Evaluation environment for anaphora resolution

Catalina Barbu, Ruslan Mitkov
School of Humanities, Languages and Social Studies
University of Wolverhampton
Stafford Street
Wolverhampton WV1 1SB
United Kingdom

E-mail: {C.Barbu, R.Mitkov}@wlv.ac.uk

Abstract

In this paper we argue that the comparative evaluation of anaphora resolution systems has to be performed using the same pre-processing tools and on the same set of data. We propose an evaluation workbench featuring three knowledge-poor anaphora resolution algorithms developed over a common interface. We evaluate the three methods on a corpus of technical texts and we present the results obtained for several evaluation measures.

1 Introduction

The evaluation of any NLP algorithm or system should be indicative not only of the efficiency or performance of a specific algorithm or system, but should also help us discover what a new approach brings to the current state of play of 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; 2000a; 2000b) that the evaluation of anaphora resolution algorithms and systems hardly provides 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 will be carried over to the systems' outputs, may vary. As a way forward we proposed the idea of developing an *evaluation workbench in anaphora resolution* (Mitkov 2000a) which allows the comparison of approaches sharing common pre-processing tools on the same data. This paper describes the implementation of this new evaluation environment, which incorporates Kennedy and Boguraev's (1996) parser-free algorithm, Baldwin's (1997) Cogniac and Mitkov's (1998) knowledge-poor approach for comparative evaluation.

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 are developing an *evaluation workbench for anaphora resolution* which allows the comparison anaphora resolution approaches sharing common principles (e.g. POS tagger, NP extractor, parser). 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 produce a better picture as to the advantages and disadvantages of the different approaches. Developing our own evaluation environment (and even re-implementing some of the key algorithms) also alleviates the formidable difficulties associated with obtaining the codes of the original programs.

Another advantage of the evaluation workbench can be seen in the fact that all approaches incorporated operate in a fully automatic mode. 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 evaluation of many anaphora resolution approaches focus on the accuracy of resolution algorithms and do not take into consideration the possible errors which inevitably occur in the pre-processing stage. The vast majority of approaches rely on some kind of pre-editing of the text which is fed to the anaphora resolution algorithm; some of the methods have been only manually simulated. As an illustration, Hobbs' naïve approach (1976, 1978) was not implemented in its original version. In (Dagan 1990, 1991), (Aone and Bennett 1995) and (Kennedy and Boguraev 1996) pleonastic pronouns are removed manually[1], whereas in (Mitkov 1998) and (Ferrandez et al. 1998) the outputs of the PoS 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[2] that are fully automatic is MARS, the latest version of Mitkov's knowledge-poor approach implemented by R. Evans. Recent work on this project has demonstrated that fully automatic anaphora resolution is more difficult than previous work has suggested (Orasan, Evans and Mitkov 2000).

The current version of the evaluation workbench employs one of the best available super-taggers in English - Conexor's FDG Parser. This super-tagger provides information on the dependency relations between words which allows the extraction of complex NPs. It also gives morphological information and the syntactic roles of words.

2.1 Pre-processing tools

2.1.1 Parser

The parser used for the evaluation workbench was the FDG parser developed at Conexor (Tapanainen and Jarvinen 1997). It performs a surface syntactic parsing of the text using dependency links that show the head-modifier relations between words.

The example below shows the output of the FDG parser run over the sentence: "This is an input file."

```
0
1  This   this     subj:>2     @SUBJ PRON DEM SG
2  is     be       Main:>0     @+FMAINV V PRES SG3
3  an     an       det:>5      @DN> DET SG
4  input  input    attr:>5     @A> N NOM SG
5  File   file     comp:>2     @PCOMPL-S N NOM SG
   .      .
```

2.1.2 Noun phrase extractor

Although FDG does not provide the identification of the noun phrases in the text, the dependencies established between words have served to building a noun phrase extractor. In the example above, the dependency relations help identifying the sequence "an input file". Every noun phrase lists features as identified by FDG

(number, part of speech, grammatical function), the position of the verb that they are arguments of and the number of the sentence where they occur. The result of the NP extractor is an SGML annotated file. We agreed upon this format for several reasons: it is easily readable, it allows a unified treatment of the files used for training and of those used for evaluation (which are already annotated in SGML format) and it is also useful if the file submitted for analysis to FDG already contains an SGML annotation; in the latter case, keeping the FDG format together with the existent SGML 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 SGML format.

