Tools to Support Textual Inference

Mark Sammons
University of Illinois, Urbana-Champaign
mssammon@illinois.edu
http://cogcomp.cs.illinois.edu
What is at the heart of “big” NLP apps?

- Information Retrieval
- Question Answering
- Translation
- Information Extraction
- Summarization
- Recognizing Textual Entailment
- ...All require comparison of spans of text to determine whether they “match” in some way
Recognizing Textual Entailment**

** “Local Textual Inference” (Zaenen et al., Manning)

Operational definition for Text Understanding:

Given two text fragments (a Text T and a Hypothesis H), T entails H if the meaning of H can be inferred from the meaning of T, as would typically be interpreted by people.

Can frame many NLP tasks as RTE:

- **IE:** Formulate relation as short sentence with generic placeholders, e.g. “Work-For” becomes “An organization employs a person.” -- the Hypothesis; Document paragraphs become Texts.

- **QA:** many questions can be rephrased as statements with generic placeholders: “Something is the fastest car in the world.”

- **Summarization:** Detect novelty of new text span by determining whether current summary entails it or not.
Hurricane Katrina petroleum-supply outlook improved somewhat, yesterday, as U.S. and European governments finally reached a consensus. They finally made up their minds to release 2 million barrels a day, of oil and refined products, from their reserves.

Offers by individual European governments involved supplies of crude or refined oil products.
Hurricane Katrina petroleum-supply outlook improved somewhat, yesterday, as U.S. and European governments finally reached a consensus. U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

Offers by individual European governments involved supplies of crude or refined oil products.
T: Hurricane Katrina petroleum-supply outlook improved somewhat yesterday, as U.S. and European governments finally reached a consensus.

U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

H: Offers by individual European governments involved supplies of crude or refined oil products.

T: U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

H: Offers by individual European governments involved supplies of crude or refined oil products.
U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

Individual European governments offered supplies of crude or refined oil products.

The diagram shows the process of nominalization promotion, where a verb is replaced with a nominalization in one of its arguments. This requires semantic role labeling (for noun predicates).
T: U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

H: Individual European governments offered supplies of crude or refined oil products.

T: U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

H: Individual European governments supplied crude or refined oil products.
U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

Individual European governments supplied crude or refined oil products.

T: U.S. and European governments finally released 2 million barrels a day, of oil and refined products, from their reserves.

H: Individual European governments supplied crude or refined oil products.

‘decided’ (almost) does not change the meaning of the embedded verb.

But what if the embedding verb had been ‘refused’?

ENTAILMENT SHOULD NOT SUCCEED

T: U.S. and European governments finally decided to release 2 million barrels a day, of oil and refined products, from their reserves.

H: Individual European governments supplied crude or refined oil products.
T: U.S. and European governments finally released 2 million barrels a day, of oil and refined products, from their reserves.

H: Individual European governments supplied crude or refined oil products.
Overview

- Common Sub-tasks in Textual Inference
- Recognizing Concepts
- Recognizing Structure Connecting Concepts
- Recognizing Relations between Concepts
- An exercise in Applied Textual Inference: Recognizing Textual Entailment
Overview

- Common Sub-tasks in Textual Inference
- Recognizing Concepts
- Recognizing Structure Connecting Concepts
- Recognizing Relations between Concepts
- An exercise in Applied Textual Inference: Recognizing Textual Entailment
Recognizing Concepts

- Standard unsupervised approaches:
  - TFIDF
  - Multi-Word Expression recognition via co-occurrence statistics
  - Give boundaries, but not types
  - Moderate precision, good coverage

- Supervised approaches
  - Shallow parsing
  - Named Entity Recognition
  - Focused type information at the cost of coverage; annotation expense

- Given some kind of structured reference collection, can we learn a good concept recognizer?
<table>
<thead>
<tr>
<th>“Wikification”: Organizing knowledge</th>
</tr>
</thead>
<tbody>
<tr>
<td>It’s a version of <strong>Chicago</strong> – the standard classic Macintosh menu font, with that distinctive thick diagonal in the ”N”.</td>
</tr>
<tr>
<td><strong>Chicago</strong> was used by default for Mac menus through MacOS 7.6, and OS 8 was released mid-1997.</td>
</tr>
<tr>
<td><strong>Chicago VIII</strong> was one of the early 70s-era <strong>Chicago</strong> albums to catch my ear, along with <strong>Chicago II</strong>.</td>
</tr>
</tbody>
</table>
Cross-document co-reference resolution

