DISCOURSE-NEW DETECTORS FOR DEFINITE DESCRIPTION RESOLUTION: A SURVEY AND A PRELIMINARY PROPOSAL

Massimo Poesio,[†] **Olga Uryupina,**[§] **Renata Vieira,**[*]
Mijail Alexandrov-Kabadjov[†] and **Rodrigo Goulart**[*]
[†]University of Essex, Computer Science and Cognitive Science (UK)
[§]Universität des Saarlandes, Computerlinguistik (Germany)
[*]Unisinos, Computação Aplicada (Brazil)

Abstract

Vieira and Poesio (2000) proposed an algorithm for definite description (DD) resolution that incorporates a number of heuristics for detecting discourse-new descriptions. The inclusion of such detectors was motivated by the observation that more than 50% of definite descriptions (DDs) in an average corpus are discourse new (Poesio and Vieira, 1998), but whereas the inclusion of detectors for non-anaphoric pronouns in algorithms such as Lappin and Leass' (1994) leads to clear improvements in precision, the improvements in anaphoric DD resolution (as opposed to classification) brought about by the detectors were rather small. In fact, Ng and Cardie (2002a) challenged the motivation for the inclusion of such detectors, reporting no improvements, or even worse performance. We re-examine the literature on the topic in detail, and propose a revised algorithm, taking advantage of the improved discourse-new detection techniques developed by Uryupina (2003).

1 Introduction

Although many theories of definiteness and many anaphora resolution algorithms are based on the assumption that definite descriptions are anaphoric, in fact in most corpora at least half of definite descriptions are DISCOURSE-NEW (Prince, 1992), as shown by the following examples, both of which are the first sentences of texts from the Penn Treebank.

(1) a. Toni Johnson pulls a tape measure across <u>the front of what was once a stately Victorian home.</u>

b. <u>The Federal Communications Commission</u> allowed American Telephone & Telegraph Co. to continue offering discount phone services for large-business customers and said it would soon re-examine its regulation of <u>the long-distance market</u>.

Vieira and Poesio (2000) proposed an algorithm for definite description resolution that incorporates a number of heuristics for detecting discourse-new (henceforth: DN) descriptions. But whereas the inclusion of detectors for non-anaphoric pronouns (e.g., *It* in *It's raining*) in algorithms such as Lappin and Leass' (1994) leads to clear improvements in precision, the improvements in anaphoric DD resolution (as opposed to classification) brought about by the detectors were rather small. In fact, Ng and Cardie (2002a) challenged the motivation for the inclusion of such detectors, reporting no improvements or even worse performance. We re-examine the literature on the topic in detail, and propose a revised algorithm, taking advantage of the improved DN detection techniques developed by Uryupina (2003).

2 Detecting Discourse-New Definite Descriptions

2.1 Vieira and Poesio

Poesio and Vieira (1998) carried out corpus studies indicating that in corpora like the Wall Street Journal portion of the Penn Treebank (Marcus et al., 1993), around 52% of DDs are discourse-new (Prince, 1992), and another 15% or so are bridging references, for a total of about 66-67% first-mention. These results led Vieira and Poesio to propose a definite description resolution algorithm incorporating independent heuristic strategies for recognizing DN definite descriptions (Vieira, 1998; Vieira and Poesio, 2000).

The heuristics proposed by Vieira and Poesio assumed a parsed input (the Penn Treebank) and aimed at identifying five categories of DDs licensed to occur as first mention on semantic or pragmatic grounds on the basis of work on definiteness including Loebner's account (1987):

1. So-called SEMANTICALLY FUNCTIONAL de-

scriptions (Loebner, 1987). This class included descriptions with modifiers like *first* or *best* that turned a possibly sortal predicate into a function (as in *the first person to cross the Pacific on a row boat*); as well as descriptions with predicates like *fact* or *belief* followed by a *that*-clause with the function of specifying the fact or belief under question. Both types of definites descriptions were recognized by consulting a hand-coded list of SPECIAL PREDICATES.

