Evaluation tool for rule-based anaphora resolution methods

Catalina Barbu
School of Humanities, Languages
and Social Sciences
University of Wolverhampton
Stafford Street
Wolverhampton WV1 1SB
United Kingdom
c.barbu@wlv.ac.uk

Ruslan Mitkov
School of Humanities, Languages
and Social Sciences
University of Wolverhampton
Stafford Street
Wolverhampton WV1 1SB
United Kingdom
r.mitkov@wlv.ac.uk

Abstract

In this paper we argue that comparative evaluation in anaphora resolution has to be performed using the same pre-processing tools and on the same set of data. The paper proposes an evaluation environment for comparing anaphora resolution algorithms which is illustrated by presenting the results of the comparative evaluation of three methods on the basis of several evaluation measures.

1 Introduction

The evaluation of any NLP algorithm or system should indicate not only its efficiency or performance, but should also help us discover what a new approach brings to the current state of play in the field. To this end, a comparative evaluation with other well-known or similar approaches would be highly desirable.

We have already voiced concern (Mitkov, 1998a), (Mitkov, 2000b) that the evaluation of anaphora resolution algorithms and systems is bereft of any common ground for comparison due not only to the difference of the evaluation data, but also due to the diversity of pre-processing tools employed by each anaphora resolution system. The evaluation picture would not be accurate even if we compared anaphora resolution systems on the basis of the same data since the pre-processing errors which would be carried over to the systems' outputs might vary. As a way forward we have proposed the idea of the *evaluation workbench* (Mitkov, 2000b) - an open-ended architecture which allows the incorporation of different algorithms and their comparison on the basis of the same pre-processing tools and the same data. Our paper discusses a particular configuration of this new evaluation environment incorporating three approaches sharing a common ”knowledge-poor philosophy”: Kennedy and Boguraev's (1996) parser-free algorithm, Baldwin's (1997) CogNiac and Mitkov's (1998b) knowledge-poor approach.

2 The evaluation workbench for anaphora resolution

In order to secure a ”fair”, consistent and accurate evaluation environment, and to address the problems identified above, we have developed an evaluation workbench for anaphora resolution which allows the comparison of anaphora resolution approaches sharing common principles (e.g. similar pre-processing or resolution strategy). The workbench enables the ”plugging in” and testing of anaphora resolution algorithms on the basis of the same pre-processing tools and data. This development is a time-consuming task, given that we have to re-implement most of the algorithms, but it is expected to achieve a clearer assessment of the advantages and disadvantages of the different approaches. Developing our own evaluation environment (and even reimplementing some of the key algorithms) also alleviates the impracticalities associated with obtaining the codes of original programs.

Another advantage of the evaluation

workbench is that all approaches incorporated can operate either in a fully automatic mode or on human annotated corpora. We believe that this is a consistent way forward because it would not be fair to compare the success rate of an approach which operates on texts which are perfectly analysed by humans, with the success rate of an anaphora resolution system which has to process the text at different levels before activating its anaphora resolution algorithm. In fact, the evaluations of many anaphora resolution approaches have focused on the accuracy of resolution algorithms and have not taken into consideration the possible errors which inevitably occur in the pre-processing stage. In the real-world, fully automatic resolution must deal with a number of hard pre-processing problems such as morphological analysis/POS tagging, named entity recognition, unknown word recognition, NP extraction, parsing, identification of pleonastic pronouns, selectional constraints, etc. Each one of these tasks introduces errors and thus contributes to a drop in the performance of the anaphora resolution system.[1] As a result, the vast majority of anaphora resolution approaches rely on some kind of pre-editing of the text which is fed to the resolution algorithm, and some of the methods have only been manually simulated. By way of illustration, Hobbs' naive approach (1976; 1978) was not implemented in its original version. In (Dagan and Itai, 1990; Dagan and Itai, 1991; Aone and Bennett, 1995; Kennedy and Boguraev, 1996) pleonastic pronouns are removed manually[2] , whereas in (Mitkov, 1998b; Ferrandez et al., 1997) the outputs of the part-of-speech tagger and the NP extractor/ partial parser are post-edited similarly to Lappin and Leass (1994) where the output of the Slot Unification Grammar parser is corrected manually. Finally, Ge at al's (1998) and Tetrault's systems (1999) make use of annotated corpora and thus do not perform any pre-processing. One of the very few systems[3] that is fully automatic is MARS, the latest version of Mitkov's knowledge-poor approach implemented by Evans. Recent work on this project has demonstrated that fully automatic anaphora resolution is more difficult than previous work has suggested (Orăsan et al., 2000).

