ARQAS - Venice, 23-24 June 2003.

# An Empirical Evaluation of Pronoun Resolution and Clausal Structure

**Joel Tetreault**
Department of Computer Science
University of Rochester
Rochester, NY, 14627, USA
tetreaul@cs.rochester.edu

**James Allen**
Department of Computer Science
University of Rochester
Rochester, NY, 14627, USA
james@cs.rochester.edu

## Abstract

This paper presents an automated empirical evaluation of the relationship between clausal structure and pronominal reference. Past work has theorized that incorporating discourse structure can constrain the search space in the resolution of pronouns since discourse segments, and thus potential antecedents, can be made inaccessible as the discourse progresses and the focus changes. However, very little empirical work has been done to evaluate these claims. In this study, we develop an automated system and use a corpus annotated for RST and coreference to test whether basic formulations of these claims hold. In particular, we develop and evaluate two pronoun resolution algorithms that incorporate clausal and discourse structure. The first is based on Grosz and Sidner's theory of discourse structure and the second is based on Cristea et al.'s Veins Theory. Our results show that incorporating basic clausal structure does not improve performance.

## 1 Introduction

In this paper we present an automated corpus-based analysis using Rhetorical Structure Theory (Mann and Thomson, 1988) to aid in pronoun resolution. Most implemented pronoun resolution methods in the past have used a combination of focusing metrics, syntax, and light semantics[1], but very few have incorporated discourse information or clausal segmentation. It has been suggested that discourse structure can improve the accuracy of reference resolution by closing off unrelated segments of discourse from consideration. However, until now, it has been extremely difficult to test this theory because of the difficulty in annotating discourse structure and relations reliably and for a large enough corpus. What limited empirical work that has been done in this area has focused primarily on how structure can constrain the search space for antecedents (Poesio and Di Eugenio, 2001; Ide and Cristea, 2000) and their results show that it can be effective. In this paper, we use a different metric, simply, how many pronouns one can resolve correctly with a constrained search space.

This paper builds on preliminary research discussed in (Tetreault, 2002) in which the RST-tagged Treebank (Carlson et al., 2001) corpus of Wall Street Journal articles merged with coreference information is constructed to provide a testing ground for the claims above. In addition, an existing pronoun resolution system (Byron and Tetreault, 1999) is augmented with modules for incorporating the information from the corpus: discourse structure and relations between clauses. With this testbed system, we evaluate two algorithms based on leading theories of decomposing discourse: Grosz and Sidner (1986) and Veins Theory (Cristea et al., 1998). Our results show that basic methods of decomposing discourse do not improve performance of pronoun resolution

[1] See Mitkov (2000) for a leading method.

methods.

In the following section we discuss theories that relate discourse and anaphora. Next we discuss two evaluations: the first determines a baseline algorithm to be compared against and the second tests the two new algorithms using RST. Finally, we close with results and discussion.

## 2 Background

### 2.1 Discourse Structure

We follow Grosz and Sidner's (1986) work in discourse structure in implementing some of our clausal-based algorithms. They claim that discourse structure is composed of three interrelated units: a linguistic structure, an intentional structure, and an attentional structure. The linguistic structure consists of the structure of the discourse segments and an embedding relationship that holds between them.

The intentional component determines the structure of the discourse. When people communicate, they have certain intentions in mind and thus each utterance has a certain purpose to convey an intention or support an intention. Grosz and Sidner (henceforth G&S) call these purposes "Discourse Segment Purposes" or DSP's. Given the nesting of DSP's, the intentional structure forms a tree, with the root of the tree being the main intention of the discourse. The intentional structure is more difficult to compute since it requires recognizing the discourse purpose and the relation between intentions.

The final structure is the attentional state, which is responsible for tracking the participant's mental model of what entities are salient or not in the discourse. It is modeled by a stack of focus spaces which is modified by changes in the intentional state. The set of focus spaces available at any time is the focusing structure. Focus spaces are removed (popped) and added (pushed) from the stack depending on their respective discourse segment purpose and whether or not their segment is opened or closed. The key points about attentional state are that it maintains a list of the salient entities, prevents illegal access to blocked entities, is dynamic, and is dependent on the intentional state.

To our knowledge, there has been no large-scale annotation of corpora for intentional structure. In our study, we use RST (Mann and Thompson, 1988) to approximate the intentional structure in Grosz and Sidner's model. With some sort of segmentation and a notion of clauses one can test pushing and popping, using the depth of the clause in relation to the surrounding clauses.

Using RST to model G&S discourse structure is not without precedent. Moser and Moore (1996) first claimed that the two were quite similar in that both had hierarchal tree structures and that while RST had explicit nucleus and satellite labels for relation pairs, DSP's also had implicit salience labels, calling the primary sentence in a DSP a "core," and subordinating constituents "contributors." However, Poesio and DiEugenio (2001) point out that an exact mapping is not an easy task as RST relations are a collection of intentional but also informational relations and it is not clear how to handle subordinating DSP's of differing relations to model pushes and pops of the attentional stack.