2. Descriptions serving as disguised PROPER NAMES, such as *The Federal Communications Commission* or *the Iran-Iraq war*. The heuristics for recognizing these definite descriptions were primarily based on capitalization (of the head or the modifiers).

3. PREDICATIVE descriptions, i.e., descriptions semantically functioning as predicates rather than as referring. These include descriptions occurring in appositive position (as in *Glenn Cox, the president of Phillips Petroleum*) and in certain copular constructions (as in *the man most likely to gain custody of all this is a career politician named Dinkins*). The heuristics used to recognize these cases examined the syntactic structure of the NP and the clause in which it appeared.

4. Descriptions ESTABLISHED (i.e., turned into functions in context) by restrictive modification, particularly by establishing relative clauses (Loebner, 1987) and prepositional phrases, as in *The hotel where we stayed last night was pretty good*. These heuristics, as well, examined the syntactic structure of the NP.

5. LARGER SITUATION definite descriptions (Hawkins, 1978), i.e., definite descriptions like *the sun*, *the pope* or *the long distance market* which denote uniquely on the grounds of shared knowledge about the situation (these are Loebner's 'situational functions'). Vieira and Poesio's system had a small list of such definites.

These heuristics were included as tests both of a decision tree concerned only with the task of DN detection, and of decision trees determining the classification of DDs as anaphoric, bridging or discourse new. In both cases, the DN detection tests were intertwined with attempts to identify an antecedent for such DDs. Both hand-coded decision trees and automatically acquired ones (trained using ID3, (Quinlan, 1986)) were used for the task of two-way classification into discourse-new and anaphoric. Vieira and Poesio found only small differences in the order of tests in the two decision trees, and small differences in performance. The hand-coded decision tree executes in the following order:

1. Try the DN heuristics with the highest accuracy (recognition of some types of semantically functional DDs using special predicates, and of potentially predicative DDs occurring in appositions);

2. Otherwise, attempt to resolve the DD as direct anaphora;

3. Otherwise, attempt the remaining DN heuristics in the order: proper names, descriptions established by relatives and PPs, proper name modification, predicative DDs occurring in copular constructions.

If none of these tests succeeds, the algorithm can either leave the DD unclassified, or classify it as DN. The automatically learned decision tree attempts direct anaphora resolution first. The overall results on the 195 DDs on which the automatically trained decision tree was tested are shown in Table 1. The baseline is the result achieved by classifying every DD as discourse-new–with 99 discourse-new DDs out of 195, this means a precision of 50.8%. Two results are shown for the hand-coded decision tree: in one version, the system doesn't attempt to classify all DDs; in the other, all unclassified DDs are classified as discourse-new.

Version of the System	P	R	F
Baseline	50.8	100	67.4
Discourse-new detection only	69	72	70
Hand-coded DT: partial	62	85	71.7
Hand-coded DT: total	77	77	77
ID3	75	75	75

Table 1: Overall results by Vieira and Poesio

2.2 Bean and Riloff

Bean and Riloff (1999) developed a system for identifying discourse-new DDs[1] that incorporates, in addition to syntax-based heuristics aimed at recognizing predicative and established DDs using postmodification heuristics similar to those used by Vieira and Poesio, additional techniques for mining from corpora unfamiliar DDs including proper names, larger situation, and semantically functional. Two

[1] Bean and Riloff use the term EXISTENTIAL for these DDs.

of the techniques proposed by Bean and Riloff are particularly worth noticing. The SENTENCE-ONE (S1) EXTRACTION heuristic identifies as discourse-new every DD found in the first sentence of a text. More general patterns can then be extracted from the DDs initially found by S1-extraction, using the EXISTENTIAL HEAD PATTERN method which, e.g., would extract `the N+ Government` from *the Salvadoran Government* and *the Guatemalan Government*. The DEFINITE ONLY (DO) list contained NPs like *the National Guard* or *the FBI* with a high DEFINITE PROBABILITY, i.e., whose nominal complex has been encountered at least 5 times with the definite article, but never with the indefinite. VACCINES were also developed that prevented the use of patterns identified by S1-extraction or DO-list elements when the definite probability of the definite was too low. Overall, the algorithm proposed by Bean and Riloff is as follows:

1. If the head noun of the DD appeared earlier in the text, classify as anaphoric.
2. Otherwise, if the DD occurs in the S1 list, classify as discourse-new unless stopped by vaccine.
3. Otherwise, classify the DD as DN if one of the following tests applies:
   (a) it occurs in the DO list;
   (b) it matches one of the EHP patterns, and is not stopped by vaccine;
   (c) it matches one of the syntactic heuristics
4. Otherwise, classify the DD as anaphoric.

