

Using Semantic Relations to Refine Coreference Decisions

Heng Ji

hengji@cs.nyu.edu

David Westbrook

Department of Computer Science
New York University
New York, NY, 10003, USA
westbroo@cs.nyu.edu

Ralph Grishman

grishman@cs.nyu.edu

Abstract

We present a novel mechanism for improving reference resolution by using the output of a relation tagger to rescore coreference hypotheses. Experiments show that this new framework can improve performance on two quite different languages -- English and Chinese.

1 Introduction

Reference resolution has proven to be a major obstacle in building robust systems for information extraction, question answering, text summarization and a number of other natural language processing tasks.

Most reference resolution systems use representations built out of the lexical and syntactic attributes of the noun phrases (or “mentions”) for which reference is to be established. These attributes may involve string matching, agreement, syntactic distance, and positional information, and they tend to rely primarily on the immediate context of the noun phrases (with the possible exception of sentence-spanning distance measures such as Hobbs distance). Though gains have been made with such methods (Tetreault 2001; Mitkov 2000; Soon et al. 2001; Ng and Cardie 2002), there are clearly cases where this sort of local information will not be sufficient to resolve coreference correctly.

Coreference is by definition a semantic relationship: two noun phrases corefer if they both refer to the same real-world entity. We should therefore expect a successful coreference system to exploit world knowledge, inference, and other

forms of semantic information in order to resolve hard cases. If, for example, two nouns refer to people who work for two different organizations, we want our system to infer that these noun phrases cannot corefer. Further progress will likely be aided by flexible frameworks for representing and using the information provided by this kind of semantic relation between noun phrases.

This paper tries to make a small step in that direction. It describes a robust reference resolver that incorporates a broad range of semantic information in a general news domain. Using an ontology that describes relations between entities (the Automated Content Extraction program¹ relation ontology) along with a training corpus annotated for relations under this ontology, we first train a classifier for identifying relations. We then apply the output of this relation tagger to the task of reference resolution.

The rest of this paper is structured as follows. Section 2 briefly describes the efforts made by previous researchers to use semantic information in reference resolution. Section 3 describes our own method for incorporating document-level semantic context into coreference decisions. We propose a representation of semantic context that isolates a particularly informative structure of interaction between semantic relations and coreference. Section 4 explains in detail our strategies for using relation information to modify coreference decisions, and the linguistic intuitions behind these strategies. Section 5 then presents the system architectures and algorithms we use to incorporate relational information into reference resolution.

¹ The ACE task description can be found at <http://www.itl.nist.gov/iad/894.01/tests/ace/> and the ACE guidelines at <http://www ldc.upenn.edu/Projects/ACE/>

Section 6 presents the results of experiments on both English and Chinese test data. Section 7 presents our conclusions and directions for future work.

2 Prior Work

Much of the earlier work in anaphora resolution (from the 1970's and 1980's, in particular) relied heavily on deep semantic analysis and inference procedures (Charniak 1972; Wilensky 1983; Carbonell and Brown 1988; Hobbs et al. 1993). Using these methods, researchers were able to give accounts of some difficult examples, often by encoding quite elaborate world knowledge. Capturing sufficient knowledge to provide adequate coverage of even a limited but realistic domain was very difficult. Applying these reference resolution methods to a broad domain would require a large scale knowledge-engineering effort.

The focus for the last decade has been primarily on broad coverage systems using relatively shallow knowledge, and in particular on corpus-trained statistical models. Some of these systems attempt to apply shallow semantic information. (Ge et al. 1998) incorporate gender, number, and animacy information into a statistical model for anaphora resolution by gathering coreference statistics on particular nominal-pronoun pairs. (Tetreault and Allen 2004) use a semantic parser to add semantic constraints to the syntactic and agreement constraints in their Left-Right Centering algorithm. (Soon et al. 2001) use WordNet to test the semantic compatibility of individual noun phrase pairs. In general these approaches do not explore the possibility of exploiting the global semantic context provided by the document as a whole.

Recently Bean and Riloff (2004) have sought to acquire automatically some semantic patterns that can be used as contextual information to improve reference resolution, using techniques adapted from information extraction. Their experiments were conducted on collections of texts in two topic areas (terrorism and natural disasters).

