COREFERENGE RESOLUTION

Heng Ji

jih@rpi.edu
March 26, 2014

Acknowledgement: some slides are from Vincent Ng
Introduction

- Supervised Coreference Overview
  - Models
  - Features

- Advanced Techniques and Trends
  - Clustering
  - Incorporating World Knowledge
  - Incorporating Syntactic Features
  - Non-referential Pronouns
  - Rule-based system
But the little prince could not restrain admiration:

"Oh! How beautiful you are!"

"Am I not?" the flower responded, sweetly. "And I was born at the same moment as the sun . . ."

The little prince could guess easily enough that she was not any too modest--but how moving--and exciting--she was!

"I think it is time for breakfast," she added an instant later. "If you would have the kindness to think of my needs--"

And the little prince, completely abashed, went to look for a sprinkling-can of fresh water. So, he tended the flower.
## Modeling Coreference Resolution

### Detailed Survey and Comparison in (Ng, 2010)

<table>
<thead>
<tr>
<th>Method</th>
<th>References</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mention-Pair model</td>
<td>Classify whether two mentions are coreferential or not + clustering</td>
<td>(Soon et al. 2001; Ng and Cardie 2002; Ji et al., 2005; McCallum &amp; Wellner, 2004; Nicolae &amp; Nicolae, 2006)</td>
<td>easy to encode features</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>greedy clustering algorithm; Each candidate antecedents is considered independently of the other</td>
</tr>
<tr>
<td>Entity-Mention Model</td>
<td>Classify whether a mention and a preceding, possibly partially formed cluster are coreferential or not</td>
<td>Pasula et al. 2003 ; Luo et al. 2004; Yang et al. 2004, 2008; Daume &amp; Marcu, 2005; Culotta et al., 2007</td>
<td>Improved expressiveness, allows cluster level features</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Each candidate cluster is considered independently of the others</td>
</tr>
<tr>
<td>Mention-Ranking Model</td>
<td>Imposes a ranking on a set of candidate antecedents</td>
<td>Denis &amp; Baldridge 2007, 2008</td>
<td>Considers all the candidate antecedents simultaneously</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Insufficient information to make an informed coreference decision; still need to do clustering</td>
</tr>
<tr>
<td>Cluster Ranking model</td>
<td>Ranks all the preceding clusters for each mention; create instances with entity-mention model, rank instances with mention-ranking model</td>
<td>Rahman and Ng, 2009</td>
<td>Combines the strength of previous models; Achieved the best performance</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-</td>
</tr>
</tbody>
</table>
Really Greedy Clustering Algorithms

- Single-link clustering (Soon et al., 2001)
  - For each NP<sub>j</sub>, select as its antecedent the closest preceding NP that is determined as coreferent with it
  - Posit NP<sub>j</sub> as non-anaphoric if no preceding NP is coreferent with it

- Best-first clustering (Ng & Cardie, 2002)
  - Same as single-link clustering, except that we select as the antecedent the NP that has the highest coreference likelihood
Why are they really greedy?

- Clusters are formed based on a small subset of the pairwise coreference decisions
  - Many pairwise decisions are not used in the clustering process

---

Mr. Clinton Clinton she
Why are they really greedy?

- Clusters are formed based on a small subset of the pairwise coreference decisions
  - Many pairwise decisions are not used in the clustering process
Why are they really greedy?

