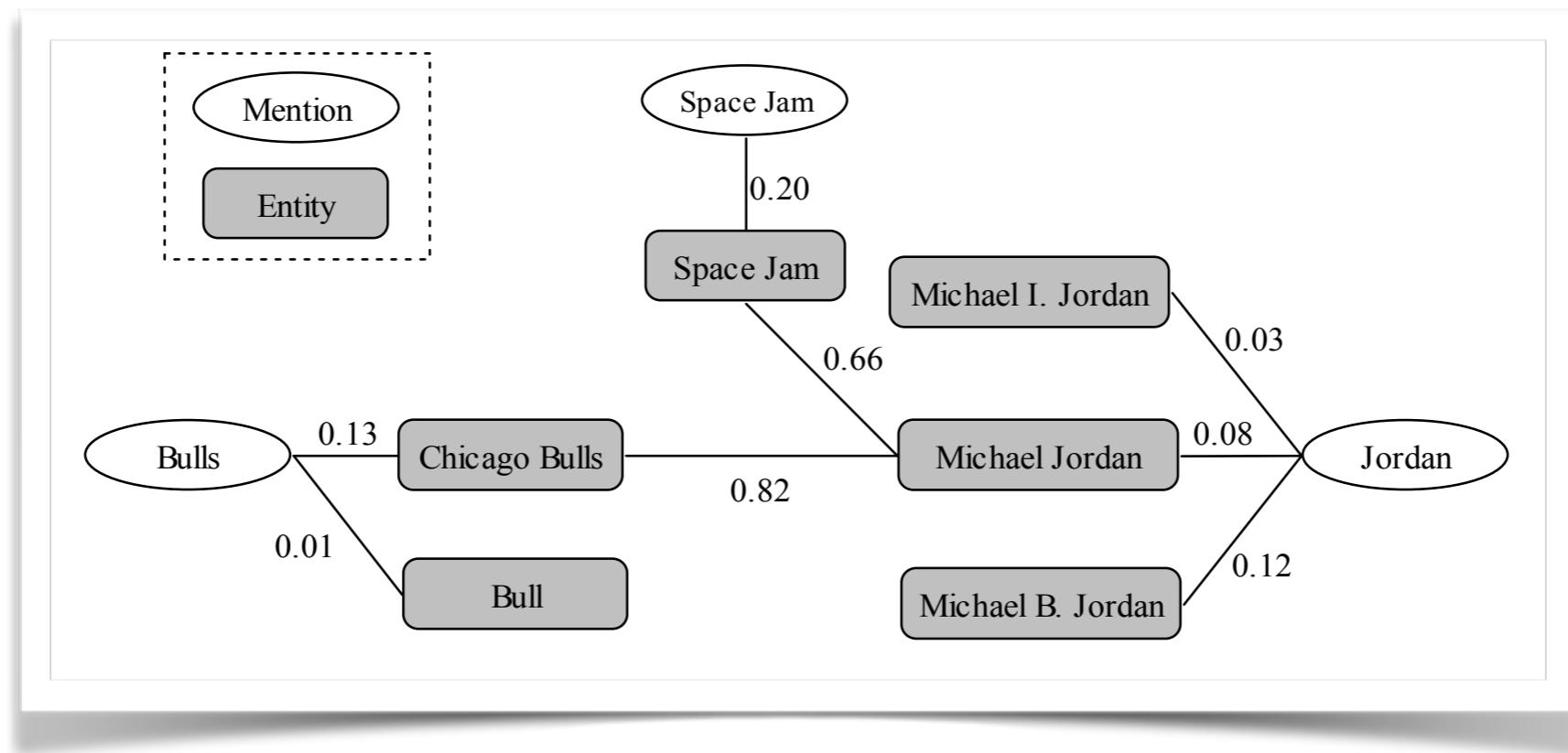
# **Local and Global Algorithms for Disambiguation to Wikipedia**