An example of the overall output of the pre-processing tools is given below; the sentence analysed is: "Protect the Portable StyleWriter from dampness or weather, such as rain and snow":

```
<P><S><W C="V" ROLE="+FMAINV" LEMMA="protect">Protect</W><NP NUM="SG"
ROLE="1OBJ" HEAD="StyleWriter" HEADPOS="4"><W C="DET" ROLE="DN+"
LEMMA="the">the</W><W C="A" ROLE="A+" LEMMA="portable">Portable</W><W C="N"
NUM="SG" ROLE="OBJ" LEMMA="stylewriter"> StyleWriter</W></NP><W C="PREP"
ROLE="ADVL" LEMMA="from">from</W><NP NUM="SG" ROLE="1-P" HEAD="dampness"
HEADPOS="6"><W C="N" NUM="SG" ROLE="-P" LEMMA="dampness">dampness</W><W
C="CC" ROLE="CC" LEMMA="or">or</W><NP NUM="SG" ROLE="1-P" HEAD="weather"
HEADPOS="9"><W C="A" ROLE="A+" LEMMA="wet">wet</W><W C="N" NUM="SG" ROLE="-
P" LEMMA="weather"> weather</W><W C=",">,</W><NP NUM="SG" ROLE="1APP"
HEAD="such" HEADPOS="11"><W C="PRON" NUM="SG" ROLE="APP" LEMMA="such">such
</W> </NP></NP></NP><W C="PREP" ROLE="ADVL" LEMMA="as">as</W> <NP NUM="SG"
ROLE="0OBJ" HEAD="rain" HEADPOS="13"><W C="N" NUM="SG" ROLE="OBJ"
LEMMA="rain">rain</W></NP><W C="CC" ROLE="CC" LEMMA="and">and</W><W C="V"
ROLE="+FMAINV" LEMMA="snow">snow</W><W C=".">.</W></S></P>
```

2.1.3 Pleonastic it identifier

Both Kennedy and Boguraev's algorithm and MARS rely on the identification of the expletive it occurrences in the analysed texts. Although Cogniac does not mention any attempt of separating the processing of the pleonastic it from that of the anaphoric it, we have also included a pleonastic it identifier as a pre-processing tool in our implementation, since this can only improve the accuracy of the system.

The system used for identifying those instances of the pronoun *it* that were not anaphorically linked to NPs in the text was the one developed by Richard Evans (Evans 2000). It is a system based on a machine learning algorithm that was trained on manually tagged texts from Susanne and BNC corpora. Each instance of the pronoun *it* was described in terms of vectors of 35 features relevant to the classification of the pronoun as pleonastic, non-nominal or NP anaphoric. On completion, the instance base contained approximately 3100 instances of it, 1025 of which were non-nominal. New instances of *it* were then assigned vectors for comparison with vectors in the training file. TIMBL (Daelemans et. A, 1999) was then used to make an automatic classification of the new vectors. The accuracy of the classification was at the level of 78.74%.

2.2 Shared resources

The three algorithms implemented receive a list of discourse referents as input. This list is generated by running an XML parser over the file resulted from the NP Extractor and selecting only the anaphoric expressions (instances of pleonastic *it* are removed). 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

The list of discourse referents is implemented as a binary tree for optimum access.

Each algorithm enriches this set of data with information relevant to its particular needs. Kennedy and Boguraev (1996), for example, also needs the information if a certain discourse referent is embedded or not, plus a pointer to the COREF class associated to the referent.

Apart from the pre-processing tools, the three systems also share a common philosophy, 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.

