ML for Coreference Resolution

- noun phrase coreference resolution
  - quick review
- a (supervised) machine learning approach
  - the truth this time
- weakly supervised approaches

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...
Noun Phrase Coreference

Identify all noun phrases that refer to the same entity

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

A Machine Learning Approach

- Classification
  - given a description of two noun phrases, $NP_i$ and $NP_j$, classify the pair as coreferent or not coreferent

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, a renowned speech therapist, was summoned to help the King overcome his speech impediment...

Aone & Bennett [1995]; Connolly et al. [1994]; McCarthy & Lehnert [1995]; Soon et al. [2001]; Ng & Cardie [2002]; …
**A Machine Learning Approach**

- **Clustering**
  - coordinates pairwise coreference decisions

  ![Clustering Algorithm Diagram]

  - [Queen Elizabeth], coref
  - [her]
  - [husband], not coref

  **Queen Elizabeth**
  **King George VI**

  **Clustering Algorithm**

  - set about transforming
  - husband
  - the King
  - his
  - Logue
  - Logue
  - a renowned speech therapist

**Training Data Creation**

- **Creating training instances**
  - texts annotated with coreference information

  - candidate antecedent
  - anaphor

  - one instance \( inst(NP_i, NP_j) \) for each ordered pair of NPs
  - \( NP_i \) precedes \( NP_j \)
  - feature vector: describes the two NPs and context
  - class value:
    - coref
    - not coref
    - pairs on the same coreference chain
    - otherwise

**Instance Representation**

- 25 features per instance
  - lexical (3)
    - string matching for pronouns, proper names, common nouns
  - grammatical (18)
    - pronoun_1, pronoun_2, demonstrative_2, indefinite_2, ...
    - number, gender, animacy
    - appositive, predicate nominative
    - binding constraints, simple contra-indexing constraints, ...
    - span, maximalnp, ...
  - semantic (2)
    - same WordNet class
    - alias
  - positional (1)
    - distance between the NPs in terms of # of sentences
  - knowledge-based (1)
    - naive pronoun resolution algorithm

**Learning Algorithm**

- RIPPER (Cohen, 1995)
  - rule learners
  - input: set of training instances
  - output: coreference classifier

- C4.5 (Quinlan, 1994)
  - rule learners
  - input: set of training instances
  - output: coreference classifier

- Learned classifier
  - input: test instance (represents pair of NPs)
  - output: classification confidence of classification
Clustering Algorithm

- Best-first single-link clustering
  - Mark each $NP_j$ as belonging to its own class: $NP_j \in c_j$
  - Proceed through the NPs in left-to-right order.
    - For each NP, $NP_j$, create test instances, $inst(NP_i, NP_j)$, for all of its preceding NPs, $NP_i$.
    - Select as the antecedent for $NP_j$ the highest-confidence coreferent NP, $NP_i$, according to the coreference classifier (or none if all have below .5 confidence); Merge $c_j$ and $c_i$.

Baseline Results

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>40.7</td>
<td>73.5</td>
</tr>
<tr>
<td>Worst MUC System</td>
<td>36</td>
<td>44</td>
</tr>
<tr>
<td>Best MUC System</td>
<td>59</td>
<td>72</td>
</tr>
<tr>
<td>Ng &amp; Cardie</td>
<td>63.3</td>
<td>76.9</td>
</tr>
</tbody>
</table>