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>It’s a version of <strong>Chicago</strong> – the standard classic <strong>Macintosh</strong> menu font, with that distinctive thick diagonal in the ”N”.</td>
<td><strong>Chicago</strong> was used by default for <strong>Mac</strong> menus through <strong>MacOS 7.6</strong>, and <strong>OS 8</strong> was released mid-1997.</td>
<td><strong>Chicago VIII</strong> was one of the early 70s-era <strong>Chicago</strong> albums to catch my ear, along with <strong>Chicago II</strong>.</td>
</tr>
</tbody>
</table>
It’s a version of [Chicago](https://en.wikipedia.org/wiki/Chicago_(font)) – the standard classic [Macintosh](https://en.wikipedia.org/wiki/Macintosh) menu font, with that distinctive thick diagonal in the "N".

[Chicago](https://en.wikipedia.org/wiki/Chicago_(font)) was used by default for [Mac](https://en.wikipedia.org/wiki/Macintosh) menus through [MacOS 7.6](https://en.wikipedia.org/wiki/MacOS_7.x), and [OS 8](https://en.wikipedia.org/wiki/MacOS_8) was released mid-1997.

[Chicago VIII](https://en.wikipedia.org/wiki/Chicago) was one of the early 70s-era [Chicago](https://en.wikipedia.org/wiki/Chicago_(font)) albums to catch my ear, along with [Chicago II](https://en.wikipedia.org/wiki/Chicago_(album)).
The “reference” collection has structure

It’s a version of **Chicago** – the standard classic **Macintosh** menu font, with that distinctive thick diagonal in the ”N”.

**Chicago** was used by default for **Mac** menus through **MacOS 7.6**, and **OS 8** was released mid-1997..

**Chicago VIII** was one of the early 70s-era **Chicago** albums to catch my ear, along with **Chicago II**.
Analysis of Information Networks

It's a version of Chicago – the standard classic Macintosh menu font, with that distinctive thick diagonal in the "N". Chicago was used by default for Mac menus through MacOS 7.6, and OS 8 was released mid-1997. Chicago VIII was one of the early 70s-era Chicago albums to catch my ear, along with Chicago II.
# Performance

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Baseline</th>
<th>Baseline+ Lexical</th>
<th>Baseline+ Lexical+ Global</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACE</td>
<td>94.05</td>
<td>96.21</td>
<td>97.83</td>
</tr>
<tr>
<td>MSN News</td>
<td>81.91</td>
<td>85.10</td>
<td>87.02</td>
</tr>
<tr>
<td>AQUAINT</td>
<td>93.19</td>
<td>95.57</td>
<td>94.38</td>
</tr>
<tr>
<td>Wikipedia Test</td>
<td>85.88</td>
<td>93.59</td>
<td>94.18</td>
</tr>
</tbody>
</table>
 Wikifier Summary

- **Broad spectrum “concept” recognizer**
  - Complements NER
  - Good anecdotal performance on unseen data
  - …without the annotation overhead

- **Context sensitive mutual disambiguation**
  - First-cut non-anaphoric co-reference capability – in a very broad domain

- A good start for bootstrapping NLP in a new domain
  - E.g. recognizing “mentions” of concepts that are/should be(?) in some ontology

- **Real-time web demo:**
  
  http://cogcomp.cs.illinois.edu/demo/wikify/
Overview

- Common Sub-tasks in Textual Inference
- Recognizing Concepts
- Recognizing Structure Connecting Concepts
- Recognizing Relations between Concepts
- An exercise in Applied Textual Inference: Recognizing Textual Entailment
Recognizing Structure Linking Concepts

- **Goal:** broad coverage tools giving coarse sentence structure with some semantic annotation
  - Intra-sentence: Semantic Role Labeling
  - Inter- and intra-sentence: Co-reference