3 Relational Model of Semantic Context

Our central goal is to model semantic and coreference structures in such a way that we can take advantage of a semantic context larger than the

individual noun phrase when making coreference decisions. Ideally, this model should make it possible to pick out important features in the context and to distinguish useful signals from background noise. It should, for example, be able to represent such basic relational facts as whether the (possibly identical) people referenced by two noun phrases work in the same organization, whether they own the same car, etc. And it should be able to use this information to resolve references even when surface features such as lexical or grammatical attributes are imperfect or fail altogether.

In this paper we present a Relational Coreference Model (abbreviated as RCM) that makes progress toward these goals. To represent semantic relations, we use an ontology (the ACE 2004 relation ontology) that describes 7 main types of relations between entities and 23 subtypes (Table 1).² These relations prove to be more reliable guides for coreference than simple lexical context or even tests for the semantic compatibility of heads and modifiers. The process of tagging relations implicitly selects relevant items of context and abstracts raw lists of modifiers into a representation that is deeper, but still relatively lightweight.

Relation Type	Example
Agent-Artifact (ART)	Rubin Military Design, the makers of the Kursk
Discourse (DISC)	each of whom
Employment/Membership (EMP-ORG)	Mr. Smith, a senior programmer at Microsoft
Place-Affiliation (GPE-AFF)	Salzburg Red Cross officials
Person-Social (PER-SOC)	relatives of the dead
Physical (PHYS)	a town some 50 miles south of Salzburg
Other-Affiliation (Other-AFF)	Republican senators

Table 1. Examples of the ACE Relation Types

Given these relations we can define a semantic context for a candidate mention coreference pair (Mention 1b and Mention 2b) using the structure

² See <http://www ldc.upenn.edu/Projects/ACE/docs/EnglishRDCV4-3-2.PDF> for a more complete description of ACE 2004 relations.

depicted in Figure 1. If both mentions participate in relations, we examine the types and directions of their respective relations as well as whether or not their relation partners (Mention 1a and Mention 2a) corefer. These values (which correspond to the edge labels in Figure 1) can then be factored into a coreference prediction. This RCM structure assimilates relation information into a coherent model of semantic context.

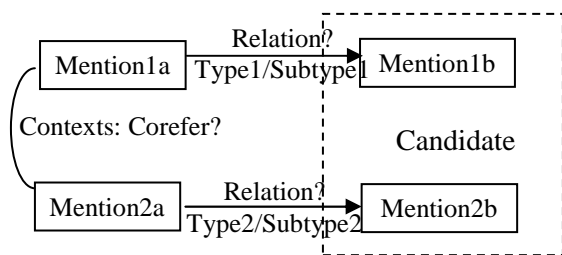


Figure 1. The RCM structure

4 Incorporating Relations into Reference Resolution

Given an instance of the RCM structure, we need to convert it into semantic knowledge that can be applied to a coreference decision. We approach this problem by constructing a set of RCM patterns and evaluating the accuracy of each pattern as positive or negative evidence for coreference. The resulting knowledge sources fall into two categories: rules that improve precision by pruning incorrect coreference links between mentions, and rules that improve recall by recovering missed links.

To formalize these relation patterns, based on Figure 1, we define the following clauses:

- A: $RelationType1 = RelationType2$
- B: $RelationSubType1 = RelationSubType2$
- C: Two Relations have the same direction
- Same_Relation: $A \wedge B \wedge C$
- CorefA: Mention1a and Mention2a corefer
- CorefBMoreLikely: Mention1b and Mention2b are more likely to corefer
- CorefBLessLikely: Mention1b and Mention2b are less likely to corefer

From these clauses we can construct the following plausible inferences:

Rule (1)
 $Same_Relation \wedge \neg CorefA \Rightarrow CorefBLessLikely$

Rule (2)
 $\neg Same_Relation \wedge CorefA \Rightarrow CorefBLessLikely$

Rule (3)
 $Same_Relation \wedge CorefA \Rightarrow CorefBMoreLikely$

Rule (1) and (2) can be used to prune coreference links that simple string matching might incorrectly assert; and (3) can be used to recover missed mention pairs.

The accuracy of Rules (1) and (3) varies depending on the type and direction of the particular relation shared by the two noun phrases. For example, if Mention1a and Mention 2a both refer to the same nation, and Mentions 1b and 2b participate in citizenship relations (GPE-AFF) with Mentions 1a and 2a respectively, we should not necessarily conclude that 1b and 2b refer to the same person. If 1a and 2a refer to the same person, however, and 1b and 2b are nations in citizenship relations with 1a and 2a, then it would indeed be the rare case in which 1b and 2b refer to two different nations. In other words, the relation of a nation to its citizens is one-to-many.