### 2.2 Veins Theory

Veins Theory (Cristea et al., 1998; Ide et al., 2000) is an extension of Centering Theory from local to global discourse. The empirically tested method makes use of discourse structure (RST trees) to determine the accessibility of referents. The theory assumes that only a subset of the clauses preceding the anaphor are actually relevant to successfully interpreting the anaphor. This subset (domain of referential accessibility, or DRA) is determined by the interaction of the tree hierarchy and whether a clause is a nucleus or a satellite. As a result of this pruning effect, the theory has the advantage over knowledge-poor approaches to pronoun resolution since it constrains the search space for a pronoun.

Using RST as the basis for their discourse representation, terminal nodes in the binary tree represent the clauses of the discourse, and non-terminal nodes represent the rhetorical relations. The DRA for a clause is computed in two steps. First, the "head" of each node is computed bottom-up by assigning a number to each terminal node. Non-terminal nodes are labeled by taking the union of the heads of its nuclear children. The second step, computing the "vein," is a top-down method. First, the vein of the root of the tree is the head. For every nuclear node, if it has a left sibling that is a satellite, its vein is the union of the head of the child and its parent vein, otherwise it inherits its parent's vein only. For every satellite node, if the node is the left child of its parent then its vein is the union of its head with the parent's vein. Otherwise, its vein is the union of its head with the parent's vein but with all prior left satellites removed. Finally, the DRA for a clause is simply all the nodes in the clause's vein that precede it. Intuitively, if a node has parents that are all nuclei, it will be more accessible to other entities since it is highly salient according to Veins Theory (VT). However, satellites serve to restrain accessibility.

## 3 Baseline Selection

Determining the usefulness of incorporating discourse information in reference resolution requires a large corpus annotated with coreference and clausal information, and a system to try different algorithms. In the following sections we discuss our corpus, our testbed system for extracting noun-phrase entities, and finally the algorithms and their results. After testing each algorithm on the same corpus, the best one would be selected as the baseline algorithm. If discourse or clausal information is used correctly we should see an improvement over the baseline algorithm.

### 3.1 Corpus Description

The test corpus was constructed by merging two different annotations of a subset of the Penn Treebank (Marcus et al., 1993). The news articles cover such varied topics as reviews of TV shows and the Japanese economy. The portion of the Treebank consists of 52 Wall Street Journal articles which includes 1241 sentences and 454 non-quoted third person pronouns that refer to noun-phrase entities. 10 of the pronouns have long-distance antecedents, where the antecedent is found two or more sentences away from the pronoun.

Carlson et al. (2001) annotated those articles with rhetorical structure information in the manner of Mann and Thompson (1988) with very high annotator reliability. This annotation breaks up each discourse into clauses connected by rhetorical relations. So from this work there is a decomposition of sentences into a smaller units (a total of 2755 clauses) as well as a discourse hierarchy for each article and relations between pairs of segments. The corresponding Penn Treebank syntactic structures for each sentence were also annotated with coreference information in the same manner as Ge et al. (1998). This meant that all third-person pronouns were marked with a specific identification number and all instances of the pronoun's antecedent were also marked with the same id. In addition, the Penn Treebank includes annotations for the syntactic tree structures of each sentence so syntactic attributes such as part-of-speech and number information were extracted. Also, each noun phrase entity was marked manually for gender information.

Finally, the RST corpus and the Penn Treebank coreference corpus were merged such that each discourse entity (in this case, only noun-phrases) had information about its syntactic status, gender, number, coreference, etc. and the following discourse information: the clause it is in, the depth of the clause in the RST tree, and the rhetorical relations that dominate the clause. The Penn Treebank data and only the clausal breakdown of each sentence are used in this evaluation. In the second evaluation, all of the RST data comes into play.

### 3.2 Algorithms

One of the problems with reporting the performance of a pronoun resolution algorithm is that researchers often test on different corpora so it is hard to compare results. For example, an algorithm tested on a news corpus may perform differently on a corpus of short stories. In this particular experiment, we have a common corpus to test different algorithms, with the goal of simply selecting the best one to use as a baseline for comparison with schemes that incorporate clausal information. We examine three well-known pronoun resolution methods: Left-Right Centering (Tetreault, 1999), the S-list algorithm (Strube, 1998), and Brennan et al.'s centering algorithm (1987), in addition to a naive metric. The naive metric involves searching through a history list starting with the last mentioned item and selecting the first one that meets gender, number, and syntactic constraints. All four algorithms are primarily syntax-based. Because of this limitation they should not be expected to fare too well in interpreting pronouns correctly since proper interpretation requires not only syntactic information but also semantics and discourse information.

Each algorithm was tested on the corpus in two different versions (see Figure 1): the first is the conventional manner of treating sentences as the smallest discourse unit (S); the second involves splitting each sentence into clauses specified by the RST annotations (C).