(Note that as in the machine-learned version of the Vieira and Poesio decision tree, a (simplified) direct anaphora test is tried first, followed by DN detectors in decreasing order of accuracy.)

Bean and Riloff trained their system on 1600 articles from MUC-4, and tested it on 50 texts. The S1 extraction methods produced 849 DDs; the DO list contained 65 head nouns and 321 full NPs. The overall results are shown in Table 2; the baseline are the results obtained when classifying all DDs as discourse-new.

Although the overall precision is not better than what obtained with the partial hand-coded decision tree used by Vieira and Poesio, recall is substantially improved.

Method	R	P
Baseline	100	72.2
Syntactic Heuristics	43	93.1
Synt. Heuristics + S1	66.3	84.3
Synt. Heuristics + EHP	60.7	87.3
Synt. Heuristics + DO	69.2	83.9
Synt. Heuristics + S1 + EHP + DO	81.7	82.2
Synt. Heuristics + S1 + EHP + DO + V	79.1	84.5

Table 2: Discourse-new prediction results by Bean and Riloff

2.3 Ng and Cardie

Ng and Cardie (2002a) directly investigate the question of whether employing a discourse-new prediction component improves the performance of a coreference resolution system (specifically, the system discussed in (Ng and Cardie, 2002b)). Ng and Cardie's work differs from the work discussed so far in that their system attempts to deal with all types of NPs, not just definite descriptions.

The discourse-new detectors proposed by Ng and Cardie are statistical classifiers taking as input 37 features and trained using either C4.5 (Quinlan, 1993) or RIPPER (Cohen, 1995). The 37 features of a candidate anaphoric expression specify, in addition to much of the information proposed in previous work, a few new types of information about NPs.

- The four boolean so-called LEXICAL features are actually string-level features: for example, `str_match` is Y if a preceding NP string-matches the anaphoric expression (except for the determiner), and `head_match` = Y if a preceding NP's head string-matches the anaphoric expression's. `embedded`=Y if the anaphoric expression is a prenominal modifier.
- The second group of 11 (mostly boolean) features specifies the type of NP: e.g., `pronoun` is Y if the anaphoric expression is a pronoun, else N.
- The third group of 7 features specifies syntactic properties of the anaphoric expression, including number, whether NP_j is the first of two NPs in an appositive or predicative construction, whether NP_j is pre- or post-modified, whether it contains a proper noun, and whether it is modified by a superlative.
- The next group of 8 features are mostly novel, and capture information not used by previous DN detectors about the exact composition of definite descriptions: e.g., `the_2n`=Y if the anaphoric expression starts with determiner *the* followed by exactly two common nouns, `the_num_n`=Y if the anaphoric expression starts with determiner *the* followed

by a cardinal and a common noun, and `the_sing_n`=Y if the anaphoric expression starts with determiner *the* followed by a singular NP not containing a proper noun.

- The next group of features consists of 4 features capturing a variety of 'semantic' information, including whether a previous NP is an 'alias' of NP_j, or whether NP_j is the title of a person (*the president*).

- Finally, the last three features capture information about the position in the text in which NP_j occurs: the header, the first sentence, or the first paragraph.

Ng and Cardie's discourse-new predictor was trained and tested over the MUC-6 and MUC-7 coreference data sets, achieving accuracies of 86.1% and 84%, respectively, against a baseline of 63.8% and 73.2%, respectively. Inspection of the top parts of the decision tree produced with the MUC-6 suggests that `head_match` is the most important feature, followed by the features specifying NP type, the `alias` feature, and the features specifying the structure of definite descriptions.