2.1 Pre-processing tools

Parser

The current version of the evaluation workbench employs one of the high performance "super-taggers" for English - Conexor's FDG Parser (Tapanainen and Järvinen, 1997). This super-tagger gives morphological information and the syntactic roles of words (in most of the cases). It also performs a surface syntactic parsing of the text using dependency links that show the head-modifier relations between words. This kind of information is used for extracting complex NPs.

In the table below the output of the FDG parser run over the sentence: "This is an input file." is shown.

```
1 This  this subj:>2    @SUBJ PRON SG
2 is     be main:>0 @+FMAINV V
3 an     an det:>5 @DN> DET SG
4 input input    attr:>5 @A> N SG
5 file file comp:>2 @PCOMPL-S N SG
    $.
$<s>
```

Example 1: FDG output for the text *This is an input file.*

Noun phrase extractor

Although FDG does not identify the noun phrases in the text, the dependencies established between words have played an important role in building a noun phrase extractor. In the example above, the dependency relations help identifying the sequence "an input file". Every noun phrase is associated with some features as identified by FDG (number, part of speech, grammatical function) and also the linear position of the verb that they are arguments of, and the number of the sentence they appear in. The result of the NP

[1] For instance, the accuracy of tasks such as robust parsing and identification of pleonastic pronouns is far below 100% See (Mitkov, 2001) for a detailed discussion.

[2] In addition, Dagan and Itai (1991) undertook additional pre-editing such as the removal of sentences for which the parser failed to produce a reasonable parse, cases where the antecedent was not an NP etc.; Kennedy and Boguraev (1996) manually removed 30 occurrences of pleonastic pronouns (which could not be recognised by their pleonastic recogniser) as well as 6 occurrences of *it* which referred to a VP or prepositional constituent.

[3] Apart from MUC coreference resolution systems which operated in a fully automatic mode.

extractor is an XML annotated file. We chose this format for several reasons: it is easily read, it allows a unified treatment of the files used for training and of those used for evaluation (which are already annotated in XML format) and it is also useful if the file submitted for analysis to FDG already contains an XML annotation; in the latter case, keeping the FDG format together with the previous XML annotation would lead to a more difficult processing of the input file. It also keeps the implementation of the actual workbench independent of the pre-processing tools, meaning that any shallow parser can be used instead of FDG, as long as its output is converted to an agreed XML format.

An example of the overall output of the pre-processing tools is given below.

```
<P><S><w ID=0 SENT=0 PAR=1 LEMMA="this" DEP="2"
GFUN="SUBJ" POS="PRON" NR="SG">This</w><w ID=1
SENT=0 PAR=1 LEMMA="be" DEP="0" GFUN="+FMAINV"
POS="V"> is </w><COREF ID="ref1"><NP> <w ID=2
SENT=0 PAR=1 LEMMA="an" DEP="5" GFUN="DN" POS="DET"
NR="SG">an </w> <w ID=3 SENT=0 PAR=1 LEMMA="input"
DEP="5" GFUN="A" POS="N" NR="SG">input</w><w ID=4
SENT=0 PAR=1 LEMMA="file" DEP="2" GFUN="PCOMPL"
POS="N" NR="SG">file</w> </NP></COREF><w ID=5
SENT=0 PAR=1 LEMMA="." POS="PUNCT">.</w> </s>
<s><COREF ID="ref2" REF="ref1"><NP><w ID=0 SENT=1
PAR=1 LEMMA="it" DEP="2" GFUN="SUBJ" POS="PRON"> It
</w></NP></COREF> <w ID=1 SENT=1 PAR=1 LEMMA="be"
DEP="3" GFUN="+FAUXV" POS="V">is </w><w ID=2 SENT=1
PAR=1 LEMMA="use" DEP="0" GFUN="-FMAINV" POS="EN">
used</w><w ID=3 SENT=1 PAR=1 LEMMA="for" DEP="3"
GFUN="ADVL" POS="PREP">for</w> <NP><w ID=4 SENT=1
PAR=1 LEMMA="evaluation" DEP="4" GFUN="PCOMP"
POS="N"> evaluation</w></NP> <w ID=5 SENT=0 PAR=1
LEMMA="." POS="PUNCT">.</w></s></p>
```