The (S) results agree with the larger study of the same algorithms in Tetreault (2001) - that the LRC performs better than the other two algorithms and that on a new corpus, one would expect the algorithm to resolve 80% of the pronouns correctly.

The (C) results are a first stab at the problem of how to incorporate clausal structure into pronoun resolution algorithms. The result is negative since each algorithm has a performance drop of at least 3%. The main result for our purposes is that LRC performs the best and thus is selected a the baseline algorithm.

## 4 Algorithms

In this section, we describe two pronoun resolution algorithms that use clausal structure to constrain the search for antecedents. We also describe a series of corpus transformations that each algorithm is tested on.

### 4.1 Grosz and Sidner Stack Approximation Algorithm

Based on Grosz and Sidner's pushing and popping discourse structure, we work under the simple assumption that an entity is inaccessible if it is more embedded in the RST tree than the referring entity, meaning if we were explicitly tracking the attentional state, that embedded utterance would have been popped before our current utterance was processed.

Thus the Grosz and Sidner approximation (henceforth G&S) works only by considering the depth of past clauses. The algorithm is as follows: for each pronoun the attentional stack is constructed on the fly since we have perfect information on the structure of the discourse. The search works by looking through past clauses that are either at the same depth or closer to the root than the previous clause visited. The reasoning is that embedded segments that are farther from the root are not related to the entities that follow them. If they were, they would share the same embedding. Clauses at the same depth can be viewed as being in the same discourse segment and clauses that are closer to the root can be viewed as dominating the current clause. In addition, the previous clause is always searched even if it is a lower depth. This follows Walker's (2000) analysis which found that reference can occur between two utterances even if they are split by a segment boundary.

Figure 2 shows how this works. Assume that clauses closest to the left are the closest to the root of the tree (lower depth). When searching for an antecedent for a pronoun in C7, first search all preceding entities in C7, if one is not found, then go back clause by clause until one is found. So the search order would be C6 (since the previous utterance is automatically search regardless of depth) then skip over C5 since it is more embedded than the current clause C6. C4, however, is accessible since it is the same depth as C4.

### 4.2 Veins Algorithm

The original formulation off Veins Theory is a metric of accessibility not resolution since it does not specify how to search the DRA or how to search clauses within the DRA. The algorithm presented here uses the constraints of VT within the framework of LRC. The algorithm is as follows: for every pronoun, search the clauses of its DRA from most recent (the current clause) to least recent, from left-to-right. If an antecedent is not found within the DRA, the LRC algorithm is used to find a suitable antecedent by searching all past clauses. This approximation accords with the VT claim that referents outside the DRA incur a higher processing load on the interpreter. This "backup mechanism" results in a 14% boost in performance.

In terms of long-distance pronominalization, the original Veins formulation was unable to resolve 6 of the 10 cases when treating sentences as the minimal discourse unit, and when considering clauses, was unable to resolve 9 of the 10 cases. All of these were pronouns and antecedents in attribution relations.

### 4.3 Corpus Transformations

Tetreault (2002) showed that using the RST tree in the Grosz and Sidner approach produced very poor results (in the 50% range). We believe that the RST decomposition produces too fine a segmentation and

| Algorithm | Right (S) | % Right (S) | Right (C) | % Right (C) |
|---|---|---|---|---|
| LRC | 367 | 80.84 | 347 | 76.43 |
| S-list | 333 | 73.35 | 318 | 70.04 |
| BFP | 270 | 59.47 | 221 | 48.68 |
| Naive | 230 | 50.66 | 254 | 55.95 |

Figure 1: Pronoun Resolution Algorithms over (S)entences and (C)lauses

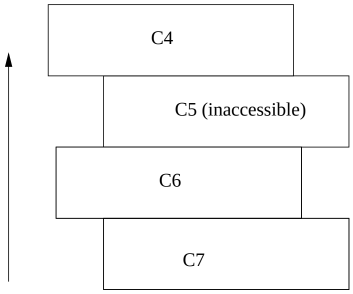


Figure 2: Accessibility due to Clause Embedding

thus many clauses are deemed unfairly inaccessible. To counter this, we developed two transformations to a RST tree: the first involves replacing multi-clausal sentences with one clause in the RST tree; and the second involves merging all subtrees that have a satellite leaf in a relation with a subtree consisting of all leaves, one of which is a nucleus (see Figure 3 (1) for an example).

The intuition with the first transform (SENT) is that many of the errors in the original approximation of G&S based on RST are intrasentential. By merging the clauses together, the tree becomes flattened, and all entities within a sentence are accessible. An example of this transform is in Figure 3 in which one assumes the clauses C1, C2 and C3 of the RST subtree in (1) are constituents of one sentence. Doing the SENT transform yields the result in (3), a subtree that is now a leaf of the sentence reconstructed.