Ng and Cardie discuss two architectures for the integration of a DN detector in a coreference system. In the first architecture, the DN detector is run first, and the coreference resolution algorithm is run only if the DN detector classifies that NP as anaphoric. In the second architecture, the system first computes `str_match` and `alias`, and runs the anaphoric resolver if any of them is Y; otherwise, it proceeds as in the first architecture. The results obtained on the MUC-6 data with the baseline anaphoric resolver, the anaphoric resolver augmented by a DN detector as in the first architecture, and as in the second architecture (using C4.5), are shown in Table 3. The results for all NPs, pronouns only, proper names only, and common nouns only are shown.[2]

As indicated in the Table, running the DN detector first leads to worse results–this is because the detector misclassifies a number of anaphoric NPs as non-anaphoric. However, looking first for a same-head antecedent leads to a statistically significant improvement over the results of the baseline anaphoric resolver. This confirms the finding both of Vieira and Poesio and of Bean and Riloff that the direct anaphora should be called very early.

[2]It's not clear to us why the overall performance of the algorithm is much better than the performance on the three individual types of anaphoric expressions considered–i.e., which other anaphoric expressions are handled by the coreference resolver.

	MUC-6			MUC-7		
	R	P	F	R	P	F
Baseline (no DN detector)	70.3	58.3	63.8	65.5	58.2	61.6
Pronouns	17.9	66.3	28.2	10.2	62.1	17.6
Proper names	29.9	84.2	44.1	27.0	77.7	40.0
Common nouns	25.2	40.1	31.0	26.6	45.2	33.5
DN detector runs first	57.4	71.6	63.7	47.0	77.1	58.4
Pronouns	17.9	67.0	28.2	10.2	62.1	17.6
Proper names	26.6	89.2	41.0	21.5	84.8	34.3
Common nouns	15.4	56.2	24.2	13.8	77.5	23.4
Same head runs first	63.4	68.3	65.8	59.7	69.3	64.2
Pronouns	17.9	67.0	28.2	10.2	62.1	17.6
Proper names	27.4	88.5	41.9	26.1	84.7	40.0
Common nouns	20.5	53.1	29.6	21.7	59.0	31.7

Table 3: Evaluation of the three anaphoric resolvers discussed by Ng and Cardie.

2.4 Uryupina

Uryupina (2003) trained two separate classifiers (using RIPPER, (Cohen, 1995)): a DN detector and a UNIQUENESS DETECTOR, i.e., a classifier that determines whether an NP refers to a unique object. This is useful to identify proper names (like *1998*, or *the United States of America*), semantic definites (like *the chairman of Microsoft*) and larger situation definite descriptions (like *the pope*). Both classifiers use the same set of 32 features. The features of an NP encode, first, of all, string-level information: e.g., whether the NP contains capitalized words, digits, or special symbols. A second group of features specifies syntactic information: whether the NP is postmodified, and whether it contains an apposition. Two types of appositions are distinguished, with and without commas. CONTEXT features specify the distance between the NP and the previous NP with the same head, if any. Finally, Uryupina's system computes four features specifying the NP's definite probability. Unlike the definite probability used by Bean and Riloff, these features are computed from the Web, using Altavista. From each NP, its head H and entire NP without determiner Y are determined, and four ratios are then computed:

$$\frac{\#\text{"the Y"}}{\#Y}, \quad \frac{\#\text{"the Y"}}{\#\text{"}aY\text{"}}, \quad \frac{\#\text{"the H"}}{\#H}, \quad \frac{\#\text{"the H"}}{\#\text{"}aH\text{"}}.$$

The classifiers were tested on 20 texts from MUC-7 (a subset of the second data set used by Ng and Cardie), parsed by Charniak's parser. 19 texts were used for training and for tuning RIPPER's parameters, one for testing. The results for the discourse new detection task are shown in Table 4, separating the results for all NPs and definite NPs only, and the results without definite probabilities and including them. The results for uniqueness detection are shown in Table 4, in which the results obtained by prioritizing precision and recall are shown separately.