**[Ratinov et al. 2011]**

- Main contribution, in steps – MD + DA
  - 1. use “local” approach (e.g., commonness) to generate a disambiguation context
  - 2. apply “global” machine learning approach on pairs
    - relatedness, PMI
      - {inlinks, outlinks} in various combinations (c and c')
      - {avg, max}

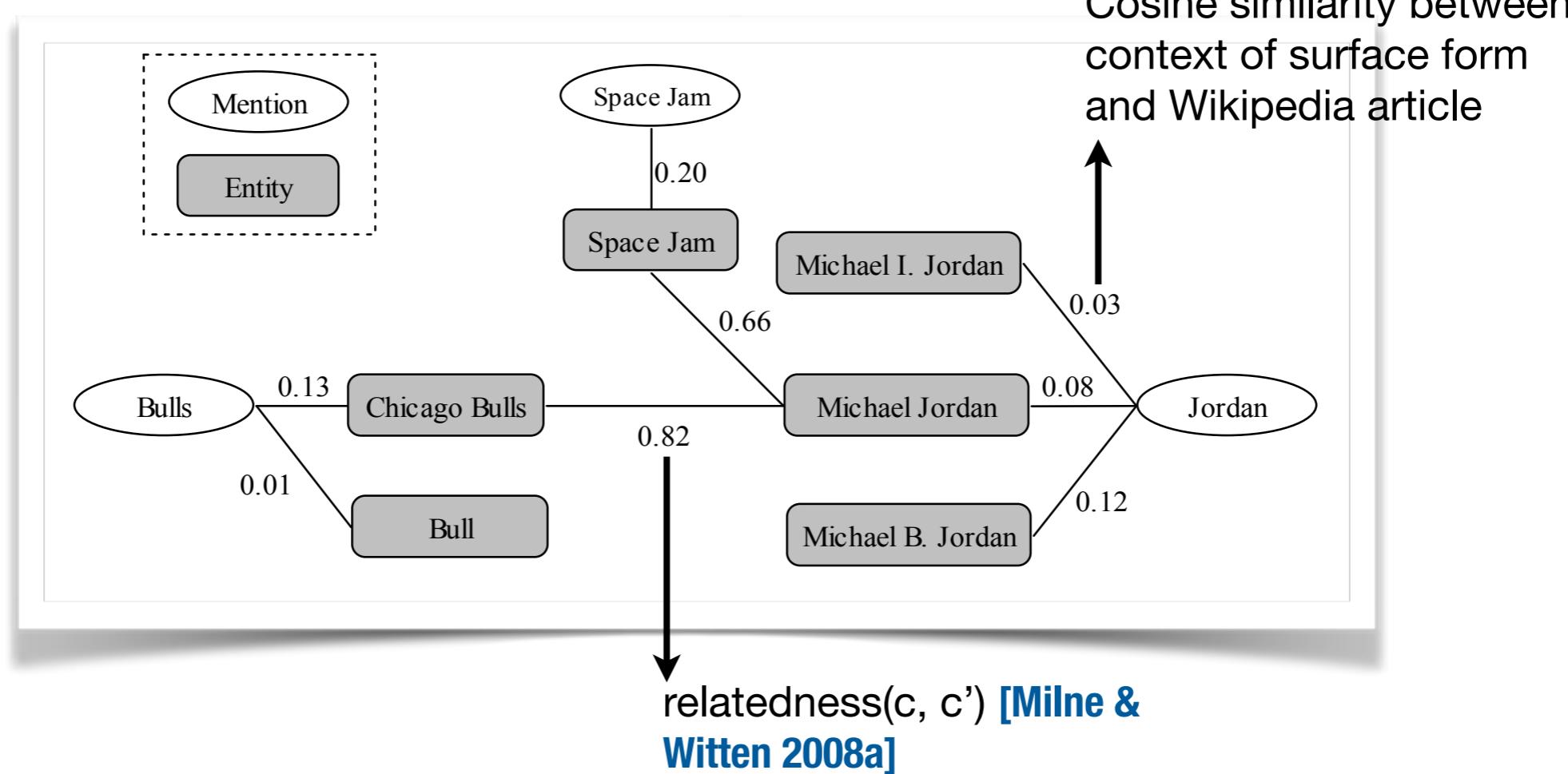
# Graph-based approaches

- Random walk with restarts [Guo & Barbosa 2014]
- Random walk on a graph defined by the intra-Wikipedia links [Han et al. 2011]