The intuition with the second transform (SAT) is that satellite leaves that modify a nucleus subtree are very closely related to the content of the nucleus leaf of that subtree, as well as the satellite leaf of that subtree. By merging them, the tree is flattened, and pronouns within the original satellite leaf can refer to clauses in the subtree since they are now at the same depth. (2) in Figure 3 provides an illustration of the satellite transformation on (1). The side-effect of this transformation is that the RST tree is no longer binary. Finally, a third transform (SENT-SAT) involves using both of the transforms on the corpus to flatten the tree even more.

In addition, Ide and Cristea note that all exceptions to accessibility in their corpus analysis come from pronouns and antecedents in attribution relations (such as “he said....”). Our corpus exhibits a similar trend: 105 pronouns don’t have an antecedent found in the DRA, out of these 105 inaccessibility cases, 73 are in attribution relations. Though there are 32 unaccounted for (usually because there was an intervening satellite node that prevented reference) the attribution relation tends to be a big block in accessibility still. Another transformation, ATT, is used to counter this by simply merging leaves that stand in attribution relations. So if a subtree has two leaves in an attribution relation, it is replaced by a leaf with the text of the two original leaves merged. This process is similar to SENT.

## 5 Results

Both algorithms were run over the original RST corpus, ATT (attribution merge) and SAT (satellite merge) transformation of our original corpus (see Figure 4). The (S) version means that the LRC intrasentential search was used over the entire sen-

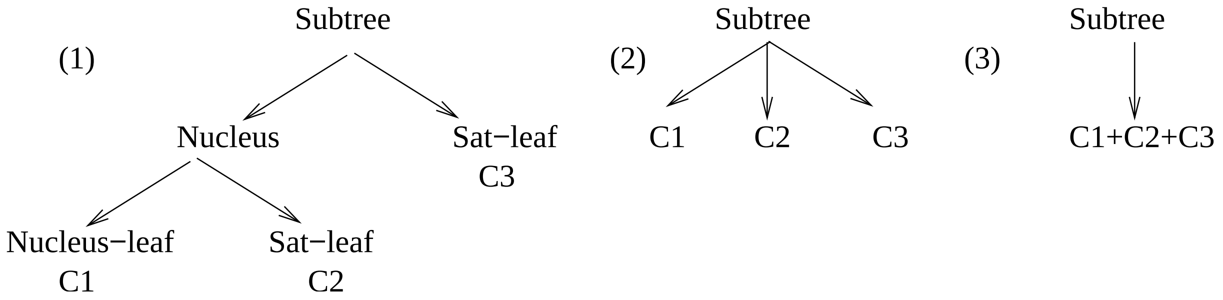


Figure 3: Satellite Transform (2) and Sentence Transform (3)

tence, not just the clause that the pronoun occupies (C). This means that the current sentence is always searched, and if a referent is not found, previous clauses are searched. The (*) signals that the algorithm does not search the previous clause as a default.

Because the SENT transformations created unbalanced RST trees, the Veins algorithm could not be tested with that transform. The results in Figure 5 show how the Grosz and Sidner algorithm fares over the SENT and SENT-SAT transforms with and without using the last-seen metric.

Without the attribution transform, the Veins Algorithm (S) gets only 6 of the 10 pronouns resolved correctly. The G&S algorithms do about as well without segmentation. With the transformations, all the algorithms resolve all 10 cases correctly. However, it should be noted that the original LRC algorithm also resolves all correctly. This success rate is due to the fact that 9 of the 10 pronouns are either “he” or “him” and there are no other candidates with masculine gender in the discourse up to that point. So a simple search through a history-list would resolve these correctly. The other long-distance pronoun is a plural (“their”) and again there are no competing antecedents.

## 6 Discussion

Discourse decomposition can be evaluated in two ways: intrasentential breakdown (clausal level) and intersentential breakdown (discourse level). In the intrasentential case, all the algorithms performed better when using the (S) method, that is, when the intrasentential search called for searching the sentence the pronoun is in, as opposed to just the clause the pronoun is in. This indicates that ordering clauses by their depth within the sentence or by the Veins information does not improve intrasentential performance, and thus one is better off searching based on grammatical function than incorporating clausal information.

One can evaluate the intersentential decomposition by testing whether the pronouns with long-distance antecedents are resolved correctly. Determining global discourse structure involves finding the middle ground between strict segmentation (using the exact RST tree) and under-segmenting. Too strict a segmentation means that antecedents can be deemed incorrectly inaccessible; very little segmentation means that too many competing antecedents become available since referents are not deemed inaccessible. In our corpus, evaluating intersentential decomposition is difficult because all of the long-distance pronouns have no competing antecedents, so no discourse structure is required to rule out competitors. Therefore it is hard to draw concrete conclusions from the fact G&S on the SENT and SENT-SAT transforms performs the same as LRC algorithm. However, it is promising that this metric does get all of them right, at least it is not overly restrictive. The only way to check if the method under-segments or is a good model is by testing it on a corpus that has long-distance pronouns with competing potential referents. Currently, we are annotating a corpus of dialogs for coreference and rhetorical structure to test this method. It should also be noted that even if an intersentential decomposition method performs the same as knowledge-poor method, it has the advantage of at least decreasing the search space for each pronoun.

