CS 181: Natural Language Processing

Lecture 21: Pronoun Resolution Algorithms

Kim Bruce
Pomona College

Spring 2008

Disclaimer: Slide contents borrowed from many sources on web!

Pronouns

- Reference to an entity already introduced called *anaphora*.
- Pronoun is *licensed* by previous mention of an *antecedent*.
- Pronoun resolution subset of general reference resolution.

Antecedent Game

- Constraints on antecedents:
  - Number agreement.
    - John his a ball. He threw *them* far.
    - *but:*
      - Microsoft released a new version of Windows today. *They* hope it will be more successful than Vista.
  - Person agreement
    - 1st, 2nd, 3rd person match
  - Gender agreement
    - he/she/it

Antecedent Game

- Binding theory constraints:
  - John bought himself an ice cream.
  - John bought him an ice cream
  - John said that Bill bought him an ice cream
  - John said that Bill bought himself an ice cream
  - He said that he bought Bill an ice cream
  - *Constraints on meaning of him, himself, he.*

Antecedent Game

- Selectional restrictions:
  - John ate his sandwich in his office.
    - It was made with roast beef.
    - It was quieter than eating in the snack bar.
- Recency:
  - Lee met Mary for lunch. They saw Sue at the restaurant. She gave Lee a hug.
- Grammatical role: *Subject > object*
  - Jane saw Sally at the market. She went over to say hello.

Antecedent Game

- Repeated mention:
    - John had a long day. He had not gotten much sleep the night before. He and Fred went to the movies that night. He had a hard time staying awake.
- Parallelism
  - Jane helped Mary with her Physics homework. Ellen helped her with her English.
- Verb Semantics:
  - Jane gave Mary the letter.
    - She was excited to receive it.
    - She had received it yesterday.

Algorithms for Pronominal Anaphora Resolution

Hobbs 1978

- Works on parse trees of sentence containing pronoun and of all previous sentences.
- Approximates binding theory, recency, and grammatical role preferences.
- Uses info on gender, person, and number constraints as a final check.

Hobbs

1. Begin at NP immediately dominating the pronoun
2. Go up tree to first NP or S node enountered. Call it X and path to it p.
3. Traverse all branches below X to left of path p in a left-to-right, breadth-first fashion. Propose as antecedent any NP node encountered which has an NP or S node between it and X.
4. If X is highest S node in sentence, traverse parse trees of previous sentences in order of recency, each in a left-to-right, breadth-first manner, and when an NP is encountered, propose as antecedent. If X not highest, go to 5

5. From X go up to first NP or S. Call new node X and path to it p.
6. If X is NP and p did not pass through Nominal that X immediately dominates, propose X as antecedent.
7. Traverse all branches below X to left of p in left-to-right, breadth-first manner, but do not go below any NP or S encountered.
8. If X is S node, traverse all branches of X to right of p in left-to-right, breadth-first manner, but do not go below any NP or S node encountered. Propose any NP encountered as antecedent.
9. Go to step 4.

Examples

- John saw a beautiful MGB at the dealership.
- He showed it to Bob.
- He bought it.

Final Check

- Parsers generally return number and person info, but usually not gender.
- Check hyponyms in WordNet of head noun.
  - Person, living thing indicate animate noun
  - female indicates female gender, ...
  - Cues in titles: Mr., Ms.

Centering Algorithm

- Claim: There is single entity being "centered" on at any point in the discourse.
- Let U_n, U_{n+1} be 2 consecutive utterances.
- Backward looking center of U_n, written $C_b(U_n)$, represents focus after U_n interpreted.
- Forward looking centers of U_n, written $C_f(U_n)$, forms ordered list of entities in U_n that can serve as $C_b(U_{n+1})$.
- $C_b(U_{n+1})$ is highest ranking elt of $C_f(U_n)$ mentioned in U_{n+1}.

Centers

- Order of entities in $C_f(U_n)$:
  - subject > existential predicate nominal > object > indirect object > demarcated adverbial PP
- Let $C_p(U_{n+1})$ be highest ranked forward looking center

State-Based Transitions

	$C_b(U_{n+1}) = C_b(U_n)$ or undefined $C_b(U_n)$	$C_b(U_{n+1}) \neq C_b(U_n)$
$C_b(U_{n+1}) = C_p(U_{n+1})$	Continue	Smooth-Shift
$C_b(U_{n+1}) \neq C_p(U_{n+1})$	Retain	Rough-Shift

- Rule 1: If any elt of Cf(Un) is realized by a pronoun in Un+1 then Cb(Un+1) must be realized as a pronoun also.
- Rule 2: Transition states are ordered. Continue > Retain > Smooth-Shift > Rough-Shift.

Centering Algorithm

- Generate possible C_b - C_f combinations for each possible set of reference assignments.
- Filter by constraints (syntactic coreference constraints, selectional, centering rules and constraints).
- Rank by transition orderings
- Assign referents based on Rule 2, if Rule 1 and other constraints not violated.