- Clusters are formed based on a small subset of the pairwise coreference decisions
  - Many pairwise decisions are not used in the clustering process
Less Greedy Clustering Algorithms

- Use all the pairwise coreference decisions

- **Graph partitioning algorithms**
  - each text is represented as a graph
    - each vertex corresponds to a NP; edge weight is coref likelihood
  - Goal: partition the graph nodes to form coreference clusters
Less Greedy Clustering Algorithms

- Use all the pairwise coreference decisions

**Graph partitioning algorithms**
- each text is represented as a graph
  - each vertex corresponds to a NP; weight of an edge indicates the likelihood that the two NPs are coreferent
  - Goal: partition the graph nodes to form coreference clusters

- **Correlation clustering** (e.g., McCallum & Wellner (2004))
  - cluster that respects as many pairwise decisions as possible

- **Minimum-cut-based clustering** (Nicolae & Nicolae, 2006)
  - Find the mincut of the graph and partition the graph nodes; repeat until some stopping criterion is reached
The American Medical Association voted yesterday to install the heir apparent as its president-elect, rejecting a strong, upstart challenge …
Which clustering algorithm is the best?

- Few empirical comparisons
- Luo et al. (2004) didn’t compare their Bell-tree approach against the really greedy algorithms
Which clustering algorithm is the best?

- Few empirical comparisons

- Luo et al. (2004) didn’t compare their Bell-tree approach against the really greedy algorithms
  - Klein (2005, pc): search space is too large, need to apply a lot of heuristics to prune the space, making it a greedy algorithm
Which clustering algorithm is the best?

- Few empirical comparisons

- Luo et al. (2004) didn’t compare their Bell-tree approach against the really greedy algorithms
  - Klein (2005, pc): search space is too large, need to apply a lot of heuristics to prune the space, making it a greedy algorithm
  - Nicolae & Nicolae (2006): not much difference in performance between Bell tree clustering and the really greedy algorithms
Supervised Coreference (Recap)

- **Step 1**: Learn a coreference model

- **Step 2**: Apply a clustering algorithm
Supervised Coreference (Recap)

- **Step 1**: Learn a coreference model
  - Mention-pair model

- **Step 2**: Apply a clustering algorithm
  - Really greedy algorithms
  - Less greedy algorithms
  - Time-aware algorithms
Weaknesses of the Mention-Pair Model

- **Limited expressiveness**
  - information extracted from two NPs may not be sufficient for making an informed coreference decision

- **Can’t determine which candidate antecedent is the best**
  - only determine how good a candidate is relative to NP to be resolved, not how good it is relative to the others
Weaknesses of the Mention-Pair Model

- **Limited expressiveness**
  - information extracted from two NPs may not be sufficient for making an informed coreference decision

- **Can’t determine which candidate antecedent is the best**
  - only determine how good a candidate is relative to NP to be resolved, not how good it is relative to the others
Improving Model Expressiveness

- Want a coreference model that can tell us how likely “she” and a preceding cluster of “she” are coreferent.
The Entity-Mention Model

- a classifier that determines whether (or how likely) an NP belongs to a preceding coreference cluster

- more **expressive** than the mention-pair model
  - can employ **cluster-level** features defined over any subset of NPs in a preceding cluster

- addresses the expressiveness problem

Pasula et al. (2003), Luo et al. (2004), Yang et al. (2004, 2008), Daume & Marcu (2005), Culotta et al. (2007), …
Weaknesses of the Mention-Pair Model

- **Limited expressiveness**
  - information extracted from two NPs may not be sufficient for making an informed coreference decision

- **Can’t determine which candidate antecedent is the best**
  - only determine how good a candidate is relative to NP to be resolved, not how good it is relative to the others
How to address this problem?

- Idea: train a model that imposes a **ranking** on the candidate antecedents for an NP to be resolved
  - so that it assigns the highest rank to the correct antecedent
How to address this problem?

- Idea: train a model that imposes a **ranking** on the candidate antecedents for an NP to be resolved
  - so that it assigns the highest rank to the correct antecedent

- A ranker allows all candidate antecedents to be considered simultaneously and captures competition among them
  - allows us find the best candidate antecedent for an NP

- There is a natural resolution strategy for a ranking model
  - An NP is resolved to the highest-ranked candidate antecedent
How to train a ranking model?