Finally, we developed an algorithm for Veins Theory that uses VT to constrain the initial search for

| Transform | Veins (S) | Veins (C) | G&S (S*) | G&S (S) | G&S (C) |
|---|---|---|---|---|---|
| original | 78.85 | 76.65 | 72.55 | 78.90 | 71.40 |
| ATT | 79.30 | 78.19 | 73.68 | 79.30 | 76.32 |
| SAT | 78.85 | 76.42 | 73.63 | 79.08 | 73.85 |

Figure 4: Pronoun Resolution Algorithms over ATT and SAT corpora

| Transform | G&S(*) | G&S |
|---|---|---|
| SENT | 78.51 | 80.84 |
| SENT-SAT | 79.74 | 80.84 |

Figure 5: Grosz and Sidner over SENT and SENT-SAT corpora

a referent, if one is not found, LRC is used as a default. As suggested by the VT authors, we merged clauses in attribution relations, and this improved performance slightly, but not enough to better 80.84%. VT run on the SAT transform offered no performance enhancement since the theory already makes the nucleus subtrees accessible to satellite leaves.

In conclusion, this study evaluates the theory that clausal segmentation should aid in pronoun resolution by testing two algorithms based on two leading theories of discourse segmentation. Both approaches have the promise of improving pronoun resolution by 1. making search more efficient by blocking utterances or classes from consideration, thus speeding up the search for an antecedent and 2. making search more successful by blocking competing antecedents. We use resolution accuracy for all pronouns and accuracy over long-distance pronominalizations as metrics of success. Our results indicate that incorporating discourse structure does not improve performance, and in most cases can actually hurt performance. However, due to the composition of long-distance pronouns in the corpus, it is necessary to test the G&S algorithm on the SENT transform before drawing a definitive conclusion on the theory.

## 7 Future Work

Since cases of long distance pronoun resolution are rare, most of the gains in improving accuracy will come from correctly resolving pronouns intrasententially and with the previous utterance. Our error analysis shows that in many cases, determining the coherence relations as Kehler suggests (2002) (such as detecting parallelism between sentences or within sentences) could improve interpretation. In addition, many errors stem from competing antecedents in which incorporating knowledge of the verbs and the entities discussed would prove invaluable.

Finally, our research here has assumed perfect knowledge of discourse structure. Ultimately, the goal is to be able to incrementally build discourse structure while processing a sentence. For this to occur, one has to take into account forms of referring expression, cue words, changes in tense, etc. There has been some work in this area such as Hahn and Strube (1997) who developed an algorithm for building a discourse hierarchy incrementally from changes in theme and centered entities.

## Acknowledgements

We would like to thank Mary Swift and three anonymous reviewers for insight and comments. Partial support for the research reported in this paper was provided by the DARPA research grant no. F30602-98-2-0133 to and the ONR grant no. N00014-01-1-1015, both to the University of Rochester.

## References

Susan E. Brennan, Marilyn W. Friedman, and Carl J. Pollard. 1987. A centering approach to pronouns. In *Proceedings, 25th Annual Meeting of the ACL*, pages 155–162.

Donna K. Byron and Joel R. Tetreault. 1999. A flexible architecture for reference resolution. In *Proceedings, 9th Conference of the EACL*, pages 229–232.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2001. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Proceedings of the 2nd SIGDIAL Workshop on Discourse and Dialogue, Eurospeech 2001*, Denmark, September.

Dan Cristea, Nancy Ide, and Laurent Romary. 1998. Veins theory: An approach to global cohesion and coherence. In *Proceedings of ACL/COLING*, pages 281–285, Montreal, Canada, August.

Niyu Ge, John Hale, and Eugene Charniak. 1998. A statistical approach to anaphora resolution. *Proceedings of the Sixth Workshop on Very Large Corpora*.

Barbara J. Grosz and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 1995. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226.

Udo Hahn and Michael Strube. 1997. Centered segmentation: Scaling up the centering model to global discourse structure. In Philip R. Cohen and Wolfgang Wahlster, editors, *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pages 104–111, Somerset, New Jersey.

Nancy Ide and Dan Cristea. 2000. A hierarchical account of referential accessibility. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, ACL 2000*, pages 416–424, Hong Kong, China.

Megumi Kameyama. 1998. Intrasentential centering: A case study. In M. Walker, A. Joshi, and E. Prince, editors, *Centering Theory in Discourse*, pages 89–112. Oxford University Press.