	Features	P	R	F
All NPs	String+Syn+Context	87.9	86.0	86.9
	All	88.5	84.3	86.3
Def NPs	String+Syn+Context	82.5	79.3	80.8
	All	84.8	82.3	83.5

Table 4: Results of Uryupina's discourse new classifier

	Features	P	R	F
Best Prec	String+Syn+Context	94.0	84.0	88.7
	All	95.0	83.5	88.9
Best Rec	String+Syn+Context	86.7	96.0	91.1
	All	87.2	97.0	91.8

Table 5: Results of Uryupina's uniqueness classifier

The first result to note is that both of Uryupina's classifiers work very well, particularly the uniqueness classifier. These tables also show that the definite probability helps somewhat the discourse new detector, but is especially useful for the uniqueness detector, as one would expect on the basis of Loebner's discussion.

2.5 Summary

Quite a lot of consensus on many of the factors playing a role in DN detection for DDs. Most of the algorithms discussed above incorporate methods for:

- recognizing predicative DDs;
- recognizing discourse-new proper names;
- identifying functional DDs;
- recognizing DDs modified by establishing relatives (which may or may not be discourse-new).

There is also consensus on the fact that DN detection cannot be isolated from anaphoric resolution (witness the Ng and Cardie results).

One problem with some of the machine learning approaches to coreference is that these systems do not achieve very good results on pronoun and definite description resolution in comparison with specialized algorithms: e.g., although Ng and Cardie's best version achieves F=65.8 on all anaphoric expressions, it only achieves F=29.6 for definite descriptions (cfr. Vieira and Poesio's best result of F=77), and F=28.2 for pronouns (as opposed to results as high as F=80 obtained by the pronoun resolution algorithms evaluated in (Tetreault, 2001)). Clearly these systems can only be properly compared by evaluating them all on the same corpora and the same data, and discussion such as (Mitkov, 2000) suggest caution in interpreting some of the results discussed in the literature as pre- and post-processing often plays a crucial role, but we feel that evaluating DN detectors in conjunction with high-performing systems would give a better idea of the improvements that one may hope to achieve.

3 Do Discourse-New Detectors Help? Preliminary Evaluations

Vieira and Poesio did not test their system without DN-detection, but Ng and Cardie's results indicate that DN detection does improve results, if not dramatically, provided that the `same_head` test is run first–although their DN detector does not appear to improve results for pronouns, the one category for which detection of non-anaphoricity has been shown to be essential (Lappin and Leass, 1994). In order to evaluate how much improvement can we expect by just improving the DN detector, we did a few preliminary evaluations both with a reimplementation of Vieira and Poesio's algorithm which does not include a discourse-new detector, running over treebank text as the original algorithm, and with a simple statistical coreference resolver attempting to resolve all anaphoric expressions and running over unparsed text, using Uryupina's features for discourse-new detection, and over the same corpus used by Ng and Cardie (MUC-7).

3.1 How much does DN-detection help the Vieira / Poesio algorithm?

GUITAR (Poesio and Alexandrov-Kabadjov, 2004) is a general-purpose anaphoric resolver that includes an implementation of the Vieira / Poesio algorithm for definite descriptions and of Mitkov's algorithm for pronoun resolution (Mitkov, 1998). It is implemented in Java, takes its input in XML format and returns as output its input augmented with the anaphoric relations it has discovered. GUITAR has been implemented in such a way as to be fully *modular*, making it possible, for example, to replace the DD resolution method with alternative implementations. It includes a pre-processor incorporating a chunker so that it can run over both hand-parsed and raw text.

A version of GUITAR without the DN detection aspects of the Vieira / Poesio algorithm was evaluated on the GNOME corpus (Poesio, 2000; Poesio et al., 2004), which contains 554 definite descriptions, of which 180 anaphoric, and 305 third-person pronouns, of which 217 anaphoric. The results for definite descriptions over hand-parsed text are shown in Table 6.