# Graph-based approaches

- Random walk with restarts [Guo & Barbosa 2014]
- Random walk on a graph defined by the intra-Wikipedia links [Han et al. 2011]



# Topic modeling approaches

- “Extend” BOW approaches to include disambiguation with LDA topics **[Han & Sun 2012]**
  - compare topic distributions of source document with candidate entities **[Pilz et al. 2011]**

**Table 1: Topics for entities with name *John Taylor* (excerpt) with associated probability value**

| disambiguation term     | $i$ | $p(t_i)$ | Important words (titles) of the topics  |
|-------------------------|-----|----------|---|
| South Carolina governor | 109 | 0.3805   | unit state, state senat, lieuten governor, hous repres, elect governor, ...       |
|                         | 120 | 0.2477   | north carolina, south carolina, unit state, west virginia, civil war, ...         |
| athlete                 | 80  | 0.4190   | summer olymp, gold medal, world record, silver medal, world championship, ...     |
|                         | 135 | 0.1047   | unit state, rhode island, baltimore maryland, new hampshir, georg washington, ... |
| racing driver           | 129 | 0.7407   | grand prix, race driver, motor race, formula, race team, sport car, ...           |
| jazz                    | 141 | 0.5781   | jazz musician, big band, new york, duke ellington, jazz band, ...                 |
| bass guitarist          | 18  | 0.2964   | rock band, solo album, play guitar, band member, rock roll, ...                   |
|                         | 70  | 0.1594   | album releas, studio album, debut album, record label, music video, ...           |

# Topic modeling approaches

- “Extend” BOW approaches to include disambiguation with LDA topics [**Han & Sun 2012**]
  - compare topic distributions of source document with candidate entities [**Pilz et al. 2011**]
- Associate each entity with a latent topic
  - Semi-supervised [**Kataria 2011**] or using sparse Gibbs sampling [**Houlsby & Ciaramita 2014**]

# Word embeddings

- Jointly embedding words and entities, using anchors as “bridge”
  - **[Wang et al. 2014]**
- PageRank on an “embeddings graph”
  - **[Zwicklbauer et al. 2016]**
- Disambiguation
  - **[He et al. 2013, Blanco et al. 2015]**