Example 2: File obtained as result of the pre-processing stage (includes previous coreference an-notation) for the text *This is an input file. It is used for evaluation.*

2.2 Shared resources

The three algorithms implemented receive as input a representation of the input file. This representation is generated by running an XML parser over the file resulting from the pre-processing phase. A list of noun phrases is explicitly kept in the file representation. Each entry in this list consists of a record containing:

- the word form
- the lemma of the word or of the head of the noun phrase
- the starting position in the text
- the ending position in the text
- the part of speech
- the grammatical function
- the index of the sentence that contains the referent
- the index of the verb whose argument this referent is

Each of the algorithms implemented for the workbench enriches this set of data with information relevant to its particular needs. Kennedy and Boguraev (1996), for example, need additional information about whether a certain discourse referent is embedded or not, plus a pointer to the COREF class associated to the referent, while Mitkov's approach needs a score associated to each noun phrase.

Apart from the pre-processing tools, the implementation of the algorithms included in the workbench is built upon a common programming interface, which allows for some basic processing functions to be shared as well. An example is the morphological filter applied over the set of possible antecedents of an anaphor.

2.3 Usability of the workbench

The evaluation workbench is easy to use. The user is presented with a friendly graphical interface that helps minimise the effort involved in preparing the tests. The only information she/he has to enter is the address (machine and directory) of the FDG parser and the file annotated with coreferential links to be processed. The results can be either specific to each method or specific to the file submitted for processing, and are displayed separately for each method. These include lists of the pronouns and their identified antecedents in the context they appear as well as information as to whether they were correctly solved or not. In addition, the values obtained for the four evaluation measures (see section 3.2) and several statistical results characteristic of each method (e.g. average number of candidates for antecedents per anaphor) are computed. Separately, the statistical values related to the annotated file are displayed in a table. We should note that (even though this is not the intended usage of the workbench) a user can also submit unannotated files for processing. In this case, the algorithms display the antecedent found for each pronoun, but no automatic evaluation can be carried out due to the lack of annotated testing data.

2.4 Envisaged extensions

While the workbench is based on the FDG shallow parser at the moment, we plan to update the environment in such a way that two different modes will be available: one making use of a shallow parser (for approaches operating on partial analysis) and one employing a full parser (for algorithms making use of full analysis). Future versions of the workbench will include access to semantic information (WordNet) to accommodate approaches incorporating such types of knowledge.

3 Comparative evaluation of knowledge-poor anaphora resolution approaches

The first phase of our project included comparison of knowledge-poorer approaches which share a common pre-processing philosophy. We selected for comparative evaluation three approaches extensively cited in the literature: Kennedy and Boguraev's parser-free version of Lappin and Leass' RAP (Kennedy and Boguraev, 1996), Baldwin's pronoun resolution method (Baldwin, 1997) and Mitkov's knowledge-poor pronoun resolution approach (Mitkov, 1998b). All three of these algorithms share a similar pre-processing methodology: they do not rely on a parser to process the input and instead use POS taggers and NP extractors; nor do any of the methods make use of semantic or real-world knowledge. We re-implemented all three algorithms based on their original description and personal consultation with the authors to avoid misinterpretations. Since the original version of CogNiac is non-robust and resolves only anaphors that obey certain rules, for fairer and comparable results we implemented the ”resolve-all” version as described in (Baldwin, 1997). Although for the current experiments we have only included three knowledge-poor anaphora resolvers, it has to be emphasised that the current implementation of the workbench does not restrict in any way the number or the type of the anaphora resolution methods included. Its modularity allows any such method to be added in the system, as long as the pre-processing tools necessary for that method are available.

3.1 Brief outline of the three approaches

All three approaches fall into the category of factor-based algorithms which typically employ a number of factors (preferences, in the case of these three approaches) after morphological agreement checks.