Total	Res	Corr	NM	WM	SM	R	P	F
180	182	121	43	16	45	67.2	66.5	66.8

Table 6: Evaluation of the GUITAR system without DN detection over a hand-annotated treebank

GUITAR without a DN recognizer takes 182 DDs (Res) as anaphoric, resolving 121 of them correctly (Corr); of the 182 DDs it attempts to resolve, only 16 are incorrectly resolved (WM); almost three times that number (45) are Spurious Matches (SM), i.e., discourse-new DDs incorrectly interpreted as anaphoric. (Res=Corr+WM+SM.) The system can't find an antecedent for 43 of the 180 anaphoric DDs. When endowed with a perfect DN detector, GUITAR could achieve a precision P=88.3 which, assuming recall stays the same (R=67.2) would mean a F=76.3.

Of course, these results are obtained assuming perfect parsing. For a fairer comparison with the results of Ng and Cardie, we report in Table 7 the results for both pronouns and definite descriptions obtained by running GUITAR off raw text.

	R	P	F
Pronouns	65.5	63.0	64.2
DDs	56.7	56.1	56.4

Table 7: Evaluation of the GUITAR system without DN detection off raw text

Notice that although these results are not particularly good, they are still better than the results reported by Ng and Cardie for pronouns and definite NPs.

3.2 How much might DN detection help a simple statistical coreference resolver?

In order to have an even closer comparison with the results of Ng and Cardie, we implemented a simple statistical coreference system, that, like Ng and Cardie's system, would resolve all types of anaphoric expressions, and would run over unparsed text, but without DN detection. We ran the system over the MUC-7 data used by Ng and Cardie, and compared the results with those obtained by using perfect knowledge about discourse novelty. The results are shown in Table 8.

	R	P	F
Without DN detection	44.7	54.9	49.3
With DN detection	41.4	80.0	54.6

Table 8: Using an oracle

These results suggest that a DN detector could lead to substantial improvements for coreference resolution in general: DN detection might improve precision by more than 30%, which more than makes up for the slight deterioration in recall. Of course, this test alone doesn't tell us how much improvement DN detection would bring to a higher-performance anaphoric resolver.

4 A New Set of Features for Discourse-New Detection

Next, we developed a new set of features for discourse new detection that takes into account the findings of the work on DN detection discussed in the previous sections. This set of features will be input to an anaphoric resolver for DDs working in two steps. For each DD,

1. The direct anaphora resolution algorithm from (Vieira and Poesio, 2000) is run, which attempts to find an head-matching antecedent within a given window and taking premodification into account. The results of the algorithm (i.e., whether an antecedent was found) is used as one of the input features of the classifier in the next step. In addition, a number of features of the DD that may help recognizing the classes of DDs discussed above are extracted from the input. Some of these features are computed accessing the Web via the Google API.
2. A decision tree classifier is used to classify the DD as anaphoric (in which case the antecedents identified at the first step are also returned) or discourse-new.

The features input to the classifier can be categorized as follows:

Anaphora A single feature, `direct-anaphora`, specifying the distance of the (same-head) antecedent from the DD, if any (values: `none`, `zero`, `one`, `more`)

Predicative NPs Two boolean features:

- `apposition`, if the DD occurs in appositive position;
- `copular`, if the DD occurs in post-verbal position in a copular construction.

Proper Names Three boolean features:

- `c-head`: whether the head is capitalized;
- `c-premod`: whether one of the premodifiers is capitalized;
- `S1`: whether the DD occurs in the first sentence of a Web page.

Functionality The four definite probabilities used by Uryupina (computed accessing the Web), plus a `superlative` feature specifying if one of the premodifiers is a superlative, extracted from the part of speech tags.

Establishing relative A single feature, specifying whether NP is postmodified, and by a relative clause or a prepositional phrase;

Text Position Whether the DD occurs in the title, the first sentence, or the first paragraph.

We are testing several classifiers included in the Weka 3.4 library (`http://www.cs.waikato.ac.nz/~ml/`) including an implementation of C4.5 and a multi-layer perceptron.