# Evaluation

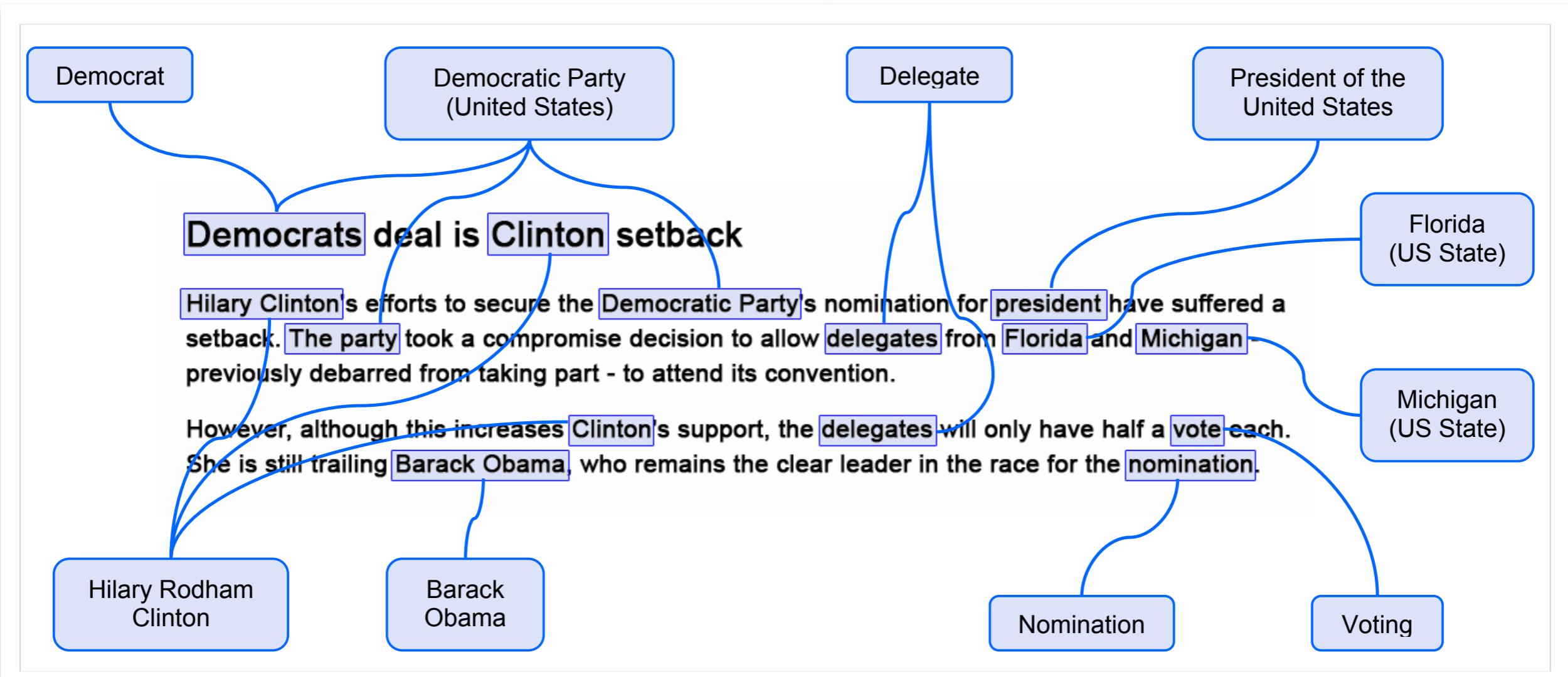


# Entity linking evaluation

- Ingredients
  - knowledge base
  - input texts
  - gold-standard annotations
  - evaluation metrics

# Evaluation metrics

- What is the task?



# Evaluation metrics

- Compounded problem
  - tagging/spanning – **MD**
  - entity linking – **LG/DA**

# Evaluation metrics

- Set-based (similar to WSD)
  - “How many correct links were retrieved?”
    - precision, recall, F-measure
- Rank-based
  - “Was the correct link(s) retrieved with a high score?”
    - precision@k, recall@k, P1, R-prec, MRR, MAP, etc.
- macro/micro
  - per mention
  - per tweet, query, sentence, document

# **Test collections**



# **Gold-standard annotations**

- Human annotators, labeling or judging links from input “documents” to entities in the KB

# Entity linking test collections

- Wikipedia
- MSNBC
- AQUAINT
- ACE
- Twitter
- AIDA (CoNLL)
- IITB (web data)
- INEX link-the-wiki
- TREC knowledge base acceleration (KBA)
- TAC knowledge base population (KBP)
- Yahoo Webscope: web search queries (in sessions)
- ERD 2014: <http://web-ngram.research.microsoft.com/ERD2014/>
- ...

# Evaluation - recap

- Even with so many test collections to choose from, there's still quite some variation
- People create their own “extracts” from WP
- Same method, same test collection, but different results in different papers
  - tokenization, normalization, ...
- We need meta-evaluations...
  - **[Hachey et al. 2013], [Cornolti et al. 2013]**

# A Framework for Benchmarking Entity-Annotation Systems

[Cornolti et al. 2013]

- Compare five publicly available entity linkers
  - [Hoffart et al. 2007] (AIDA)
  - [Ratinov et al. 2011]
  - [Ferragina & Scaiella 2010] (TAGME)
  - [Milne & Witten 2008] (wikipedia-miner)
  - DBpedia Spotlight
- And also investigate parameter/cut-off settings