Kennedy and Boguraev

Kennedy and Boguraev (1996) describe an algorithm for anaphora resolution based on Lappin and Leass' (1994) approach but without employing deep syntactic parsing. Their method has been applied to personal pronouns, reflexives and possessives. The general idea is to construct coreference equivalence classes that have an associated value based on a set of ten factors. An attempt is then made to resolve every pronoun to one of the previous introduced discourse referents by taking into account the salience value of the class to which each possible antecedent belongs.

Baldwin's Cogniac

CogNiac (Baldwin, 1997) is a knowledge-poor approach to anaphora resolution based on a set of high confidence rules which are successively applied over the pronoun under consideration. The rules are ordered according to their importance and relevance to anaphora resolution. The processing of a pronoun stops when one rule is satisfied. The original version of the algorithm is non-robust, a pronoun being resolved only if one of the rules is applied. The author also describes a robust extension of the algorithm, which employs two more weak rules that have to be applied if all the others fail.

Mitkov's approach

Mitkov's approach (Mitkov, 1998b) is a robust anaphora resolution method for technical texts which is based on a set of boosting and impeding indicators applied to each candidate for antecedent. The boosting indicators assign a positive score to an NP, reflecting a positive likelihood that it is the antecedent of the current pronoun. In contrast, the impeding ones apply a negative score to an NP, reflecting a lack of confidence that it is the antecedent of the current pronoun. A score is calculated based on these indicators and the discourse referent with the highest aggregate value is selected as antecedent.

3.2 Evaluation measures used

The workbench incorporates an automatic scoring system operating on an XML input file where the correct antecedents for every anaphor have been marked. The annotation scheme recognised by the system at this moment is MUC, but support for the MATE annotation scheme is currently under developement as well.

We have implemented four measures for evaluation: precision and recall as defined by Aone and Bennett (1995)[4] as well as success rate and critical success rate as defined in (Mitkov, 2000a). These four measures are calculated as follows:

- Precision = number of correctly resolved anaphor / number of anaphors attempted to be resolved

- Recall = number of correctly resolved anaphors / number of all anaphors identified by the system

- Success rate = number of correctly resolved anaphors / number of all anaphors

- Critical success rate = number of correctly resolved anaphors / number of anaphors with more than one antecedent after a morphological filter was applied

The last measure is an important criterion for evaluating the efficiency of a factor-based anaphora resolution algorithm in the ”critical cases” where agreement constraints alone cannot point to the antecedent. It is logical to assume that good anaphora resolution approaches should have high critical success rates which are close to the overall success rates. In fact, in most cases it is really the critical success rate that matters: high critical success rates naturally imply high overall success rates.

Besides the evaluation system, the workbench also incorporates a basic *statistical calculator* which addresses (to a certain extent) the question as to how reliable or realistic the obtained performance figures are - the latter depending on the nature of the data used for evaluation. Some evaluation data may contain anaphors which are more difficult to resolve, such as anaphors that are (slightly) ambiguous and require real-world knowledge for their resolution, or anaphors that have a high number of competing candidates, or that have their antecedents far away both in terms of sentences/clauses and in terms of number of ”intervening” NPs etc. Therefore, we suggest that in addition to the evaluation results, information should be provided in the evaluation data as to how difficult the anaphors are to resolve.[5] To this end, we are working towards the development of suitable and practical measures for quantifying the average ”resolution complexity” of the anaphors in a certain text. For the time being, we believe that simple statistics such as the number of anaphors with more than one candidate, and more generally, the average number of candidates per anaphor, or statistics showing the average distance between the anaphors and their antecedents, could serve as initial quantifying measures (see Table 2). We believe that these statistics would be more indicative of how ”easy” or ”difficult” the evaluation data is, and should be provided in addition to the information on the numbers or types of anaphors (e.g. intrasentential vs. intersentential) occurring or coverage (e.g. personal, possessive, reflexive pronouns in the case of pronominal anaphora) in the evaluation data.

3.3 Evaluation results

We have used a corpus of technical texts manually annotated for coreference. We have decided on

[4] This definition is slightly different from the one used in (Baldwin, 1997) and (Gaizauskas and Humphreys, 2000). For more discussion on this see (Mitkov, 2000a; Mitkov, 2000b).