Our system learns broad restrictions like these by evaluating the accuracy of Rules (1) and (3) when they are instantiated with each possible relation type and direction and used as weak classifiers. For each such instantiation we use cross-validation on our training data to calculate a reliability weight defined as:

$$\frac{|\text{Correct decisions by rule for given instance}|}{|\text{Total applicable cases for given instance}|}$$

We count the number of correct decisions for a rule instance by taking the rule instance as the only source of information for coreference resolution and making only those decisions suggested by the rule's implication (interpreting CorefBMoreLikely as an assertion that mention 1b and mention 2b do in fact corefer, and interpreting CorefBLessLikely as an assertion that they do not corefer).

Every rule instance with a reliability weight of 70% or greater is retained for inclusion in the final system. Rule (2) cannot be instantiated with a single type because it requires that the two relation types be different, and so we do not perform this filtering for Rule (2) (Rule (2) has 97% accuracy across all relation types).

This procedure yields 58 reliable (reliability weight > 70%) type instantiations of Rule (1) and (3), in addition to the reliable Rule 2. We can

recover an additional 24 reliable rules by conjoining additional boolean tests to less reliable rules. Tests include equality of mention heads, substring matching, absence of temporal key words such as “current” and “former,” number agreement, and high confidence for original coreference decisions (Mention1b and Mention2b). For each rule below the reliability threshold, we search for combinations of 3 or fewer of these restrictions until we achieve reliability of 70% or we have exhausted the search space.

We give some examples of particular rule instances below.

Example for Rule (1)

Bush campaign officials ... decided to tone down a post-debate rally, and were even considering canceling it.

...

The Bush and Gore campaigns did not talk to each other directly about the possibility of postponement, but went through the debate commission's director, Janet Brown...Eventually, Brown recommended that the debate should go on, and neither side objected, according to campaign officials.

Two mentions that do not corefer share the same nominal head (“officials”). We can prune the coreference link by noting that both occurrences of “officials” participate in an Employee-Organization (EMP-ORG) relation, while the Organization arguments of these two relation instances do not corefer (because the second occurrence refers to officials from both campaigns).

Example for Rule (2)

Despite the increases, college remains affordable and a good investment, said College Board President Gaston Caperton in a statement with the surveys. ...

A majority of students need grants or loans -- or both -- but their exact numbers are unknown, a College Board spokesman said.

“Gaston Caperton” stands in relation EMP-ORG/Employee-Executive with “College Board”, while “a College Board spokesman” is in relation EMP-ORG/Employee-Staff with the same organiza-

tion. We conclude that “Gaston Caperton” does not corefer with “spokesman.”

Example for Rule (3)

In his foreign policy debut for Syria, this Sunday Bashar Assad met Sunday with Egyptian President Hosni Mubarak in talks on Mideast peace and the escalating violence in the Palestinian territories.

...

The Syrian leader's visit came on a fourth day of clashes that have raged in the West Bank, Gaza Strip and Jerusalem.....

If we have detected a coreference link between “Syria” and “Syrian,” as well as EMP-ORG/Employee-Executive relations between this country and two noun phrases “Bashar Assad” and “leader”, it is likely that the two mentions both refer to the same person. Without this inference, a resolver might have difficulty detecting this coreference link.

5 Algorithms

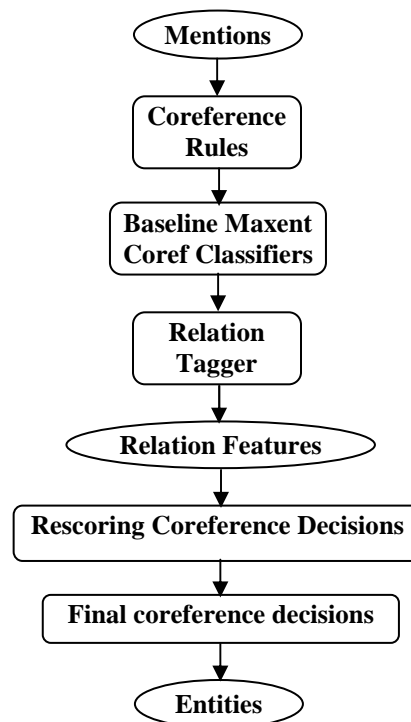


Figure 2. System Pipeline (Test Procedure)

In this section we will describe our algorithm for incorporating semantic relation information from the RCM into the reference resolver. In a nutshell, the system applies a baseline statistical resolver to generate multiple coreference hypotheses, applies a relation tagger to acquire relation information, and uses the relation information to rescore the coreference hypotheses. This general system architecture is shown in Figure 2.