5 Evaluation

Data We are using three corpora for the evaluation, including texts from different genres, in which all anaphoric relations between (all types of) NPs are marked. The GNOME corpus includes pharmaceutical leaflets and museum 'labels' (i.e., descriptions of museum objects and of the artists that realized them). As said above, the corpus contains 554 definite descriptions. In addition, we are using the 14 texts from the Penn Treebank included in the corpus used by Vieira and Poesio. We transferred these texts to XML format, and added anaphoric information for all types of NPs according to the GNOME scheme. Finally, we are testing the system on the MUC-7 data used by Ng and Cardie

Methods We will compare three versions of the DD resolution component:

1. The baseline algorithm without DN detection incorporated in GUITAR described above (i.e., only the direct anaphora resolution part of (Vieira and Poesio, 2000));
2. A complete implementation of the Vieira and Poesio algorithm, including also the DN detecting heuristics;
3. An algorithm using the statistical classifier discussed above.

Results Regrettably, the system is still being tested. We will report the results at the workshop.

6 Discussion and Conclusions

Discussions and conclusions will be based on the final results.

Acknowledgments

Mijail Alexandrov-Kabadjov is supported by Conacyt. Renata Vieira and Rodrigo Goulart are partially supported by CNPq.

References

D. L. Bean and E. Riloff. 1999. Corpus-based identification of non-anaphoric noun phrases. In *Proc. of the 37th ACL*, pages 373–380, University of Maryland. ACL.

W. Cohen. 1995. Fast effective rule induction. In *Proc. of ICML*.

J. A. Hawkins. 1978. *Definiteness and Indefiniteness*. Croom Helm, London.

S. Lappin and H. J. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–562.

S. Loebner. 1987. Definites. *Journal of Semantics*, 4:279–326.

M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a large annotated corpus of english: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

R. Mitkov. 1998. Robust pronoun resolution with limited knowledge. In *Proc. of the 18th COLING*, pages 869–875, Montreal.

R. Mitkov. 2000. Towards more comprehensive evaluation in anaphora resolution. In *Proc. of the 2nd International Conference on Language Resources and Evaluation*, pages 1309–1314, Athens, May.

V. Ng and C. Cardie. 2002a. Identifying anaphoric and non-anaphoric noun phrases to improve coreference resolution. In *Proc. of 19th COLING*.

V. Ng and C. Cardie. 2002b. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Meeting of the ACL*.

M. Poesio and M. Alexandrov-Kabadjov. 2004. A general-purpose, off the shelf anaphoric resolver. In *Proc. of LREC*, Lisbon, May.

M. Poesio and R. Vieira. 1998. A corpus-based investigation of definite description use. *Computational Linguistics*, 24(2):183–216, June. Also available as Research Paper CCS-RP-71, Centre for Cognitive Science, University of Edinburgh.

M. Poesio, R. Stevenson, B. Di Eugenio, and J. M. Hitzeman. 2004. Centering: A parametric theory and its instantiations. *Computational Linguistics*. To appear.

M. Poesio. 2000. Annotating a corpus to develop and evaluate discourse entity realization algorithms: issues and preliminary results. In *Proc. of the 2nd LREC*, pages 211–218, Athens, May.

E. F. Prince. 1992. The ZPG letter: subjects, definiteness, and information status. In S. Thompson and W. Mann, editors, *Discourse description: diverse analyses of a fund-raising text*, pages 295–325. John Benjamins.

J. R. Quinlan. 1986. Induction of decision trees. *Machine Learning*, 1(1):81–106.

J. R. Quinlan. 1993. *C4.5: programs for machine learning*. Morgan Kaufmann, San Mateo, CA.

J. R. Tetreault. 2001. A corpus-based evaluation of centering and pronoun resolution. *Computational Linguistics*, 27(4):507–520.

O. Uryupina. 2003. High-precision identification of discourse-new and unique noun phrases. In *Proc. of the ACL 2003 Student Workshop*, pages 80–86.

R. Vieira and M. Poesio. 2000. An empirically-based system for processing definite descriptions. *Computational Linguistics*, 26(4), December.

R. Vieira. 1998. *Definite Description Resolution in Unrestricted Texts*. Ph.D. thesis, University of Edinburgh, Centre for Cognitive Science, February.