# A Framework for Benchmarking Entity-Annotation Systems

[Cornolti et al. 2013]

- On five publicly available test collections
  - AIDA **[Hoffart et al. 2007]**
    - based on CoNLL 2003: noun annotations
    - 1393 Reuters newswire articles
    - hand-annotated all nouns with entities in YAGO2
  - AQUAINT **[Milne & Witten 2008]**
  - MSNBC **[Cucerzan 2007]**
  - IITB **[Kulkarni et al. 2010]** (web data)
  - Twitter **[Meij et al. 2012]**

# **A Framework for Benchmarking Entity-Annotation Systems**

**[Cornolti et al. 2013]**

- Main findings
  - different systems perform well in different scenarios
  - AIDA and TagMe seem to be the winners overall

# **GERBIL – General Entity Annotator Benchmarking Framework**

**[Usbeck et al. 2015]**

- REST-based platform for entity linking using uniform measuring approaches
  - Systems: Wikipedia Miner, Illinois Wikifier, DBpedia Spotlight, TagMe 2, KEA, WAT, AGDISTIS, Babelfy, NERD-ML, AIDA, entityclassifier.eu, FREME e-Entity
  - Data sets: AIDA/CoNLL, AQUAINT, DBpedia Spotlight, IITB, KORE50, MSNBC, Microposts, N3-RSS-500, N3-Reuters-128, OKE 2015

# GERBIL – General Entity Annotator Benchmarking Framework

[Usbeck et al. 2015]

- Provides linkable experiments, e.g.,

| Annotator | Dataset             | Micro F1 | Micro Precision | Micro Recall | Macro F1 | Macro Precision | Macro Recall | Error Count | Timestamp           | GERBIL version |
|-----------|---------------------|----------|-----------------|--------------|----------|-----------------|--------------|-------------|---------------------|----------------|
| AGDISTIS  | ACE2004             | 0.6583   | 0.696           | 0.6245       | 0.7763   | 0.801           | 0.7583       | 0           | 2015-10-14 11:24:23 | 1.1.4          |
| AGDISTIS  | AIDA/CoNLL-Complete | 0.6027   | 0.6513          | 0.5608       | 0.5697   | 0.6355          | 0.5311       | 0           | 2015-10-16 16:11:17 | 1.1.4          |
| AGDISTIS  | AIDA/CoNLL-Test A   | 0.5906   | 0.6393          | 0.5487       | 0.5336   | 0.5912          | 0.4981       | 0           | 2015-10-16 15:59:50 | 1.1.4          |
| AGDISTIS  | AIDA/CoNLL-Test B   | 0.5832   | 0.63            | 0.5429       | 0.5803   | 0.6442          | 0.5449       | 0           | 2015-10-16 15:59:57 | 1.1.4          |
| AGDISTIS  | AIDA/CoNLL-Training | 0.6105   | 0.6595          | 0.5682       | 0.5753   | 0.6434          | 0.5353       | 0           | 2015-10-16 16:06:43 | 1.1.4          |

# DIY Entity Linking – footnotes

- ClueWeb annotated with Freebase (FACC1)
- wiki-links
- Dictionaries for Linking Text, Entities and Ideas

| concept: “soccer”    |                |                |  |
|----------------------|----------------|----------------|--|
| football <i>and</i>  |                |                |  |
| Football             |                |                |  |
| Soccer <i>and</i>    |                |                |  |
| soccer               |                |                |  |
| Association football | サッカー           | piłkarz        |  |
| fútbol <i>and</i>    | 축구             | voetbalclub    |  |
| Fútbol               | footballeur    | អុំពល          |  |
| footballer           | Fußballspieler | bóng đá        |  |
| Futbol <i>and</i>    | sepak bola     | voetbal        |  |
| futbol               | 足球             | Foutbaal       |  |
| Fußball              | فوتبال         | futebolista    |  |
| futebol              | футболист      | لعبة كرة القدم |  |
| futbolista           | כדורגל         | fotbal         |  |

# Outline

- Entity Linking
  - introduction
  - methods
  - evaluation