- Convert the problem of ranking $m$ NPs into the a set of pairwise ranking problems
  - Each pairwise ranking problem involves determining which of two candidate antecedents is better for an NP to be resolved
    - Each one is essentially a classification problem
How to train a ranking model?

- Convert the problem of ranking $m$ NPs into the a set of pairwise ranking problems
  - Each pairwise ranking problem involves determining which of two candidate antecedents is better for an NP to be resolved
    - Each one is essentially a classification problem

- First supervised coreference model: Connolly et al. (1994)
  - Train a decision tree to determine which of the two candidate antecedents of an NP is more likely to be its antecedent
  - During testing, need to heuristically combine the pairwise ranking results to select an antecedent for each NP
Revival of the Ranking Approach

- The ranking model is theoretically better but far less popular than the mention-pair model in the decade following its proposal.

- Rediscovered almost ten years later independently by:
  - Yang et al. (2003): twin-candidate model
  - Iida et al. (2003): tournament model
The Mention-Ranking Model

- Denis & Baldridge (2007, 2008): train the ranker using maximum entropy
  - model outputs a rank value for each candidate antecedent
  - obviates need to heuristically combine pairwise ranking results
The Mention-Ranking Model

- Denis & Baldridge (2007, 2008): train the ranker using maximum entropy
  - model outputs a rank value for each candidate antecedent
  - obviates need to heuristically combine pairwise ranking results
Caveat

- Since a ranker only imposes a ranking on the candidates, it cannot determine whether an NP is anaphoric
  - Need to train a classifier to determine if an NP is anaphoric
### Recap

<table>
<thead>
<tr>
<th>Problem</th>
<th>Entity Mention</th>
<th>Mention Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limited expressiveness</td>
<td>✔</td>
<td>✗</td>
</tr>
<tr>
<td>Cannot determine best candidate</td>
<td>✗</td>
<td>✔</td>
</tr>
</tbody>
</table>

Can we combine the strengths of these two model?
Consider preceding clusters, not candidate antecedents
Consider preceding clusters, not candidate antecedents.

Rank preceding clusters.

Rank candidate antecedents.
The Cluster-Ranking Model

Mention-ranking model
- Rank candidate antecedents

Entity-mention model
- Consider preceding clusters, not candidate antecedents
- Rank preceding clusters
The Cluster-Ranking Model (Rahman & Ng, 2009)

**Training**
- train a *ranker* to rank preceding clusters

**Testing**
- resolve each NP to the highest-ranked preceding cluster
The Cluster-Ranking Model (Rahman & Ng, 2009)

- **Training**
  - train a *ranker* to rank preceding clusters

- **Testing**
  - resolve each NP to the highest-ranked preceding cluster

Lappin & Leass’s (1994) heuristic pronoun resolver
The Cluster-Ranking Model (Rahman & Ng, 2009)

- As a ranker, the cluster-ranking model cannot determine whether an NP is anaphoric
  - Before resolving an NP, we still need to use an anaphoricity classifier to determine if it is anaphoric
    - yields a pipeline architecture

- Potential problem
  - errors made by the anaphoricity classifier will be propagated to the coreference resolver

- Solution
  - joint learning for anaphoricity and coreference resolution
Some Empirical Results on ACE 2005

<table>
<thead>
<tr>
<th>Model</th>
<th>B³</th>
<th>CEAF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Mention-Pair Baseline</td>
<td>50.8</td>
<td>57.9</td>
</tr>
<tr>
<td>Entity-Mention Baseline</td>
<td>51.2</td>
<td>57.8</td>
</tr>
<tr>
<td>Mention-Ranking Baseline (Pipeline)</td>
<td>52.3</td>
<td>61.8</td>
</tr>
<tr>
<td>Mention-Ranking Baseline (Joint)</td>
<td>50.4</td>
<td>65.5</td>
</tr>
<tr>
<td>Cluster-Ranking Model (Pipeline)</td>
<td>55.3</td>
<td>63.7</td>
</tr>
<tr>
<td>Cluster-Ranking Model (Joint)</td>
<td>54.4</td>
<td>70.5</td>
</tr>
</tbody>
</table>
Some Empirical Results on ACE 2005