In section 5.1 below we present our baseline coreference system. In Section 5.2 we describe a system that combines the output of this baseline system with relation information to improve performance.

5.1 Baseline System

Baseline reference resolver

As the first stage in the resolution process we apply a baseline reference resolver that uses no relation information at all. This baseline resolver goes through two successive stages.

First, high-precision heuristic rules make some positive and negative reference decisions. Rules include simple string matching (e.g., names that match exactly are resolved), agreement constraints (e.g., a nominal will never be resolved with an entity that doesn't agree in number), and reliable syntactic cues (e.g., mentions in apposition are resolved). When such a rule applies, it assigns a confidence value of 1 or 0 to a candidate mention-antecedent pair.

The remaining pairs are assigned confidence values by a collection of maximum entropy models. Since different mention types have different coreference problems, we separate the system into different models for names, nominals, and pronouns. Each model uses a distinct feature set, and for each instance only one of these three models is used to produce a probability that the instance represents a correct resolution of the mention. When the baseline is used as a standalone system, we apply a threshold to this probability: if some resolution has a confidence above the threshold, the highest confidence resolution will be made. Otherwise the mention is assumed to be the first mention of an entity. When the baseline is used as a component of the system depicted in figure 2, the confidence value is passed on to the rescoring stage described in 5.2 below.

Both the English and the Chinese coreference models incorporate features representing agreement of various kinds between noun phrases (number, gender, humanness), degree of string similarity, synonymy between noun phrase heads, measures of distance between noun phrases (such as the number of intervening sentences), the presence or absence of determiners or quantifiers, and a wide variety of other properties.

Relation tagger

The relation tagger uses a K-nearest-neighbor algorithm. We consider a mention pair as a possible instance of a relation only when: (1) there is at most one other mention between their heads, and (2) the coreference probability produced for the pair by the baseline resolver is lower than a threshold. Each training / test example consists of the pair of mentions and the sequence of intervening words. We defined a distance metric between two examples based on:

- whether the heads of the mentions match
- whether the ACE types of the heads of the mentions match (for example, both are people or both are organizations)
- whether the intervening words match

To tag a test example, we find the k nearest training examples, use the distance to weight each neighbor, and then select the most heavily weighted class in the weighted neighbor set.

Name tagger and noun phrase chunker

Our baseline name tagger consists of a HMM tagger augmented with a set of post-processing rules. The HMM tagger generally follows the Nymble model (Bikel et al. 1997), but with a larger number of states (12 for Chinese, 30 for English) to handle name prefixes and suffixes, and, for Chinese, transliterated foreign names separately. For Chinese it operates on the output of a word segmenter from Tsinghua University. Our nominal mention tagger (noun phrase chunker) is a maximum entropy tagger trained on treebanks from the University of Pennsylvania.

5.2 Rescoring stage

To incorporate information from the relation tagger into the final coreference decision, we split the maxent classification into two stages. The first

stage simply applies the baseline maxent models, without any relation information, and produces a probability of coreference. This probability becomes a feature in the second (rescoring) stage of maxent classification, together with features representing the relation knowledge sources. If a high reliability instantiation of one of the RCM rules (as defined in section 4 above) applies to a given mention-antecedent pair, we include the following features for that pair: the type of the RCM rule, the reliability of the rule instantiation, the relation type and subtype, the direction of the relation, and the tokens for the two mentions.

The second stage helps to increase the margin between correct and incorrect links and so effects better disambiguation. See figure 3 below for a more detailed description of the training and testing processes.

Training

1. Calculate reliability weights of relation knowledge sources using cross-validation (for each of k divisions of training data, train relation tagger on $k - 1$ divisions, tag relations in remaining division and compute reliability of each relation knowledge source using this division).
2. Use high reliability relation knowledge sources to generate relation features for 2nd stage Maxent training data.
3. Apply baseline coreference resolver to 2nd stage training data.
4. Using output of both 2 and 3 as features, train 2nd stage Maxent resolver.