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 type of knowledge. Although for the current experiments we have only included three knowledge-poor anaphora resolvers, it has to be mentioned 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 Comparative evaluation of knowledge-poor anaphora resolution approaches

The first phase of our project includes comparison of knowledge-poorer approaches which share a common pre-processing philosophy. We have selected for comparative evaluation 3 approaches that have been 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 1998). All three of these algorithms share a similar pre-processing methodology: they do not rely on a parser to process the input and use instead POS taggers and NP extractors; none of the methods make use of semantic or real-world knowledge. We re-implemented Kennedy and Boguraev's and Baldwin's algorithms and made use of the version of Mitkov's algorithm implemented by Richard Evans, referred to as MARS (Orasan, Evans and Mitkov 2000). 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). As previously mentioned, both Kennedy and Boguraev's and Baldwin's approaches benefit from

Richard Evans' program for identifying and filtering instances of non-nominal anaphora (which includes occurrences of pleonastic pronouns).

3.1 Brief outline of the three approaches

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

3.1.1 Kennedy and Boguraev

Kennedy and Boguraev describe an algorithm for anaphora resolution based on Lappin and Leass approach but without employing deep syntactic parsing. Their method is able to deal with personal pronouns, reflexives and possessives.

The general idea of the method 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. It is expected for this method to perform better than Baldwin's and Mitkov's approach since it exploits more syntactic information for determining disjoint reference.

3.1.2 Baldwin's Cogniac

Cogniac is a knowledge-poor approach to anaphora resolution that is based on a set of high confidence rules which are successively applied over the pronoun under processing. The rules are ordered according to their importance and relevance to anaphora resolution. The processing of a pronoun stops when one rule was 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 failed.

3.1.3 Mitkov's approach

Mitkov's approach is a robust anaphora resolution method for technical texts and it is based on a set of boosting and impeding indicators that are applied on each antecedent of a pronoun.

A score is calculated based on these indicators and the discourse referent with the highest aggregate value is selected as antecedent.

3.2 Brief outline of the data used for evaluation

We have used for evaluation a corpus of technical texts that was manually annotated for coreference. The corpus contains more than 50 000 words, with 19 305 noun phrases and 484 anaphoric pronouns. The files that were used are: "Beowulf HOW TO" (referred in Table 1 as *Beo*), "Linux CD-Rom HOW TO" (*CDR*), "Macintosh Help file" (*Mac*), "Portable StyleWriter Help File" (*PSW*), "Windows Help file" (*Win).*

3.3 Evaluation measures used

The workbench incorporates an automatic scoring system that operates based on an SGML 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 we intend to build support for the MATE annotation scheme as well.

We have implemented four measures for evaluation: precision and recall as defined by Aone and Bennett[3] (1995) plus success rate and critical success rate as defined in (Mitkov, 2000a). These four measures are computed 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

This 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 that are close to the overall success rates. In fact, in most cases it is really the critical success rate that matters: high critical success rate naturally implies high overall success rate.

The results are visually displayed on the screen and they can also be saved on file. For easier visual comparison, each anaphor is displayed in parallel with the antecedents found by the three anaphora resolvers.

3.4 Statistics

Besides the evaluation system, the workbench also incorporates a basic statistical calculator of the anaphoric occurrences in the input file. The parameters calculated are: the total number of anaphors, the number of anaphors in each morphological category (personal pronoun, noun, reflexive, possessive) and the number of inter- and intrasentential anaphors.

3.5 Evaluation results

In the table below the values obtained for the success rate of the three anaphora resolvers on a set of 5 files are described. The overall success rate calculated for the 426 anaphoric pronouns found in the texts was 62.5% for MARS, 59.02% for Cogniac and 63.64% for Kennedy and Boguraev's method.

File	Number of pronouns	Success Rate		
		Mars	Cogniac	Kennedy&Boguraev
PSW	77	79.74	72.1	79.8
MAC	148	66.06	60.8	67.1
WIN	51	56.86	55.9	58.7
BEO	67	45.16	45.0	46.3
CDR	83	64.83	61.3	66.3
Total	426	62.53	59.02	63.64

Table 1: Evaluation results

The results described above are only preliminary, as the development of the workbench is still in progress. We will provide additional results for the other evaluation measures and we will also perform a more extensive evaluation (in terms of number of inter- and intra-sentential anaphors, average number of candidates per anaphor, evaluation with and without identification of the pleonastic *it* instances).