<table>
<thead>
<tr>
<th></th>
<th>$B^3$</th>
<th>CEAF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Mention-Pair Baseline</td>
<td>50.8</td>
<td>57.9</td>
</tr>
<tr>
<td>Entity-Mention Baseline</td>
<td>51.2</td>
<td>57.8</td>
</tr>
<tr>
<td>Mention-Ranking Baseline (Pipeline)</td>
<td>52.3</td>
<td>61.8</td>
</tr>
<tr>
<td>Mention-Ranking Baseline (Joint)</td>
<td>50.4</td>
<td>65.5</td>
</tr>
<tr>
<td>Cluster-Ranking Model (Pipeline)</td>
<td>55.3</td>
<td>63.7</td>
</tr>
<tr>
<td>Cluster-Ranking Model (Joint)</td>
<td>54.4</td>
<td>70.5</td>
</tr>
</tbody>
</table>

- Cluster ranking is better than mention ranking, which in turn is better than the entity-mention model and the mention-pair model.
- Joint models perform better than pipeline models.
Three Sources of World Knowledge

1. Online encyclopedia and lexical knowledge bases
   - YAGO
   - FrameNet

2. Coreference-annotated data

3. Unannotated data
Three Sources of World Knowledge

1. Online encyclopedia and lexical knowledge bases
   - YAGO
   - FrameNet

2. Coreference-annotated data

3. Unannotated data
YAGO (Suchanek et al., 2007)

- contains 5 million facts derived from Wikipedia and WordNet
- each fact is a triple describing a relation between two NPs
  - \(<\text{NP1}, \text{rel}, \text{NP2}>\), rel can be one of 90 YAGO relation types
YAGO (Suchanek et al., 2007)

- contains 5 million facts derived from Wikipedia and WordNet
- each fact is a triple describing a relation between two NPs
  - \(<\text{NP1, rel, NP2}>\), rel can be one of 90 YAGO relation types
- focuses on two types of YAGO relations: TYPE and MEANS
  (Bryl et al., 2010, Uryupina et al., 2011)
YAGO (Suchanek et al., 2007)

- contains 5 million facts derived from Wikipedia and WordNet
- each fact is a triple describing a relation between two NPs
  - \(<\text{NP1, rel, NP2}>\), rel can be one of 90 YAGO relation types
- focuses on two types of YAGO relations: TYPE and MEANS
  (Bryl et al., 2010, Uryupina et al., 2011)
  - TYPE: the IS-A relation
    - \(<\text{AlbertEinstein, TYPE, physicist}>\>
    - \(<\text{BarackObama, TYPE, US president}>\)
YAGO (Suchanek et al., 2007)

- contains 5 million facts derived from Wikipedia and WordNet
- each fact is a triple describing a relation between two NPs
  - \(<\text{NP1, rel, NP2}>\), rel can be one of 90 YAGO relation types
- focuses on two types of YAGO relations: **TYPE** and **MEANS** (Bryl et al., 2010, Uryupina et al., 2011)
  - **TYPE**: the IS-A relation
    - \(<\text{AlbertEinstein, TYPE, physicist}>\>
    - \(<\text{BarackObama, TYPE, US president}>\>
  - **MEANS**: addresses synonymy and ambiguity
    - \(<\text{Einstein, MEANS, AlbertEinstein}>\>
    - \(<\text{Einstein, MEANS, AlfredEinstein}>\>
YAGO (Suchanek et al., 2007)

- contains 5 million facts derived from Wikipedia and WordNet
- each fact is a triple describing a relation between two NPs
  - `<NP1, rel, NP2>`, rel can be one of 90 YAGO relation types
- focuses on two types of YAGO relations: TYPE and MEANS (Bryl et al., 2010, Uryupina et al., 2011)
  - TYPE: the IS-A relation
    - `<AlbertEinstein, TYPE, physicist>`, `<BarackObama, TYPE, US president>`
  - MEANS: addresses synonymy and ambiguity
    - `<Einstein, MEANS, AlbertEinstein>`, `<Einstein, MEANS, AlfredEinstein>`
- provide evidence that the two NPs involved are coreferent
Why YAGO?