- **Philosophy:** integrate statistical models with domain-specific constraints
  - Local decisions made by machine-learned classifiers
  - Global decision reached by optimizing local decisions with respect to constraints
  - Chosen formalism: Integer Linear Programming
Semantic Role Labeling

- Real-time web demo:
  [http://cogcomp.cs.illinois.edu/demo/srl/](http://cogcomp.cs.illinois.edu/demo/srl/)
Co-reference

- Real-time web demo:
  http://cogcomp.cs.illinois.edu/demo/coref/
Overview

- Common Sub-tasks in Textual Inference
- Recognizing Concepts
- Recognizing Structure Connecting Concepts
- Recognizing Relations between Concepts
- An exercise in Applied Textual Inference: Recognizing Textual Entailment
Required Capabilities

- In applications requiring textual inference, we often need to know when two terms are substitutable in some way:

  **T**: John Smith met *Mel Gibson* yesterday.
  **H**: John Smith met an *actor* yesterday.

  **T**: An earthquake strikes *Taiwan*.
  **H**: An earthquake strikes *Japan*.
Similarity vs. Substitutability

- Similarity measures, e.g. distributional similarity metrics, identify relatedness of terms...
- ...but don’t tell you how the terms are related

\[ T: \text{An earthquake strikes Taiwan.} \]
\[ H: \text{An earthquake strikes Japan.} \]

\[ T: \text{An earthquake strikes Honshu.} \]
\[ H: \text{An earthquake strikes Japan.} \]

- We need specialized resources to make these finer distinctions.
So you want to compare some text....

- **How similar are two lexical expressions?**
  - Depends on what they are
  - String edit distance is usually a weak measure
  - ... think about coreference resolution...

<table>
<thead>
<tr>
<th>String 1</th>
<th>String 2</th>
<th>Norm. edit sim.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shiite</td>
<td>Shi’ ‘ite</td>
<td>0.667</td>
</tr>
<tr>
<td>Mr. Smith</td>
<td>Mrs. Smith</td>
<td>0.900</td>
</tr>
<tr>
<td>Wilbur T. Gobsmack</td>
<td>Mr. Gobsmack</td>
<td>0.611</td>
</tr>
<tr>
<td>Frigid</td>
<td>Cold</td>
<td>0.167</td>
</tr>
<tr>
<td>Wealth</td>
<td>Wreath</td>
<td>0.667</td>
</tr>
<tr>
<td>Paris</td>
<td>France</td>
<td>0.167</td>
</tr>
</tbody>
</table>

- **Solution:** specialized metrics
NESim

- Set of entity-type-specific measures
  - Acronyms, Prefix/Title rules, distance metric
- Score reflects similarity based on type information
- Score is asymmetric

<table>
<thead>
<tr>
<th>String 1</th>
<th>String 2</th>
<th>Norm. edit distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shiite</td>
<td>Shi’ ‘ite</td>
<td>0.922</td>
</tr>
<tr>
<td>Joan Smith</td>
<td>John Smith</td>
<td>0</td>
</tr>
<tr>
<td>Wilbur T. Gobsmack</td>
<td>Mr. Gobsmack</td>
<td>0.95</td>
</tr>
<tr>
<td>Frigid</td>
<td>Cold</td>
<td>0</td>
</tr>
<tr>
<td>Wealth</td>
<td>Wreath</td>
<td>0.900</td>
</tr>
<tr>
<td>Paris</td>
<td>France</td>
<td>0.411</td>
</tr>
</tbody>
</table>
Broad-spectrum ontologies exist!