Andrew Kehler. 2002. *Coherence, Reference, and the Theory of Grammar*. CSLI Publications.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Towards a functional theory of text organization. *TEXT*, 8(3):243–281.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Lingusitics*, 19(2):313–330.

R. Mitkov. 2000. Towards a more consistent and comprehensive evaluation of anaphora resolution algorithms and systems. In *2nd Discourse Anaphora and Anaphora Resolution Colloquium*, pages 96–107.

M. Moser and J.D. Moore. 1996. Toward a synthesis of two accounts of discourse structure. *Computational Linguistics*, 22(3):409–419.

Massimo Poesio and Barbara di Eugenio. 2001. Discourse structure and accessibility. In *ESSLI 2001 Workshop on Information Structure, Discourse Structure and Discourse Semantics*.

Michael Strube. 1998. Never look back: An alternative to centering. In *COLING/ACL '98: 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, volume 2, pages 1251–1257, Montreal, Canada.

Joel R. Tetreault. 1999. Analysis of syntax-based pronoun resolution methods. In *Proceedings, 27th Annual Meeting of the Association for Compuational Linguisitcs*, pages 602–605.

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

Joel R. Tetreault. 2002. Clausal structure and pronoun resolution. In *4th Discourse Anaphora and Anaphora Resolution Colloquium*, pages 217–220.

Marilyn Walker. 2000. Toward a model of the interaction of centering with global discourse structure. *Verbum*.

# Summarizing documents based on cue-phrases and references

**Dan Cristea**
„Al.I.Cuza" University of Iasi
Faculty of Computer Science
and
Romanian Academy
Institute of Theoretical Computer Science – the Iasi branch
dcristea@infoiasi.ro

**Oana Postolache, Georgiana Puşcaşu, Laurenţiu Ghetu**
„Al.I.Cuza" University of Iasi
Faculty of Computer Science
16, Berthelot St.
6600 – Iasi, Romania
{oanap, georgie, laug}@infoiasi.ro

## Abstract

The paper presents a method of building the discourse structure of a discourse by combining indications on local structure given by cue-phrases with indications on global structure given by references found in the text. Then the discourse structure is used to obtain focused summaries.

## 1 Introduction

There is generally accepted that a strong correlation exists between the structure of discourse and referentiality (Fox, 1987; Vonk *et al.*, 1992, Cristea *et al.*, 2000). On another hand, document summarization could take advantage from knowing the structure. Marcu (2000), for instance, has shown how parameterized summaries of a document can be build provided its rhetorical structure is known. Putting all together, one can arrive, without any surprise, at the strong interdependence between referentiality and summaries, intermediated by the discourse structure. A simple argument in support of this interdependence is that a summary cannot be coherent if it contains dangling references.

In this paper we present a method to obtain coherent focused summaries based on the discourse structure of the discourse, which is partially inferred from indications on local structure given by cue-phrases and partially from references found in the text.

The summaries that we obtain are extracts (Mani, 2001). We say that a summary of a document is focused on a certain discourse entity if the summary reveals on short what the document tells about the key-entity, within the context of the whole document. A possible scenario addressing the need for a focused summary is that of a user interested in reviewing scientific texts, in particular in findings on a certain drug. Using Google or another search engine she gets a tremendous lists of documents mentioning the searched entity. Since time does not allow her to read all the found documents, abstracts would be of value. The problem with a general abstract that can be obtained by passing the task to a common abstracting engine is that the item searched could be secondary to the theme of the document, in which case it will not be included in the generated abstract. The user would be interested to know, briefly, why is that entity mentioned in a document, therefore a focused abstract.

At the base of our method stays the assumption that if summarization is the goal, a less precise discourse structure is sufficient. To obtain it, cue-words and phrases are good indicators of local structural interdependencies between elementary discourse units (*edu*s); based on cue-phrases, elementary sentence-level trees (*sdt*s) are inferred; they are then integrated into a global coherent discourse tree using indications on discourse structure brought by references, as outlined by veins theory (Cristea *et al*, 1998).

The paper is structured as follows: section 2 presents the method, sections 3 and 4 present the basics (veins theory and the resolution of anaphora), section 5 describes a set of consistency constrains for the discovery of *sdt*s, sections 6 displays the methods of integration of the *sdt*s into a whole discourse tree based on references, section 7 presents the data employed in the experiment, and some results, and section 8 discusses possible extensions.

## 2 The method

A text can be read in many ways. Practically each *edu* of the text gives a specific perspective from which to interpret the whole text. Such a perspective centred on a certain *edu* should be thought as revealing what would be the meaning of that particular *edu* in the overall context. Cristea *et al* (1998) propose a theory to evidence centred interpretations, cut up from a rhetorical-like discourse tree structure. The vein expressions defined there constitute means to look at *edu*s from inside out. Each such vein expression, claiming to express what the text says about that specific *edu* in the overall context, gives also a way of summarising the text, focussed on the entities mentioned in that *edu*.