- combines the information in Wikipedia and WordNet
- can resolve the celebrity to Martha Stewart
  - neither Wikipedia nor WordNet alone can
Using YAGO for Coreference Resolution

- create a binary-valued YAGO feature

  - Mention-pair model

- Cluster-ranking model
Using YAGO for Coreference Resolution

- create a binary-valued YAGO feature
  - Mention-pair model
    - determines whether two NPs are coreferent
    - each instance corresponds to two NPs
      \[
      \begin{cases} 
      1 & \text{if the two NPs are in a TYPE or MEANS relation} \\
      0 & \text{otherwise}
      \end{cases}
      \]
  - Cluster-ranking model
Using YAGO for Coreference Resolution

- create a binary-valued YAGO feature
  - Mention-pair model
    - determines whether two NPs are coreferent
    - each instance corresponds to two NPs
      - \[ \begin{align*}
      1 & \text{ if the two NPs are in a TYPE or MEANS relation} \\
      0 & \text{ otherwise}
      \end{align*} \]
  - Cluster-ranking model
    - ranks coreference clusters preceding each NP to be resolved
    - each instance corresponds to \( NP_k \) and a preceding cluster \( c \)
    - features are defined between \( NP_k \) and \( c \)
      - \[ \begin{align*}
      1 & \text{ if } NP_k \text{ and at least 1 NP in } c \text{ are in a TYPE or MEANS relation} \\
      0 & \text{ otherwise}
      \end{align*} \]
Three Sources of World Knowledge

1. Online encyclopedia and lexical knowledge bases
   - YAGO
   - FrameNet

2. Coreference-annotated data

3. Unannotated data
Motivating Example

Peter Anthony *decries* program trading as “limiting the game to a few,” but he is not sure whether he wants to *denounce it* because …
Motivating Example

Peter Anthony *decries* program trading as “limiting the game to a few,” but he is not sure whether he wants to *denounce* it because …

- To resolve *it* to program trading, it may be helpful to know
  1. *it* and program trading have the same semantic role
  2. decry and decounce are “semantically related”
Observation

- Features encoding
  - the semantic roles of the two NPs under consideration
  - whether the associated predicates are “semantically related” could be useful for identifying coreference relations.
Observation

- Features encoding
  - the semantic roles of the two NPs under consideration
  - whether the associated predicates are “semantically related” could be useful for identifying coreference relations.

Use ASSERT
- Provides PropBank-style roles (Arg0, Arg1, …)
Observation

- Features encoding
  - the semantic roles of the two NPs under consideration
  - whether the associated predicates are "semantically related"
    could be useful for identifying coreference relations.

Use ASSERT
- Provides PropBank-style roles (Arg0, Arg1, …)

Use FrameNet
- Checks whether the two predicates appear in the same frame
Observation

- Features encoding
  - the semantic roles of the two NPs under consideration
  - whether the associated predicates are “semantically related”
    could be useful for identifying coreference relations.