- Simple approach: determine relations between concepts using static resources
  - WordNet, VerbNet
  - Some clever integration of e.g. WordNet + Wikipedia (YAGO)
  - Some clever “growth” of resources, e.g. Extended WordNet (Snow et al. 06, …)

- …but there are problems:
  - Noisy (low precision)
  - Limited coverage (low recall)
  - Ontology/occurrence mismatch (e.g. Camry Vs. Toyota Camry)
WNSim

- Generate table mapping terms linked in WordNet ontology
  - Synonymy, Hypernymy, Meronymy
- Score reflects distance (up to 3 edges, undirected – e.g. via lowest common subsumer)
- Score is symmetric

<table>
<thead>
<tr>
<th>String 1</th>
<th>String 2</th>
<th>WNSim distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shiite</td>
<td>Shi’ite</td>
<td>0</td>
</tr>
<tr>
<td>Mr. Smith</td>
<td>Mrs. Smith</td>
<td>0</td>
</tr>
<tr>
<td>Wilbur T. Gobsmack</td>
<td>Mr. Gobsmack</td>
<td>0</td>
</tr>
<tr>
<td>Frigid</td>
<td>Cold</td>
<td>1</td>
</tr>
<tr>
<td>Wealth</td>
<td>Wreath</td>
<td>0</td>
</tr>
<tr>
<td>Paris</td>
<td>France</td>
<td>0</td>
</tr>
</tbody>
</table>
Taxonomic Relation Classifier (TAREC): On-demand Ontological Relations

- In textual inference, ontologies are useful to identify relations between concepts – typically, to determine whether two concepts are substitutable.

- The functionality we need is, given two candidate concepts X and Y, to determine whether
  - X is substitutable for Y
  - X is definitely not substitutable for Y (direct evidence *against* a match)
  - X is not related to Y (but no direct evidence against a match)
## Basic Relations

<table>
<thead>
<tr>
<th>Relation</th>
<th>Meaning</th>
<th>$x$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x \leftarrow y$</td>
<td>ancestor</td>
<td>actor</td>
<td>Mel Gibson</td>
</tr>
<tr>
<td>$x \rightarrow y$</td>
<td>child</td>
<td>Makalu</td>
<td>mountain</td>
</tr>
<tr>
<td>$x \leftrightarrow y$</td>
<td>sibling</td>
<td>copper</td>
<td>oxygen</td>
</tr>
<tr>
<td>$x \leftrightarrow y$</td>
<td>none</td>
<td>egg</td>
<td>C++</td>
</tr>
</tbody>
</table>
Taxonomic Relation Classifier (TAREC)

- Normalize query terms to reference collection
  - Use pattern-based extraction + web search to identify alternative terms (e.g., delimiter-based list extraction)
- Train a local classifier to compare query terms
  - Mine Wikipedia for related terms: article titles, content, and categories
- PMI: \( \text{pmi}(x,y) = \log \left[ \frac{Nf(x,y)}{f(x)f(y)} \right] \)
  where \( f(.) \) counts the # of its argument; \( N \) is the total # of Wikipedia pages.

<table>
<thead>
<tr>
<th>Bag of words - Degree of similarity</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{texts}(x) ) vs. ( \text{categories}(y) )</td>
</tr>
<tr>
<td>( \text{categories}(x) ) vs. ( \text{texts}(y) )</td>
</tr>
</tbody>
</table>
Improving Decisions with Constraints

- Improve local classifier by using concepts related to query terms X, Y to constrain them
  - Extract related terms from static ontology (YAGO)
  - Use local classifier to determine relations between them
  - Select best set of relation labels linking X, Y and other concepts that does not match a pre-specified violation pattern (e.g. b, d)
### Performance

<table>
<thead>
<tr>
<th>System</th>
<th>Wiki</th>
<th>WordNet</th>
<th>non-Wiki</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strube07</td>
<td>24.59</td>
<td>24.13</td>
<td>21.18</td>
</tr>
<tr>
<td>Snow06</td>
<td>41.23</td>
<td>46.91</td>
<td>34.46</td>
</tr>
<tr>
<td>Yago07</td>
<td>69.95</td>
<td>70.42</td>
<td>34.26</td>
</tr>
<tr>
<td>TAREC (local)</td>
<td>89.37</td>
<td>89.72</td>
<td>31.22</td>
</tr>
<tr>
<td>TAREC</td>
<td>91.03</td>
<td>91.2</td>
<td>45.21</td>
</tr>
</tbody>
</table>