[5] To a certain extent, the critical success rate defined above addresses this issue in the evaluation of anaphora resolution algorithms by providing the success rate for the anaphors that are more difficult to resolve.

File	Number of words	Number of pronouns	Anaphoric pronouns	Success Rate			Precision		
				Mitkov	Cogniac	K&B	Mitkov	Cogniac	K&B
ACC	9617	182	160	52.34%	45.0%	55.0%	42.85%	37.18%	48.35%
WIN	2773	51	47	55.31%	44.64%	63.82%	50.98%	41.17%	58.82%
BEO	6392	92	70	48.57%	42.85%	55.71%	36.95%	32.60%	42.39%
CDR	9490	97	85	71.76%	67.05%	74.11%	62.88%	58.76%	64.95%
Total	28272	422	362	56.9%	49.72%	61.6%	48.81%	42.65%	52.84%

Table 1: Evaluation results

File	Pronouns	Personal	Possesive	Reflexive	Intrasentential anaphors	Average referential distance		Average no of antecedents
						Sentences	NPs	
ACC	182	161	18	3	90	1.2	4.2	9.4
WIN	51	40	11	0	41	1.1	4.1	11.9
BEO	92	74	18	0	56	1.4	5.1	12.9
CDR	97	85	10	2	54	1.4	3.7	9.2
Total	422	360	57	5	241	1.275	4.275	10.85

Table 2: Statistical results

this genre because both Kennedy&Boguraev and Mitkov report results obtained on technical texts.

The corpus contains 28,272 words, with 19,305 noun phrases and 422 pronouns, out of which 362 are anaphoric. The files that were used are: ”Beowulf HOW TO” (referred in Table 1 as BEO), ”Linux CD-Rom HOW TO” (CDR), ”Access HOW TO” (ACC), ”Windows Help file” (WIN). The evaluation files were pre-processed to remove irrelevant information that might alter the quality of the evaluation (tables, sequences of code, tables of contents, tables of references). The texts were annotated for full coreferential chains using a slightly modified version of the MUC annotation scheme. All instances of identity-of-reference direct nominal anaphora were annotated. The annotation was performed by two people in order to minimize human errors in the testing data (see (Mitkov et al., 2000) for further details).

Table 1 describes the values obtained for the success rate and precision[6] of the three anaphora resolvers on the evaluation corpus. The overall success rate calculated for the 422 pronouns found in the texts was 56.9% for Mitkov's method, 49.72% for Cogniac and 61.6% for Kennedy and Boguraev's method.

Table 2 presents statistical results on the evaluation corpus, including distribution of pronouns, referential distance, average number of candidates for antecedent per pronoun and types of anaphors.[7]

As expected, the results reported in Table 1 do not match the original results published by Kennedy and Boguraev (1996), Baldwin (1997) and Mitkov (1998b) where the algorithms were tested on different data, employed different pre-processing tools, resorted to different degrees of manual intervention and thus provided no common ground for any reliable comparison. By contrast, the evaluation workbench enables a uniform and balanced comparison of the algorithms in that (i) the evaluation is done on the same data and (ii) each algorithm employs the same pre-processing tools and performs the resolution in fully automatic fashion. Our experiments also confirm the finding of Orasan, Evans and Mitkov (2000) that fully automatic resolution is more difficult than previously thought with the performance of all the three algorithms essentially lower than originally reported.

4 Conclusion

We believe that the evaluation workbench for anaphora resolution proposed in this paper

[6] Note that, since the three approaches are robust, recall is equal to precision.

[7] In Tables 1 and 2, only pronouns that are treated as anaphoric and hence tried to be resolved by the three methods are included. Therefore, pronouns in first and second person singular and plural and demonstratives do not appear as part of the number of pronouns.

alleviates a long-standing weakness in the area of anaphora resolution: the inability to fairly and consistently compare anaphora resolution algorithms due not only to the difference of evaluation data used, but also to the diversity of pre-processing tools employed by each system. In addition to providing a common ground for comparison, our evaluation environment ensures that there is fairness in terms of comparing approaches that operate at the same level of automation: formerly it has not been possible to establish a correct comparative picture due to the fact that while some approaches have been tested in a fully automatic mode, others have benefited from post-edited input or from a pre- (or manually) tagged corpus. Finally, the evaluation workbench is very helpful in analysing the data used for evaluation by providing insightful statistics.