Use ASSERT
- Provides PropBank-style roles (Arg0, Arg1, …)

Use FrameNet
- Checks whether the two predicates appear in the same frame
- Consider two verbs related as long as there exists a frame that contains both of them
Features based on FrameNet and ASSERT
Features based on FrameNet and ASSERT

- Assume $NP_j$ and $NP_k$ are the arguments of two predicates

1. Encode knowledge from FrameNet as one of three values
   - The two predicates appear in the same frame
   - Both appear in FrameNet but never in the same frame
   - One or both of them do not appear in FrameNet

2. Encode semantic roles of $NP_j$ and $NP_k$ as one of five values
   - Arg0-Arg0, Arg1-Arg1, Arg0-Arg1, Arg1-Arg0, OTHERS

3. Create 15 binary-valued features by pairing the 3 possible values from FrameNet and 5 possible values from ASSERT
Features based on FrameNet and ASSERT

1. Encode knowledge from FrameNet as one of three values
   - The two predicates appear in the same frame
   - Both appear in FrameNet but never in the same frame
   - One or both of them do not appear in FrameNet

2. Encode semantic roles of $NP_j$ and $NP_k$ as one of five values
   - Arg0-Arg0, Arg1-Arg1, Arg0-Arg1, Arg1-Arg0, OTHERS

3. Create 15 binary-valued features by pairing the 3 possible values from FrameNet and 5 possible values from ASSERT
Features based on FrameNet and ASSERT

- Assume $NP_j$ and $NP_k$ are the arguments of two predicates

1. Encode knowledge from FrameNet as one of three values
   - The two predicates appear in the same frame
   - Both appear in FrameNet but never in the same frame
   - One or both of them do not appear in FrameNet

2. Encode semantic roles of $NP_j$ and $NP_k$ as one of five values
   - Arg0-Arg0, Arg1-Arg1, Arg0-Arg1, Arg1-Arg0, OTHERS

3. Create 15 binary-valued features by pairing the 3 possible values from FrameNet and 5 possible values from ASSERT
Features based on FrameNet and ASSERT

- Assume $NP_j$ and $NP_k$ are the arguments of two predicates

1. Encode knowledge from FrameNet as one of three values
   - The two predicates appear in the same frame
   - Both appear in FrameNet but never in the same frame
   - One or both of them do not appear in FrameNet

2. Encode semantic roles of $NP_j$ and $NP_k$ as one of five values
   - Arg0-Arg0, Arg1-Arg1, Arg0-Arg1, Arg1-Arg0, OTHERS

3. Create 15 binary-valued features by pairing the 3 possible values from FrameNet and 5 possible values from ASSERT
Incorporating Features into Models

- Mention-pair model
  - the 15 features can be employed directly by the mention-pair model, since they are defined on two NPs

- Cluster-ranking model
  - extend their definitions so that they can be computed between an NP and a preceding cluster
Related Work

- No coreference work that employs FrameNet

- But … related to
  - Bean & Riloff’s (2004) use of patterns for inducing domain-specific contextual role knowledge
  - Ponzetto & Strube’s (2006) use of semantic roles for inducing features
Three Sources of World Knowledge

1. Online encyclopedia and lexical knowledge bases
   - YAGO
   - FrameNet

2. Coreference-annotated data

3. Unannotated data
World Knowledge from Annotated Data

- Observation
  - Since world knowledge is needed for coreference resolution, a human annotator must have employed world knowledge when coreference-annotating a document

- Goal
  - Design features that can “recover” such world knowledge
World Knowledge from Annotated Data

• Observation
  - Since world knowledge is needed for coreference resolution, a human annotator must have employed world knowledge when coreference-annotating a document

• Goal
  - Design features that can “recover” such world knowledge

What kind of world knowledge can we extract from annotated data?
World Knowledge from Annotated Data

1. world knowledge for **identifying coreference relations**
   - if Barack Obama and U.S. president appear in the same coreference chain in a training text, we can gather the world knowledge that Barack Obama is a U.S. president
World Knowledge from Annotated Data

1. world knowledge for identifying coreference relations
   - if Barack Obama and U.S. president appear in the same coreference chain in a training text, we can gather the world knowledge that Barack Obama is a U.S. president

2. world knowledge for determining non-coreference
   - infer that a lion and a tiger are unlikely to refer to the same entity after realizing that they never appear in the same coreference chain in the training data
World Knowledge from Annotated Data