- **Limitations:** Useful for Things rather than Relations
  - Majority of Wikipedia pages are about entity-like concepts
  - Need to supplement with additional knowledge for textual inference
TAREC summary

- Broad spectrum ontology-like resource
- Functional interface matched to typical inference need
- Leverages Wikipedia as reference collection
  - Dynamic resource – regular updates
- Normalizes input terms to reference “ontology”
- Uses local classification plus constrained optimization to incorporate common-sense constraints

Real-time web demo:
http://cogcomp.cs.illinois.edu/demo/relation/
TEXTUAL ENTAILMENT SYSTEM
Alignment in RTE: Lexical Level

- **Alignment**: a mapping from elements in the Hypothesis to elements in the Text

![Diagram showing alignment between John Smith bought three cakes and two oranges and John bought two oranges]
Alignment is Useful for Machine Learning in RTE

- **Machine Learning** approaches provide much-needed robustness for NLP tasks
- **RTE data sets** are small, given complexity of problem
- **Global, 2- or 3-class label** on each pair
- We would like to resolve entailment by **combining local decisions** (e.g. word-level, phrase level); but *which* decisions?
- **Alignment** can be used to select a subset of the many possible comparisons, and thereby **augments global label with (proxy for) finer-grained structure**; can be used...
  - ...to determine active features
  - ...to generate labels for local classifiers
Multiple alignments at multiple granularities

- Intuition: exploit differences/agreements between different views of the entailment pair; avoid canonization
- Accommodates analysis at different granularities
- Resources with comparable scores can compete with each other – pick the “best”
  - e.g. Words, Multi-word Expressions, Phrasal Verbs
- Unscaled resources occupy different alignments (SRL, NE)
- Metrics can return negative numbers; use magnitude in alignments, preserve negative edge label
  - May be useful for contradiction features
## Multiple Alignments for RTE

<table>
<thead>
<tr>
<th>NE</th>
<th>[PER] John Smith</th>
<th>[PER] Jane</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUM</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>#:3 [unit: cake]</td>
<td>#:2 [unit: orange]</td>
</tr>
<tr>
<td>SRL</td>
<td>say</td>
<td>buy</td>
</tr>
<tr>
<td></td>
<td>John Smith</td>
<td>Jane</td>
</tr>
<tr>
<td></td>
<td></td>
<td>three cakes</td>
</tr>
</tbody>
</table>

**T:** John Smith said Jane bought three cakes and two oranges

**H:** Jane bought three oranges

**SRL**

<table>
<thead>
<tr>
<th>Jane</th>
<th>buy</th>
<th>three oranges</th>
</tr>
</thead>
</table>

**NUM**

<table>
<thead>
<tr>
<th>[PER] Jane</th>
<th></th>
</tr>
</thead>
</table>

| [#:3][unit: orange] |

**NE**
Learning from Multiple Alignments

- Extract features based on individual alignments
  - Can use high-precision, low-recall resources as filter features
  - Typical match features within alignments – e.g. proportion of tokens matched

- Extract features based on agreement, disagreement between different alignments
  - E.g. Predicate-Argument, Numerical Quantities

- Allows graceful degradation if some resources are unreliable; learner assigns low weights to corresponding features
Multiple Alignments ctd.

- Model each alignment as optimization problem
  - Penalize distant mappings of neighboring constituents in H, T (proxy for deep structure – favor chunk alignment)
  - Constraints: each token in H can be covered exactly once by an aligned constituent; edge scores must account for number of constituents covered
  - Solve by brute-force search

\[
\frac{1}{m} \left[ \sum e(H_i, T_j) + \alpha \sum \Delta(e(H_i, T_j), e(H_{i+1}, T_k)) \right] \\
\sum_j I[e(H_i, T_j)] \leq 1
\]
Feature Extraction

- Main types of features:
  - Features assessing quality of alignment in a given view
  - Features assessing agreement between views

- Quality of Alignment features:
  - Proportion of constituents matched in Word, NE, SRL views
  - “Distortion” of match pattern

- Agreement features:
  - Proportion of token alignments agreeing with SRL constituent alignments
  - Negation of predicate in SRL relation match