Detailed Results

<table>
<thead>
<tr>
<th></th>
<th>C4.5</th>
<th>RIPPER</th>
</tr>
</thead>
<tbody>
<tr>
<td>MUC-6</td>
<td>MUC-7</td>
<td>MUC-6</td>
</tr>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Original Soon</td>
<td>58.6</td>
<td>67.3</td>
</tr>
<tr>
<td>Duplicated Soon Bsln</td>
<td>64.0</td>
<td>67.0</td>
</tr>
<tr>
<td>Learning Framework</td>
<td>62.4</td>
<td>73.5</td>
</tr>
<tr>
<td>All Feats</td>
<td>70.1</td>
<td>58.3</td>
</tr>
<tr>
<td>Hand Feats</td>
<td>64.1</td>
<td>74.9</td>
</tr>
<tr>
<td>pronouns</td>
<td>-</td>
<td>77.5</td>
</tr>
<tr>
<td>proper</td>
<td>-</td>
<td>94.8</td>
</tr>
<tr>
<td>generic</td>
<td>54.7</td>
<td>64.8</td>
</tr>
</tbody>
</table>

Results

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Ng &amp; Cardie</td>
<td>63.3</td>
<td>76.9</td>
</tr>
<tr>
<td>Best MUC System</td>
<td>59</td>
<td>72</td>
</tr>
</tbody>
</table>
Problem 1

- Coreference is a rare relation
  - skewed class distributions (2% positive instances)
  - remove some negative instances

Problem 2

- Coreference is a discourse-level problem with different solutions for different types of NPs
  * proper names: string matching and aliasing
  - inclusion of “hard” positive training instances
  - *positive example selection*: selects easy positive training instances (cf. Harabagiu et al. (2001))

Problem 3

- Coreference is an equivalence relation
  - loss of transitivity
  - need to tighten the connection between classification and clustering
  - prune learned rules w.r.t. the clustering-level coreference scoring function

Queen Elizabeth set about transforming her husband, King George VI, into a viable monarch. Logue, the renowned speech therapist, was summoned to help the King overcome his speech impediment...
### Results

<table>
<thead>
<tr>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>40.7</td>
</tr>
<tr>
<td>NEG-SELECT</td>
<td>46.5</td>
</tr>
<tr>
<td>POS-SELECT</td>
<td>53.1</td>
</tr>
<tr>
<td>NEG-SELECT + POS-SELECT</td>
<td>63.4</td>
</tr>
<tr>
<td>NEG-SELECT + POS-SELECT + RULE-SELECT</td>
<td>63.3</td>
</tr>
</tbody>
</table>

- Ultimately: large increase in F-measure, due to gains in recall

### Comparison with Best MUC Systems

<table>
<thead>
<tr>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Best MUC System</td>
<td>59</td>
</tr>
</tbody>
</table>

| NEG-SELECT + POS-SELECT + RULE-SELECT | 63.3 | 76.9 | 69.5 | 54.2 | 76.3 | 63.4 |

### Supervised ML for NP Coreference

- Good performance compared to other systems, but...lots of room for improvement
  - Common nouns < pronouns < proper nouns
  - Tighter connection between classification and clustering is possible
    » Rich Caruana’s (2004) ensemble methods
    » Statistical methods for learning probabilistic relational models (Getoor et al., 2001; Lafferty et al., 2001; Taskar et al., 2003; McCallum and Wellner, 2003).
  - Need additional data sets
    » ACE data from Penn’s LDC
    » General problem: reliance on manually annotated data...

### Plan for the Talk

- noun phrase coreference resolution
- a (supervised) machine learning approach
  weakly supervised approaches
  - background
  - two techniques
  - evaluation
Weakly Supervised Approaches

- **Idea:**
  bootstrap (NP coreference) classifiers using a *small amount of labeled data* (expensive) and a *large amount of unlabeled data* (cheap)

- **Methods**
  - Co-training
  - Self-training

---

Co-Training [Blum and Mitchell, 1998]

- **Labeled data (L)**
- **Unlabeled data (U)**

Co-Training [Blum and Mitchell, 1998]

- **Classifier** $h_1$
- **Classifier** $h_2$

Co-Training [Blum and Mitchell, 1998]

- **Classifier** $h_1$
- **Classifier** $h_2$

---

Co-Training [Blum and Mitchell, 1998]

- **Classifier** $h_1$
- **Classifier** $h_2$
Potential Problems with Co-Training