1. World knowledge for identifying coreference relations
   - if Barack Obama and U.S. president appear in the same coreference chain in a training text, we can gather the world knowledge that Barack Obama is a U.S. president

2. World knowledge for determining non-coreference
   - infer that a lion and a tiger are unlikely to refer to the same entity after realizing that they never appear in the same coreference chain in the training data
     - features computed based on WordNet distance or distributional similarity may incorrect suggest that the two are coreferent
World Knowledge from Annotated Data

- Observation
  - The NP pairs collected from coreference-annotated training data could be useful features (e.g., <Obama, U.S. president>)
World Knowledge from Annotated Data

• Observation
  • The NP pairs collected from coreference-annotated training data could be useful features (e.g., <Obama, U.S. president>)

• How to compute values for these features?
  • Mention-pair model: feature value is
    \[
    \begin{cases}
    1 & \text{if the feature is composed of the two NPs under consideration} \\
    0 & \text{otherwise}
    \end{cases}
    \]

  • Cluster-ranking model
    • Extend this feature definition so that the feature can be applied to an NP and a preceding cluster
World Knowledge from Annotated Data

- Potential problem
  - **Data sparsity**: many NP pairs in training data may not appear in test data
World Knowledge from Annotated Data

- Potential problem
  - **Data sparsity**: many NP pairs in training data may not appear in test data

- Solution
  - Employ not only the NP pairs as features but also generalized versions of these features. E.g.,
    - replace a named entity by its named entity tag
    - replace a common NP by its head noun
    - …
Any Other Useful Knowledge from Annotated Data?
Any Other Useful Knowledge from Annotated Data?

- Recall that … features encoding
  - the semantic roles of two NPs
  - whether the associated verbs are “semantically related” could be useful features for coreference resolution

- Goal: create variants of these features
Recall that … features encoding
- the semantic roles of two NPs
- whether the associated verbs are “semantically related”
could be useful features for coreference resolution

Goal: create variants of these features
Any Other Useful Knowledge from Annotated Data?

- Recall that … features encoding
  - the semantic roles of two NPs
  - the associated verbs
  could be useful features for coreference resolution

- Goal: create variants of these features
Recall that … features encoding

- the semantic roles of two NPs
- the associated verbs

could be useful features for coreference resolution

Goal: create variants of these features

Each feature is represented by two verbs and the semantic roles
- e.g., <decry, denounce, Arg1-Arg1>
Why would these features be useful for coreference?

- They allow a learner to learn from annotated data whether two NPs serving as the objects of *decry* and *denounce* are likely to be coreferent, for instance.
Three Sources of World Knowledge

1. Online encyclopedia and lexical knowledge bases
   - YAGO
   - FrameNet

2. Coreference-annotated data

3. Unannotated data
World Knowledge from Unannotated Data

- can extract syntactic appositions heuristically
  - shown to be useful for coreference resolution (e.g., Daume & Marcu, 2005, Ng, 2007, Haghighi & Klein, 2009)

- Each extraction is an NP pair. E.g.,
  - <Barack Obama, the president>, ...
World Knowledge from Unannotated Data

- can extract syntactic appositions heuristically
  - shown to be useful for coreference resolution (e.g., Daume & Marcu, 2005, Ng, 2007, Haghighi & Klein, 2009)

- Each extraction is an NP pair. E.g.,
  - <Barack Obama, the president>, <Delta Airlines, the carrier>

- Create a database consisting of the syntactic appositions extracted from an unannotated corpus
  - 1.057 million NP pairs
Features based on Syntactic Appositions

- Create a binary-valued feature

- **Mention-pair model**: feature value is
  \[
  \begin{cases}
  1 & \text{if the two NPs appear as a pair in the database} \\
  0 & \text{otherwise}
  \end{cases}
  \]

- **Cluster-ranking model**
  - extend the definition above so that the feature can be applied to an NP and a preceding cluster