References

Chinatsu Aone and Scot W. Bennett. 1995. Evaluating automated and manual acquisition of anaphora resolution rules. In *Proceedings of the 33th Annual Meeting of the Association for Computational Linguistics (ACL '95)*, pages 122–129.

Breck Baldwin. 1997. Cogniac: High precision coreference with limited knowledge and linguistic resources. In R. Mitkov and B. Boguraev, editors, *Operational factors in practical, robust anaphora resolution for unrestricted texts*, pages 38 – 45.

Ido Dagan and Alon Itai. 1990. Automatic processing of large corpora for the resolution of anaphora references. In *Proceedings of the 13th International Conference on Computational Linguistics (COLING'90)*, volume III, pages 1–3.

Ido Dagan and Alon Itai. 1991. A statistical filter for resolving pronoun references. In Y.A. Feldman and A. Bruckstein, editors, *Artificial Intelligence and Computer Vision*, pages 125 – 135. Elsevier Science Publishers B.V.

Antonio Ferrandez, Manolo Palomar, and L. Moreno. 1997. Slot unification grammar and anaphora resolution. In *Proceedings of the International Conference on Recent Advances in Natural Language Proceeding (RANLP'97)*, pages 294–299.

Robert Gaizauskas and Kevin Humphreys. 2000. Quantitative evaluation of coreference algorithms in an information extraction system. In Simon Botley and Antony Mark McEnery, editors, *Corpus-based and Computational Approaches to Discourse Anaphora*, Studies in Corpus Linguistics, chapter 8, pages 145 – 169. John Benjamins Publishing Company.

Niyu Ge, J. Hale, and E. Charniak. 1998. A statistical approach to anaphora resolution. In *Proceedings of the Sixth Workshop on Very Large Corpora, COLING-ACL '98*, pages 161 – 170, Montreal, Canada.

Jerry Hobbs. 1976. Pronoun resolution. Research report 76-1, City College, City University of New York.

Jerry Hobbs. 1978. Pronoun resolution. *Lingua*, 44:339–352.

Christopher Kennedy and Branimir Boguraev. 1996. Anaphora for everyone: pronominal anaphora resolution without a parser. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96)*, pages 113–118, Copenhagen, Denmark.

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

Ruslan Mitkov, R. Evans, C. Orasan, C. Barbu, L. Jones, and V. Sotirova. 2000. Coreference and anaphora: developing annotating tools, annotated resources and annotation strategies. In *Proceedings of the Discourse, Anaphora and Reference Resolution Conference (DAARC2000)*, pages 49–58, Lancaster, UK.

Ruslan Mitkov. 1998a. Evaluating anaphora resolution approaches. In *Proceedings of the Discourse Anaphora and Anaphora Resolution Colloquium (DAARC'2)*, pages 164 – 172, Lancaster, UK.

Ruslan Mitkov. 1998b. Robust pronoun resolution with limited knowledge. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING'98/ACL'98*, pages 867 – 875. Morgan Kaufmann.

Ruslan Mitkov. 2000a. Towards a more consistent and comprehensive evaluation of anaphora resolution algorithms and systems. In *Proceedings of the Discourse, Anaphora and Reference Resolution Conference (DAARC2000)*, pages 96 – 107, Lancaster, UK.

Ruslan Mitkov. 2000b. Towards more comprehensive evaluation in anaphora resolution. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, volume III, pages 1309 – 1314, Athens, Greece.

Ruslan Mitkov. 2001. Outstanding issues in anaphora resolution. In Al. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 110–125. Springer.

Constantin Orăsan, Richard Evans, and Ruslan Mitkov. 2000. Enhancing preference-based anaphora resolution with genetic algorithms. In *Proceedings of Natural Language Processing - NLP2000*, pages 185 – 195. Springer.

P. Tapanainen and T. Järvinen. 1997. A non-projective dependency parser. In *Proceedings of the 5th Conference of Applied Natural Language Processing*, pages 64 – 71, Washington D.C., USA.

Joel R. Tetreault. 1999. Analysis of syntax-based pronoun resolution methods. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99)*, pages 602 – 605, Maryland, USA.