Ensemble-Based Coreference Resolution

Altaf Rahman and Vincent Ng
Human Language Technology Research Institute
The University of Texas at Dallas
Coreference Resolution

- Identify all noun phrases (mentions) that refer to the same real world entity

John Simon, Chief Financial Officer of Prime Corp. since 1986, saw his pay jump 20%, to $1.3 million, as the 37-year-old also became the financial-services company’s president...
Ensemble Approach

1. What?
   - Employ an ensemble of models for making coreference decisions

1. Why?
   - **Hypothesis**: Existing coreference models have complementary strengths and weaknesses, i.e., no single model is the best!

1. Goal
   - Investigate new methods for creating and applying ensembles for coreference resolution
Related Works

- Existing methods for creating ensemble for coreference resolution:
  - Munson et al. (2005) employ different learning algorithms.
  - Ng (2005) employs different clustering algorithms.
Creating an Ensemble

1. Two new methods
   1. Method 1: employs different linguistic feature sets
   2. Method 2: employs different supervised coreference models
Ensemble Creation : Method 1

1. 3 different feature set

1. Conventional Feature Set

- It contains 39 commonly-used coreference features, which can be divided into four categories:
  - String-matching features: exact and partial string match, ...
  - Grammatical features: gender and number agreement, ...
  - Semantic features: alias, semantic class compatibility, ...
  - Positional features: distance between two NPs in sentences, ...

2. Lexical Feature Set

- It contains word pairs collected from coreference-annotated documents, for example: his-president, Simon-his, Prime Corp-his

3. Combined Feature Set

- Union of the above two features.

---

John Simon, Chief Financial Officer of Prime Corp. since 1986, saw his pay jump 20%, to $1.3 million, as the 37-year-old also became the financial-services company's president...
Ensemble Creation: Method 2

3 different supervised models

1. **Mention Pair (MP) model** (Soon et al., 2001; Ng & Cardie, 2002)
   - A classifier that determines whether two NPs are coreferent.
   - Weaknesses:
     - Each candidate antecedent is considered independently of the others.
     - Insufficient information to make an informed coreferenced decision based on two NPs only.

2. **Mention Ranking (MR) model** (Denis & Baldridge, 2008)
   - A ranker that ranks the candidate antecedents for each anaphor.
   - Advantage:
     - Considers all the candidate antecedents simultaneously.

3. **Cluster Ranking (CR) model** (Rahman & Ng, 2009)
   - A ranker that ranks the preceding clusters for each anaphor.
   - Advantage:
     - Considers all the candidate antecedents simultaneously.
     - It employs cluster-level features:
       - Defined over any subset of NPs in a preceding cluster.
       - Derived from the combined features by applying logical predicates.
   - Advantage:
     - Considers all the candidate antecedents simultaneously.
     - It also improves expressiveness by using cluster level features.
Creating the Ensemble

Given these two methods, we create a 9-member ensemble

- Since each of the three models can be trained in combination with each of the three feature sets, we can create nine coreference systems.
Applying the Ensemble

• Challenge:
  • Our ensemble is **model-heterogeneous**, so comprising both pair-wise models (e.g., the MP model) and a cluster-based model (i.e., the CR model), combining the coreference decisions made by different models is not straightforward
  • Consequently, we propose **4 methods** for applying our ensemble.
Method 1: Applying Best Per-NP-Type Model

Motivation: different members of the ensemble are good at resolving different types of NPs.

Identify the best model resolving each type of NPs by using a held-out dev-set.

Resolving an NP:

1. Identify the type of the NP
2. Resolve it using the model that was determined to be the best at handling this NP type.
Method 1: Applying Best Per-NP-Type Model (cont.)

1. How many NP types should be used?

- Three super types (*Name, Nominal* and *Pronoun*) are further divided into
  total 10 subtypes:
  - Name and Nominal:
    - e (exact string match)
    - p (partial string match)
    - n (no string match)
  - Pronoun:
    - 1+2 (1st and 2nd person pronoun)
    - G3 (gendered 3rd person)
    - U3 (ungendered 3rd person)
    - oa (other anaphoric pronoun)

2. How can we determine which model performs the best for an NP type on the development set?

- For each type C of NP we use a model and rest of the NPs are resolved by the oracle.
- Compute F-measure score only on the NPs belong to type C
Method 2: Antecedent-Based Voting

1. Given an NP to resolve, \( NP_k \), each of the 9 models selects an antecedent \( NP_k \) independently -

2. The candidate antecedent that receives the largest number of votes will be selected as the antecedent for \( NP_k \)

3. Caveat: since Cluster Ranking (CR) members select preceding clusters, we force them to select the last NP of the cluster as the antecedent.
Method 3: Cluster-Based Voting

1. A natural alternative to method 2.

1. **Idea**: instead of forcing the CR-based members to select antecedents, we force the MP- and MR-based members to select preceding clusters
   - if the MP and MR model selects NP$_j$ as the antecedent, then we assume that it selects the preceding cluster containing NP$_j$
   - Every NP in the selected preceding cluster gets one vote
   - The NP with the largest number of votes wins
Method 4: Weighted Cluster-Based Voting

1. **Motivation**: In Method 3, all the votes casted for a candidate antecedent have equal weights; in practice, however, some members are more important than the others, so their votes should have higher weights.