Test

1. Tag relations.
 2. Convert relation knowledge sources into features for second stage Maxent models.
 3. Use baseline Maxent models to get coreference probabilities for use as features in second stage Maxent models.
 4. Using output of 2 and 3 as features for 2nd stage Maxent model, apply 2nd stage resolver to make final coreference decisions.
-

Figure 3. Training and Testing Processes

6 Evaluation Results

6.1 Corpora

We evaluated our system on two languages: English and Chinese. The following are the training corpora used for the components in these two languages.

English

For English, we trained the baseline maxent coreference model on 311 newswire and newspaper texts from the ACE 2002 and ACE 2003 training corpora. We trained the relation tagger on 328 ACE 2004 texts. We used 126 newswire texts from the ACE 2004 data to train the English second-stage model, and 65 newswire texts from the ACE 2004 evaluation set as a test set for the English system.

Chinese

For Chinese, the baseline reference resolver was trained on 767 texts from ACE 2003 and ACE 2004 training data. Both the baseline relation tagger and the rescoring model were trained on 646 texts from ACE 2004 training data. We used 100 ACE texts for a final blind test.

6.2 Experiments

We used the MUC coreference scoring metric (Vilain et al 1995) to evaluate³ our systems.

To establish an upper limit for the possible improvement offered by our models, we first did experiments using perfect (hand-tagged) mentions and perfect relations as inputs. The algorithms for

³ In our scoring, we use the ACE keys and only score mentions which appear in both the key and system response. This therefore includes only mentions identified as being in the ACE semantic categories by both the key and the system response. Thus these scores cannot be directly compared against coreference scores involving all noun phrases. (Ng 2005) applies another variation on the MUC metric to several systems tested on the ACE data by scoring all response mentions against all key mentions. For coreference systems that don't restrict themselves to mentions in the ACE categories (or that don't succeed in so restricting themselves), this scoring method could lead to some odd effects. For example, systems that recover more correct links could be penalized for this greater recall because all links involving non-ACE mentions will be incorrect according to the ACE key. For the sake of comparison, however, we present here English system results measured according to this metric: On newswire data, our baseline had an F of 62.8 and the rescoring method had an F of 64.2. Ng's best F score (on newspaper data) is 69.3. The best F score of the (Ng and Cardie 2002) system (also on newspaper data) is 62.1. On newswire data the (Ng 2005) system had an F score of 54.7 and the (Ng and Cardie 2002) system had an F score of 50.1. Note that Ng trained and tested these systems on different ACE data sets than those we used for our experiments.

these experiments are identical to those described above except for the omission of the relation tagger training. Tables 2 and 3 show the performance of the system for English and Chinese.

Performance	Recall	Precision	F-measure
Baseline	74.5	86.6	80.1
Rescoring	78.3	87.0	82.4

Table 2. Performance of English system with perfect mentions and perfect relations

Performance	Recall	Precision	F-measure
Baseline	87.5	83.2	85.3
Rescoring	88.8	84.7	86.7

Table 3. Performance of Chinese system with perfect mentions and perfect relations

We can see that the relation information provided some improvements for both languages. Relation information increased both recall and precision in both cases.

We then performed experiments to evaluate the impact of coreference rescoring when used with mentions and relations produced by the system. Table 4 and Table 5 list the results.⁴

Performance	Recall	Precision	F-measure
Baseline	77.2	87.3	81.9
Rescoring	80.3	87.5	83.7

Table 4. Performance of English system with system mentions and system relations

Performance	Recall	Precision	F-measure
Baseline	75.0	76.3	75.6
Rescoring	76.1	76.5	76.3

Table 5. Chinese system performance with system mentions and system relations

⁴ Note that, while English shows slightly less relative gain from rescoring when using system relations and mentions, all of these scores are higher than the perfect mention/perfect relation scores. This increase may be a byproduct of the fact that the system mention tagger output contains almost 8% fewer scoreable mentions than the perfect mention set (see footnote 3). With a difference of this magnitude, the particular mention set selected can be expected to have a sizable impact on the final scores.

The improvement provided by rescoring in trials using mentions and relations detected by the system is considerably less than the improvement in trials using perfect mentions and relations, particularly for Chinese. The performance of our relation tagger is the most likely cause for this difference. We would expect further gain after improving the relation tagger.

