Outline

- noun phrase coreference resolution
  - a (supervised) machine learning approach
    - evaluation
    - problems...some solutions
- weakly supervised approaches

Knowledge-based approaches are still common. E.g.
- Lappin & Leass [1994]
- CogNIAC [Baldwin, 1996]
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], ...

coref ? coref ? coref ?

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

[Queen Elizabeth], not coref

[her] coref

[husband] not coref

Queen Elizabeth

her

King George VI

husband

the King

his

Logue

a renowned speech therapist

Clustering Algorithm
Training Data Creation

- Creating training instances
  - texts annotated with coreference information

  candidate antecedent \hspace{1cm} \text{anaphor}

  - one instance $\text{inst}(NP_i, NP_j)$ for each ordered pair of NPs
    - $NP_i$ precedes $NP_j$
    - feature vector: describes the two NPs and context
    - class value:
      - $\text{coref}$ pairs on the same coreference chain
      - $\text{not coref}$ 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)
    » naïve pronoun resolution algorithm
Learning Algorithm

- RIPPER (Cohen, 1995)
  - 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 \( c_j \).
Outline

- noun phrase coreference resolution
- a (supervised) machine learning approach
  - evaluation
  - problems...some solutions
- weakly supervised approaches
Evaluation

- MUC-6 and MUC-7 coreference data sets
- documents annotated w.r.t. coreference
- 30 + 30 training texts (dry run)
- 30 + 20 test texts (formal evaluation)
- scoring program
  - recall
  - precision
  - F-measure: $2PR/(P+R)$
# 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>

<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>
</tbody>
</table>
Classifier for MUC-6 Data Set
Problem 1

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

farthest antecedent
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))

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...
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], ...

[coref? coref? not coref?]
### Results

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th></th>
<th>MUC-7</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
</tr>
<tr>
<td>Baseline</td>
<td>40.7</td>
<td>73.5</td>
<td>52.4</td>
<td>27.2</td>
</tr>
<tr>
<td>NEG-SELECT</td>
<td>46.5</td>
<td>67.8</td>
<td>55.2</td>
<td>37.4</td>
</tr>
<tr>
<td>POS-SELECT</td>
<td>53.1</td>
<td>80.8</td>
<td>64.1</td>
<td>41.1</td>
</tr>
<tr>
<td>NEG-SELECT + POS-SELECT</td>
<td>63.4</td>
<td>76.3</td>
<td>69.3</td>
<td>59.5</td>
</tr>
<tr>
<td>NEG-SELECT + POS-SELECT + RULE-SELECT</td>
<td>63.3</td>
<td>76.9</td>
<td><strong>69.5</strong></td>
<td>54.2</td>
</tr>
</tbody>
</table>

- Ultimately: large increase in F-measure, due to gains in recall
## Comparison with Best MUC Systems

<table>
<thead>
<tr>
<th>Method</th>
<th>MUC-6</th>
<th>MUC-7</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>NEG-SELECT + POS-SELECT + RULE-SELECT</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>
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
  - Need additional data sets
    » ACE data from Penn’s LDC
    » General problem: reliance on manually annotated data...
Outline

- 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]

\[ \text{Labeled data (L)} \]

\[ \text{Unlabeled data (U)} \]

\[ \text{Classifier } h_1 \]

\[ \text{Classifier } h_2 \]

view \( V_1 \)

view \( V_2 \)
Co-Training [Blum and Mitchell, 1998]

Labeled data (L)

Unlabeled data (U)

Classifier $h_1$

Classifier $h_2$

view $V_1$

view $V_2$
Co-Training [Blum and Mitchell, 1998]

- Labeled data (L)
  - View $V_1$
  - Classifier $h_1$
- Unlabeled data (U)
  - View $V_2$
  - Classifier $h_2$
- Data pool (D)
Co-Training [Blum and Mitchell, 1998]
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)
Self-Training with Bagging
[Banko and Brill, 2001]

x x x

Labeled data (L)

Bagged Classifier $h_1$

Bagged Classifier $h_2$

. . .

Bagged Classifier $h_n$

x x x x x x x x x x x x x

Unlabeled data (U)
Self-Training with Bagging
[Banko and Brill, 2001]

Unlabeled data (U)

Labeled data (L)

Bagged Classifier \( h_1 \)

Bagged Classifier \( h_2 \)

\[ \ldots \]

Bagged Classifier \( h_n \)
Self-Training with Bagging
[Banko and Brill, 2001]

Consistently labeled

Labeled data (L)

Bagged Classifier \( h_1 \)

Bagged Classifier \( h_2 \)

... ...

Bagged Classifier \( h_n \)

Unlabeled data (U)

\[ \text{Bagged} \]

\[ \text{Classifier} h \]

\[ x \]

\[ x \]

\[ x \]

\[ x \]

\[ x \]

\[ x \]

\[ x \]

\[ x \]
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)

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th></th>
<th>MUC-7</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
<td><strong>55.5</strong></td>
<td>52.8</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td><strong>81.9</strong></td>
<td><strong>60.1</strong></td>
<td>40.6</td>
</tr>
</tbody>
</table>

- co-training produces improvements over the baseline at its best parameter settings
## Results (Co-Training)

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th></th>
<th></th>
<th>MUC-7</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
<td>P</td>
<td>F</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
<td>55.5</td>
<td>52.8</td>
<td>37.4</td>
<td>43.8</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
<td>60.1</td>
<td>40.6</td>
<td>77.6</td>
<td>53.3</td>
</tr>
<tr>
<td>Supervised ML* (~500,000 insts)</td>
<td>63.3</td>
<td>76.9</td>
<td>69.5</td>
<td>54.2</td>
<td>76.3</td>
<td>63.4</td>
</tr>
</tbody>
</table>

- 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

Number of Co-Training Iterations

F-measure
Baseline
Learning Curve for Co-Training (MUC-6)

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

|L| = 1000

Number of Iterations

Baseline
F-measure
Learning Curve for Co-Training (MUC-6)

pool size: 5000; growth size: 50; views: Mueller’s

Number of Co-Training Iterations

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></th>
<th>MUC-7</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
<td><strong>55.5</strong></td>
<td>52.8</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
<td><strong>60.1</strong></td>
<td>40.6</td>
</tr>
<tr>
<td>Self-Training with Bagging</td>
<td>54.1</td>
<td>78.6</td>
<td><strong>64.1</strong></td>
<td>54.6</td>
</tr>
</tbody>
</table>

- Self-training performs better than co-training
Self-Training: Effect of the Number of Bags (MUC-6)

Number of Bags

F-measure
Baseline
## Results

<table>
<thead>
<tr>
<th></th>
<th>MUC-6</th>
<th></th>
<th>MUC-7</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
</tr>
<tr>
<td>Baseline</td>
<td>58.3</td>
<td>52.9</td>
<td><strong>55.5</strong></td>
<td>52.8</td>
</tr>
<tr>
<td>Co-Training</td>
<td>47.5</td>
<td>81.9</td>
<td><strong>60.1</strong></td>
<td>40.6</td>
</tr>
<tr>
<td>Self-Training with Bagging</td>
<td>54.1</td>
<td>78.6</td>
<td><strong>64.1</strong></td>
<td>54.6</td>
</tr>
<tr>
<td>Supervised ML* (<em>~500,000 insts</em>)</td>
<td><strong>63.3</strong></td>
<td>76.9</td>
<td>69.5</td>
<td>54.2</td>
</tr>
</tbody>
</table>
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
...and also

1. Illustrate how much you’ve learned
2. Realities of doing work in NLP+ML
3. Introduce some cool weakly supervised learning methods