2. **Dev-set**: we learn the weights on held-out development data using a hill-climbing algorithm which optimizes the weight of one member at a time, selecting the weight from the set \{-4, -3, -2, -1, 0, 1, 2, 3, 4\}

3. **Testing**: we then perform cluster-based voting, except that votes are weighted
   - The antecedent NP with the largest number of weighted votes wins
Experimental Setup

1. **Corpus:** ACE 2005, which has 6 data sources
   - broadcast news (bn), broadcast conversations (bc), newswire (nw), webblog (wb), usenet (un), and conversational telephone speech (cts)
2. For each data source, use 80% of data for training; 20% for testing
3. Extract NPs using a mention detector trained on training texts
4. All coreference models are trained using SVM\textsuperscript{light}
5. System output is scored using B\textsuperscript{3} (Bagga & Baldwin, 1998)
Evaluation

1. **Baselines**: Since our goal is to determine the effectiveness of ensemble approaches, the baselines are non-ensemble-based
   
   - 9 baselines, corresponding to the 9 members of the ensemble.
Baseline Results

<table>
<thead>
<tr>
<th>src</th>
<th>MP Models</th>
<th>MR Models</th>
<th>CR Models</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>conv</td>
<td>lex</td>
<td>comb</td>
</tr>
<tr>
<td>bc</td>
<td>50.8</td>
<td>57.4</td>
<td>55.7</td>
</tr>
<tr>
<td>bn</td>
<td>53.4</td>
<td>62.3</td>
<td>62.7</td>
</tr>
<tr>
<td>cts</td>
<td>57.0</td>
<td>61.1</td>
<td>61.3</td>
</tr>
<tr>
<td>nw</td>
<td>57.7</td>
<td>64.9</td>
<td>60.8</td>
</tr>
<tr>
<td>un</td>
<td>53.7</td>
<td>54.8</td>
<td>55.4</td>
</tr>
<tr>
<td>wb</td>
<td>63.3</td>
<td>65.2</td>
<td>57.6</td>
</tr>
<tr>
<td>all</td>
<td>56.2</td>
<td>61.2</td>
<td>58.8</td>
</tr>
</tbody>
</table>

1. 9 baseline systems on the test set, reported in terms of B^3 F-measure

- Columns labeled ‘conv’, ‘lex’, and ‘comb’ correspond to the *Conventional, Lexical, and Combined* feature sets, respectively.

- Aggregate results are in the last row

- The best performing baseline is CR-comb, which achieves comparable performance to Haghighi & Klein's (2010) system on the same test set.
Ensemble Results

<table>
<thead>
<tr>
<th>src</th>
<th>MP Models</th>
<th>MR Models</th>
<th>CR Models</th>
<th>Ensembles</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cnv</td>
<td>lex</td>
<td>cmb</td>
<td>cnv</td>
</tr>
<tr>
<td>bc</td>
<td>50.8</td>
<td>57.4</td>
<td>55.7</td>
<td>52.9</td>
</tr>
<tr>
<td>bn</td>
<td>53.4</td>
<td>62.3</td>
<td>62.7</td>
<td>55.8</td>
</tr>
<tr>
<td>cts</td>
<td>57.0</td>
<td>61.1</td>
<td>61.3</td>
<td>58.6</td>
</tr>
<tr>
<td>nw</td>
<td>57.7</td>
<td>64.9</td>
<td>60.8</td>
<td>60.2</td>
</tr>
<tr>
<td>un</td>
<td>53.7</td>
<td>54.8</td>
<td>55.4</td>
<td>55.6</td>
</tr>
<tr>
<td>wb</td>
<td>63.3</td>
<td>65.2</td>
<td>57.6</td>
<td>65.2</td>
</tr>
<tr>
<td>all</td>
<td>56.2</td>
<td>61.2</td>
<td>58.8</td>
<td>58.2</td>
</tr>
</tbody>
</table>

1. Ensemble approaches: M1, M2, M3, M4 correspond to the 4 methods for applying ensembles.

- All four ensemble methods perform better than CR-comb
- Ensemble approaches can indeed improve coreference resolution (M1 < M2 < M3 < M4)
- M4 (best ensemble method, F-measure: 66.8) outperforms CR-comb by 4.0% and achieves the best performance on each data source.
Ensemble Results

<table>
<thead>
<tr>
<th></th>
<th>CR-comb</th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>P</td>
<td>F</td>
<td>R</td>
<td>P</td>
</tr>
<tr>
<td>all</td>
<td>54.4</td>
<td>74.8</td>
<td>62.8</td>
<td>55.1</td>
<td>75.6</td>
</tr>
</tbody>
</table>

- M1, M2, M3 and M4 - all improve on both recall and precision over CR-comb model.
Summary

- New methods for creating and applying ensembles of learning-based coreference systems
  - Uses different supervised models (pair-wise and cluster-based) and different feature sets.
- Experimental results on the ACE 2005 data set show that all four ensemble methods outperform the best baseline.
  - The best result was achieved by applying weighted cluster-based voting.
Thank You !!!