Many discourse parsing methods have been described (Cole *et al.*, 1995; Marcu, 2000; Cristea, 2000). Cristea (2000), for instance, presents an incremental discourse parsing method, which places at the base the principle that a text has the one discourse structure, that displays the smoothest centering transitions (Grosz *et al.*, 1995) along veins, as well as most of the references satisfied along the veins. These two criteria combine to score a number of plausible partial trees and to retain at each step of the incremental development N best scored trees. Unlike Cristea (2000), the method proposed here does not first build a tree in order to accept it, if well scored, or to filter it out, if badly scored, but rather uses references as a guide during the development the tree.

The other clue used in building the structure is given by cue-phrases (Knott and Dale, 1992), (Marcu, 2000) which are used to build *sdt*s. To do that, we use Soricutu and Marcu's (2003) claim that the text span corresponding to one sentence is merely covered by one node in the structure (more than 90% of the cases, according to them).

The preparatory phases suppose POS-tagging (Tufis, 1999), syntactic tagging done by an FDG parser and NP-tagging (Ait-Mohtar, and Chanod, 1997). Then *edu* are detected (Puscasu, forthcoming) based on the identification of finite verbs and detection of their syntactic roles. Local corrections, mainly due to cue-phrases, are also possible.

Following, *sdt*s are build (as will be described in section 5). In parallel, or following the *sdt*-building phase, antecedents of anaphors are looked for, by running the AR-engine (described in section 4). The chains of co-referential links are then used to sew pieces together in a complete discourse structure (described in section 6).

Having the discourse structure, the vein expressions of the *edu* containing the entity search for will configure the output summary.

## 3 Veins theory

By using the RST notion of nuclearity, veins theory (VT) (Cristea *et al.*, 1998), (Ide&Cristea, 2000) reveals a "hidden" structure in the discourse tree, called *vein*, which enables to evidence for each unit of a discourse a *domain of evocative accessibility* (*dea*) as a string of units where antecedents of anaphors belonging to the unit should be found.

The fundamental intuition underlying the unified view on discourse structure and accessibility in VT is that the RST-specific distinction between nuclei and satellites constrains the range of referents to which anaphors can be resolved; in other words, the nucleus-satellite distinction, superimposed over a tree-like structure of discourse, induces for each anaphor a *dea*.

The observations that underline the computation of vein expressions in VT are as follows (discourse units are noted here after $u_1$, $u_2$, $u_3$, while relations $R$, $R_1$, $R_2$; when used as arguments of relations, the units' nuclearity will be marked by a superscript $^n$ – for nucleus, and $^s$ – for satellite; we will say that "a unit $u_2$ refers a unit $u_1$" and we will understand "a referential expression belonging to a unit $u_2$ refers a discourse entity also referred from the unit $u_1$"):

- a satellite or a nucleus can refer a nuclear sibling to its left: in sequences $u_1^n\ R\ u_2^s$, or $u_1^n\ R\ u_2^n$, $u_2$ can refer $u_1$;
- a nucleus can refer its own left satellite: in sequences $u_1^s\ R\ u_2^n$, $u_2$ can refer $u_1$;
- a right satellite of a nucleus $u$ cannot be accessed from another right sibling, nuclear or satellite, of $u$: in sequences $(u_1^n\ R_1\ u_2^s)^n\ R_2\ u_3^n$ or $(u_1^n\ R_1\ u_2^s)^n\ R_2\ u_3^s$, $u_3$ can refer $u_1$ but not $u_2$;
- a nucleus blocks the accessibility from a right satellite to a left satellite: in sequences $(u_1^s\ R_1\ u_2^n)^n\ R_2\ u_3^s$, $u_3$ can refer $u_2$ but not $u_1$.

VT contributes with a view on top-down summarization, similar to Marcu's (2000), while also revealing how focused summaries can be produced.

## 4 Anaphora Resolution

In (Cristea and Dima 2001), (Cristea *et al.*, 2002a) a framework incorporating a general anaphora resolution (AR) engine and able to accommodate different AR models is proposed. This approach sees the linguistic and semantic entities involved in the cognitive process of anaphora resolution represented on three layers: the **text layer** – populated with referential expressions (*re*s), the **projected layer** – where feature structures are filled-in with information fetched from the text layer (in the following, projected structures – *ps*s) and the deep **semantic layer** – where discourse entities (*de*s), actually a representation of the entities the discourse talks about, are placed. It is said that a *ps* is **projected** from an *re* and that a *de* is **proposed** or **evoked** by a *ps*.

Within the AR-engine framework an AR model is defined in terms of four components: a set of attributes and the corresponding types of the objects populating the projection and semantic layers, a **set of knowledge sources** (virtual processors) intended to fetch values from the text to the attributes of the *ps*, **a set of matching rules and heuristics** responsible to decide whether the *ps* corresponding to an *re* introduces a new *de* or, if not, which of the existing *de*s it evokes, and **a set of heuristics that configure the domain of referential accessibility**, establishing the order in which *de*s have to be checked, or certain proximity restrictions.