4 Conclusion

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

Notes

[1] In addition, Dagan and Itai (1991) undertook additional pre-editing such as removing 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.

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

[3] This definition is slightly different from the one used in (Baldwin 1997) and (Gaizauskas and Humphreys 1996). For more discussion on that see (Mitkov 2000a).

References

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

Baldwin, Breck (1997), "CogNIAC: high precision coreference with limited knowledge and linguistic resources". Proceedings of the ACL'97/EACL'97 workshop on Operational factors in practical, robust anaphora resolution, 38-45, Madrid, Spain.

Daelemans, Walter, Zavarel Jakub, van der Slot Ko and van den Bosch, Antal (1999). "Timbl:Tilburg Memory Based Learner, version 2.0, reference guide, ilk technical report 99-01. ILK 99-01, Tilburg University

Dagan, Ido and Alon Itai (1990), "Automatic processing of large corpora for the resolution of anaphora references", Proceedings of the 13th International Conference on Computational Linguistics (COLING'90), Vol. III, 1-3, Helsinki, Finland.

Dagan, Ido and Alon Itai (1991), "A statistical filter for resolving pronoun references". In Y.A. Feldman and A. Bruckstein (Eds) Artificial Intelligence and Computer Vision, 125-135. Elsevier Science Publishers B.V. (North-Holland).

Evans, Richard (2000), "A Comparison of Rule-Based and Machine Learning Methods for Identifying Non-nominal It". In Natural Language Processing-NLP2000, Second International Conference Proceedings, Lecture Notes in Artificial Intelligence, Springer Verlag, pp.233-242

Ferrandez, Antonio, Manolo Palomar and Moreno L (1997), "Slot unification grammar and anaphora resolution". Proceedings of the International Conference on Recent Advances in Natural Language Proceeding (RANLP'97), 294-299. Tzigov Chark, Bulgaria.

Gaizauskas, Robert and Kevin Humphreys (1996), Quantitative evaluation of coreference algorithms in an information extraction system. Paper presented at the Discourse Anaphora and Anaphor Resolution Colloquium (DAARC), Lancaster, UK. To appear in S. Botley and T. McEnery (Eds) Discourse Anaphora and Anaphor Resolution, John Benjamins, 2000.

Ge, Niyu, John Hale and Eugene Charniak (1998), "A statistical approach to anaphora resolution". Proceedings of the Workshop on Very Large Corpora, 161-170. Montreal, Canada.

Hobbs, Jerry R. (1976), "Pronoun resolution", Research Report 76-1. New York: Department of Computer Science,City University of New York.

Hobbs, Jerry R. (1978), "Resolving pronoun references". Lingua, 44, 339-352.

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

Lappin, Shalom and Herbert Leass (1994), "An algorithm for pronominal anaphora resolution". Computational Linguistics, 20(4), 535-561.

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

Mitkov, Ruslan (1998b), "Robust pronoun resolution with limited knowledge". Proceedings of the 18.th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference, 869-875. Montreal, Canada.

Mitkov, Ruslan (2000a), "Towards more consistent and comprehensive evaluation in anaphora resolution". Proceedings of LREC'2000, Athens, Greece, 1309-1314

Mitkov Ruslan (2000b), "Towards more consistent and comprehensive evaluation of robust anaphora resolution algorithms and systems" (forthcoming).

Orasan Constantin, Evans Richard and Mitkov Ruslan (2000), "Enhancing Preference-Based Anaphora Resolution with Genetic Algorithms", Proceedings of NLP'2000, Patras, Greece. 185-195

Tapanainen, P. and Jarvinen, T. (1997), "A non-projective Dependency Parser". In the Proceedings of the 5th Conference of Applied Natural Language Processing, 64-71, ACL, US

Tetreault, Joel R. (1999), "Analysis of Syntax-Based Pronoun Resolution Methods". Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL '99), 602-605. Maryland, USA.