- Extension: Using Coreference:
  - Augment SRL predicates: add arguments using Coref chains
  - Introduces inter-sentence structure
Results

<table>
<thead>
<tr>
<th>Corpus</th>
<th>System</th>
<th>RTE5 Dev</th>
<th>RTE5 Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Baseline</td>
<td>0.628</td>
<td>0.600</td>
</tr>
<tr>
<td></td>
<td>No NE*</td>
<td>0.640</td>
<td>0.629</td>
</tr>
<tr>
<td></td>
<td>Basic NE</td>
<td>0.623</td>
<td>0.633</td>
</tr>
<tr>
<td></td>
<td>No WN</td>
<td>0.647</td>
<td>0.603</td>
</tr>
<tr>
<td></td>
<td>All*</td>
<td>0.648</td>
<td>0.644</td>
</tr>
<tr>
<td></td>
<td>All + Coref</td>
<td>0.663</td>
<td>0.666</td>
</tr>
</tbody>
</table>

* Submitted runs had ~60 buggy alignments in dev test; results using non-buggy alignments shown here
RTE system demo

- Note: Where are the transformations?
  - We found that chaining offered little gain while significantly complicating the architecture
  - We use transformation rules as mappings between predicate-argument structures in the SRL Comparator

- Real-time web demo:
  http://cogcomp.cs.illinois.edu/cgi-bin/rte/entailment.py
Can we do better?

- Presently, we heuristically align our representations of text and hypothesis to reduce the problem complexity and make learning tractable.
- Even if we use machine learning for alignment, a pipeline architecture leads to error propagation.
- Alternative: "indirect supervision".
  - Specify space of alignments, and a feature-based representation for it.
  - Use binary RTE labels to optimize alignment that gives best performance on binary task.
- A way to learn "purposefulness"?
Chang et al. 2010

- Apply indirect supervision approach to RTE and other tasks
- Use unified graph based on same input representation as fixed alignment system
- Specify match features for nodes (based on similarity score), edges, and node deletion
- Specify constraints on matching edges
  - Edge can only match if source/sink nodes are also matched
- Goal:
  - learn weights on node/edge match features such that...
  - The highest-scoring alignments for entailment pairs...
  - Yield maximum performance when used to decide binary entailment label (using threshold)
Indirect Supervision for RTE (cont’d)

- Optimization for alignment: needs a key insight
  - The *best* alignment for a negative example is “not good enough” (maximum alignment-based score should be low)
  - A positive entailment example has *at least one good alignment* (maximum alignment-based score exceeds some threshold)

- Procedure: for each example
  - Find best alignment using current hypothesis
  - Predict entailment label
  - If prediction is incorrect, update alignment feature weights

- Results: comparable to two-stage architecture
Summary

- We take a compositional approach to textual inference
  - Multi-view representation/architecture
  - Annotator/comparator pairing

- We are trying to build components that isolate specific knowledge domains, but are easy to use
  - Simple functional interface (metrics)
  - Goal: consistent API

- We are using Wikipedia as a broad-coverage general knowledge resource
  - Developed Wikifier, TAREC
  - Currently, trying to integrate them with NLP tools like Co-reference Resolver and RTE system

- Many of our tools have live demos; many are available…
References

- Annie Zaenen, Lauri Kartunnen, and Richard Crouch, “Local textual inference: can it be defined or circumscribed?”, EMSEE (2005)

- Christopher D. Manning, “Local Textual Inference: It's hard to circumscribe, but you know it when you see it - and NLP needs it.”, MS, Stanford University (2006)

- Quang Do and Dan Roth, “Constraints based Taxonomic Relation Classification”, EMNLP (2010)

- Lev Ratinov, Doug Downey, Mike Anderson, Dan Roth, “Local and global algorithms for disambiguation to Wikipedia”, ACL (2011)

E. Bengtson and D. Roth, “Understanding the value of features for coreference resolution”, *EMNLP* (2008)

Q. Do, D. Roth, M. Sammons, Y. Tu, V. Vydiswaran, “Robust, light-weight approaches to compute lexical similarity”, *Computer Science Research and Technical Reports, University of Illinois* (2009)