Example redux

- John saw a beautiful MGB at the dealership.
- He showed it to Bob.
- He bought it.

Example redux

- John saw a beautiful MGB at the dealership.
  - $C_f(U_1)$ = {John, MGB, dealership} - *in order*
  - $C_p(U_1)$ = John
  - $C_b(U_1)$: undefined (*highest ranked from prev Cf*)

- He showed it to Bob. {it = MGB?}
  - $C_f(U_2)$ = {John, MGB, Bob}
  - $C_p(U_2)$ = John
  - $C_b(U_2)$: John *highest from $C_f(U_1)$*
  - Result: continue - *$C_p(U_2) = C_b(U_2)$, $C_b(U_1)$ undefined*
- He showed it to Bob. {it = dealership?}
  - $C_f(U_2)$ = {John, dealership, Bob}
  - $C_p(U_2)$ = John
  - $C_b(U_2)$: John
  - Result: continue - *$C_p(U_2) = C_b(U_2)$, $C_b(U_1)$ undefined*
- Tied, arb pick MGB since 1st in $C_f(U_1)$

- He bought it. {it = MGB, he = John?}
  - $C_f(U_3)$ = {John, MGB}
  - $C_p(U_3)$ = John
  - $C_b(U_3)$: John *highest from $C_f(U_2)$*
  - Result: continue - *$C_p(U_3) = C_b(U_3) = C_b(U_2)$*
- He bought it. {it = MGB, he = Bob?}
  - $C_f(U_3)$ = {Bob, MGB}
  - $C_p(U_3)$ = Bob
  - $C_b(U_3)$: Bob
  - Result: Smooth-Shift - *$C_p(U_3) = C_b(U_3)$, $C_b(U_3)$!= $C_b(U_2)$*
- Pick John as continue > Smooth-shift

Centering

- Implicitly incorporates grammatical role, recency, and repeated mention.
- Can get confused.
  - Bob opened a new bike shop last week. John took a look at the road bikes in his shop. He ended up buying one.
  - Incorrectly assigns he to "Bob" because $C_b(U_2)$ = Bob so get continue, while "John" gets smooth-shift.

Machine Learning

- Train classifier: Log-linear (we skipped) or Naive Bayes.
- Rely on hand-labeled corpus where each pronoun linked to antecedent.
- Present positive and negative results for training.
- Extract features for training.

Features

- Commonly used for anaphora resolution:
  - strict gender [boolean]
  - compatible gender [boolean]
  - strict number [boolean]
  - compatible number [boolean]
  - sentence distance [0,1,2,...] from pronoun
  - Hobbs distance [0,1,2,...] # Hobbs NP skipped
  - Grammatical role [subject, object, PP]
  - Linguistic form [proper, definite, indefinite, pronoun]

Example

- John saw an MGB at the dealership. (U_1)
- He showed it to Bob. (U_3)
- He bought it. (U_3)

	He (U_2)	it (U_2)	Bob (U_2)	John (U_1)
strict number	1	1	1	1
compatible number	1	1	1	1
strict gender	1	0	1	1
compatible gender	1	0	1	1
sentence distance	1	1	1	2
Hobbs distance	2	1	0	3
grammatical role	subject	object	PP	subject
linguistic form	pronoun	pronoun	proper	proper

Training

- Train on vectors.
  - Filter out pleonastic "it" as in "it is raining"
  - Results in weights for each of the features and combinations of features.

Co-reference Resolution

Coreferences

- Extract coreference chains
  - Secretary of State Colin Powell, he, Mr. Powell, Powell.
  - Condoleeza Rice, she, Rice
  - President Bush, Bush
- Can use machine learning classifier as before
  - Process from left to right.
  - For each NP, search backwards for match using classifier

Need more features

- Need to recognize that Microsoft is company to make sense of:
  - Microsoft announced record profits today. The company ...
- Jane .... The 30 year old mother of two ...

Common Features

- Anaphor edit distance [0,1,2,...]:

$$100 * \frac{m - (s + i + d)}{m}$$

  where m = size of antecedent.
- Antecedent edit distance [0,1,2,...]

$$100 * \frac{n - (s + i + d)}{n}$$

  where n = size of anaphor

Common Features

- alias [true or false]: names equivalent or acronyms.
- appositive [true or false]: Mary, the new student, ...
- linguistic form [proper, definite, indef, pronoun] type of anaphor

Psychological Justification

- Reading time experiments
  - Clark & Sengal found reading time faster when referent for pronoun in most recent clause, rather than 2 or 3 back (for which speeds same)
  - Crawley found subjects identified antecedent of pronoun if subject more often than if object.
  - Smyth found strong impact of parallel placement.
  - Matthews & Chodorow found slower comprehension when pronoun antecedent occupied early syntactically deep position

Any Questions?