- Strong assumptions on the views (Blum and Mitchell, 1998)
  - each view must be sufficient for learning the target concept
  - the views must be conditionally independent given the class
  - empirically shown to be sensitive to these assumptions (Muslea et al., 2002)
- A number of parameters need to be tuned
  - views, data pool size, growth size, number of iterations, initial size of labeled data
  - algorithm is sensitive to its input parameters (Nigam and Ghani, 2000; Pierce and Cardie, 2001; Pierce 2003)

Multi-view algorithm
- Is there any natural feature split for NP coreference?
  - view factorization is a non-trivial problem for coreference
    *Mueller et al.*’s (2002) greedy method
Self-Training with Bagging
[Banko and Brill, 2001]

Labeled data (L)

Unlabeled data (U)

Bagged Classifier $h_1$

Bagged Classifier $h_2$

... 

Bagged Classifier $h_n$

consistently labeled
Plan for the Talk

- noun phrase coreference resolution
- a (supervised) machine learning approach
- weakly supervised approaches
  - background
  - two techniques
  - evaluation

Evaluation

- MUC-6 and MUC-7 coreference data sets
- labeled data (L): one dryrun text
  - 3500-3700 instances
- unlabeled data (U): remaining 29 dryrun texts
- vs. fully supervised ML
  - ~500,000 instances (30 dryrun texts)

Results (Baseline)

- train a naïve Bayes classifier on the single (labeled) text using all 25 features

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
</tr>
</tbody>
</table>

Evaluating the Weakly Supervised Algorithms

- Determine the best parameter setting of each algorithm (in terms of its effectiveness in improving performance)
Co-Training Parameters

- **Views (3 heuristic methods for view factorization)**
  - Mueller et al.’s (2002) greedy method
  - random splitting
  - splitting according to the feature type
- **Pool size**
  - 500, 1000, 5000
- **Growth size**
  - 10, 50, 100, 200, 250
- **Number of co-training iterations**
  - run until performance stabilized

Results (Co-Training)

- co-training produces improvements over the baseline at its best parameter settings

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
</tr>
</tbody>
</table>

Supervised ML* (~500,000 insts)

- co-training produces improvements over the baseline at its best parameter settings

Learning Curve for Co-Training (MUC-6)

- pool size: 5000; growth size: 50; views: feature type

F-measure
Baseline
Self-Training Parameters

- Number of bags
  - tested all odd number of bags between 1 and 25
- 25 bags are sufficient for most learning tasks (Breiman, 1996)

Results (Self-Training with Bagging)

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
</tr>
<tr>
<td>Self-Training with Bagging</td>
<td>54.1</td>
<td>78.6</td>
</tr>
</tbody>
</table>

- Self-training performs better than co-training
**Summary**

- **Supervised ML approach to NP coreference resolution**
  - Good performance relative to other approaches
  - Still lots of room for improvement
- **Weakly supervised approaches are promising**
  - Not as good performance as fully supervised, but use much less manually annotated training data
- **For problems where no natural view factorization exists...**
  - Single-view weakly supervised algorithms
    - Self-training with bagging

---

**Results**

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
</tr>
<tr>
<td>Self-Training with Bagging</td>
<td>54.1</td>
<td>78.6</td>
</tr>
<tr>
<td>Supervised ML* (~500,000 insts)</td>
<td>63.3</td>
<td>76.9</td>
</tr>
</tbody>
</table>

---

**Self-Training: Effect of the Number of Bags (MUC-6)**

- Number of Bags: 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25
- F-measure
- Baseline

---

**Table**

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
</tr>
<tr>
<td>Self-Training with Bagging</td>
<td>54.1</td>
<td>78.6</td>
</tr>
<tr>
<td>Supervised ML* (~500,000 insts)</td>
<td>63.3</td>
<td>76.9</td>
</tr>
</tbody>
</table>