A sign test applied to a 5-way split of each of the test corpora indicated that for both languages, for both perfect and system mentions/relations, the system that exploited relation information significantly outperformed the baseline (at the 95% confidence level, judged by F measure).

6.3 Error Analysis

Errors made by the RCM rules reveal both the drawbacks of using a lightweight semantic representation and the inherent difficulty of semantic analysis. Consider the following instance:

Card's interest in politics began when he became **president of the class of 1965 at Holbrook High School**...In 1993, he became **president** and chief executive of **the American Automobile Manufacturers Association**, where he oversaw the lobbying against tighter fuel-economy and air pollution regulations for automobiles...

The two occurrences of “president” should corefer even though they have EMP-ORG/Employ-Executive relations with two different organizations. The relation rule (Rule 1) fails here because it doesn't take into account the fact that relations change over time (in this case, the same person filling different positions at different times). In these and other cases, a little knowledge is a dangerous thing: a more complete schema might be able to deal more thoroughly with temporal and other essential semantic dimensions.

Nevertheless, performance improvements indicate that the rewards of the RCM's simple semantic representation outweigh the risks.

7 Conclusion and Future Work

We have outlined an approach to improving reference resolution through the use of semantic relations, and have described a system which can exploit these semantic relations effectively. Our experiments on English and Chinese data showed

that these small inroads into semantic territory do indeed offer performance improvements. Furthermore, the method is low-cost and not domain-specific.

These experiments also suggest that some gains can be made through the exploration of new architectures for information extraction applications. The “resolve coreference, tag relations, resolve coreference” procedure described above could be seen as one and a half iterations of a “resolve coreference then tag relations” loop. Seen in this way, the system poses the question of whether further gains could be made by pushing the iterative approach further. Perhaps by substituting an iterative procedure for the pipeline architecture’s linear sequence of stages we can begin to address the knotty, mutually determining nature of the interaction between semantic relations and coreference relations. This approach could be applied more broadly, to different NLP tasks, and also more deeply, going beyond the simple one-and-a-half-iteration procedure we present here. Ultimately, we would want this framework to boost the performance of each component automatically and significantly.

We also intend to extend our method both to cross-document relation detection and to event detection.

Acknowledgements

This research was supported by the Defense Advanced Research Projects Agency under Grant N66001-04-1-8920 from SPAWAR San Diego, and by the National Science Foundation under Grant 03-25657. This paper does not necessarily reflect the position or the policy of the U.S. Government.

References

- David Bean, Ellen Riloff. 2004. Unsupervised learning of contextual role knowledge for coreference resolution. *Proc. HLT-NAACL 2004*, pp. 297-304.
- Daniel M. Bikel, Scott Miller, Richard Schwartz, and Ralph Weischedel. 1997. Nymble: A high-performance learning name-finder. *Proc. Fifth Conf. on Applied Natural Language Processing*, Washington, D.C., pp. 194-201.
- Carbonell, Jaime and Ralf Brown. 1988. Anaphora resolution: A multi-strategy approach. *Proc. COLING 1988*, pp.96-101

- Eugene Charniak. 1972. Toward a model of children's story comprehension. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.
- Niyu Ge, John Hale and Eugene Charniak. 1998. A statistical approach to anaphora resolution. *Proc. the Sixth Workshop on Very Large Corpora*.
- Jerry Hobbs, Mark Stickel, Douglas Appelt and Paul Martin. 1993. Interpretation as abduction. *Artificial Intelligence*, 63, pp. 69-142.
- Ruslan Mitkov. 2000. Towards a more consistent and comprehensive evaluation of anaphora resolution algorithms and systems. *Proc. 2nd Discourse Anaphora and Anaphora Resolution Colloquium*, pp. 96-107
- Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. *Proc. ACL 2002*, pp.104-111
- Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, Volume 27, Number 4, pp. 521-544
- Joel R. Tetreault. 2001. A corpus-based evaluation of centering and pronoun resolution. *Computational Linguistics*, Volume 27, Number 4, pp. 507-520
- Joel R. Tetreault and James Allen. 2004. Semantics, Dialogue, and Pronoun Resolution. *Proc. CATALOG '04* Barcelona, Spain.
- Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. *Proc. the 6th Message Understanding Conference (MUC-6)*. San Mateo, Cal. Morgan Kaufmann.
- Robert Wilensky. 1983. *Planning and Understanding*. Addison-Wesley.