In (Cristea *et al.*, 2002a), pronominal as well as noun anaphora were investigated. To a great extend, the results proved the initial hypothesis, namely that models behave better and better as more features are fired. For a small corpus of about two pages taken from the novel "1984" of G. Orwell (where five characters have been tracked, whose co-reference chains in the golden annotation had lengths of 23, 14, 3, 25 and 16 referential expressions), the best models experienced proved 100% precision and a recall in the range 70% to 100%. In another research (Cristea *et al.*, 2002b) the investigation was extended over cases generally considered difficult to tackle (co-reference resolution triggered by positional constrains, common nouns anaphor and antecedent with disagreement in lemma, noun and pronoun anaphors displaying number disagreement with the antecedents, bridging anaphora, as well as anaphoric references other than net co-references).

## 5 Consistency constraints for elementary discourse trees

In this section we propose a representation and a method of determining safe inter-*edu*s local dependencies, contributed by cue words or phrases (in the following, called *markers*). The dependencies will configure an elementary discourse tree structure covering mainly a sentence (sometimes even more than that) in which inner nodes are labelled with markers and terminal nodes with *edu* labels. Each node of the tree is also marked by a nuclearity function in the set {*n*, *s*} (for nuclear, satellite) such that at each level, between the two descendents of an inner node, at least one is marked *n*.

Example 1:

[*John is determined to pass the NLP exam*$^{1}$] *so, because* [*he has missed many courses* $^{2}$] *and* [*was only vaguely implicated at the working sessions* $^{3}$] , [*he will have a hard time until summer.* $^{4}$]

In this example, the notation indicates the segmentation in *edu*s (in square brackets) and the cue words with an impact in the determination of the discourse structure (underlined): *so* – indicates that a unit subordinate to the preceding one follows; *because* (following another cue word, as well as in a sentence-initial position) – indicates that the unit it is prefixing is a subordinate of a unit that follows after this one; *and* – indicates a conjunction between two units of equal nuclearity that prefix and, respectively, succeeds it. This arrives at associating to markers argument patterns, as suggested in Figure 1:

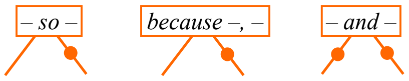


Figure 1: Arguments patterns of cue-phrases

Example 1 displays the following distribution of *edu*s and markers: 1 *so, because* 2 *and* 3, 4.

On each branch stemming out of a marker in Figure 1, virtually any domain of arguments, obtained by a combination of the units laying in the text on the corresponding part, could be formed. In the following notation, we will mark these domains as ordered lists of discourse units, while also underlying the nuclear arguments:

[1] *so* [2,3,4]
*because* [2,3,4], [2,3,4]
[1,2] *and* [3,4]

Among all possible combinations of lists of *edu*s encumbered by this scheme, we will reject from the beginning the empty lists, as well as those whose content concatenation display gaps in the sequence. The following pairs of argument-lists remain:

| *– so –* | *because –, –* | *– and –* |
|---|---|---|
| [1] – [2,3,4] | [2] – [3,4] | [1,2] – [3,4] |
| [1] – [2,3] | [2,3] – [4] | [2] – [3,4] |
| [1] – [2] | | [1,2] – [3] |
| | | [2] – [3] |

Among the 3 * 2 * 4 = 24 possibilities, many are still inconsistent. The following rules state further constrains (we will note the lists with $M_1$, $M_2$, etc.). They express natural conditions of tree well-formedness:

**The "nesting-arguments" rule**:

If $x \in M_i \cap M_j$ with $i \neq j$, then either $M_i \subseteq M_j$ or $M_i \supseteq M_j$.

This rule states that it is impossible to have two inner nodes of the tree, which cover crossing text spans on the terminal frontier.

The combination $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2], $M_4$=[3,4] (for *because*), and $M_5$=[1,2], $M_6$=[3,4] (for *and*) do not abbey this rule, because $2 \in M_2$, $2 \in M_5$ and neither $M_2 \subseteq M_5$, nor $M_2 \supseteq M_5$.

Instead, the combination $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2], $M_4$=[3,4] (for *because*), and $M_5$=[2], $M_6$=[3,4] (for *and*) do abbey the nesting arguments rule.

**The "balanced-displacement" rule**:

For any two *edus* $x$, $y$ placed in sequence ($x$ before $y$), at least a marker, denoted by $m$, exists such that: $x \in$ left_subtree($m$) and $y \in$ right_subtree($m$).

This rule forbids the existence of dangling *edus* in an elementary discourse tree. It stems from the assumption that a sufficient number of markers are found in the text. There where the text contributes with no marker, an empty cue-word ∅ is considered instead, with the default argument pattern: – ∅ –. Any of the combinations of nuclearity labels ($n$, $n$), ($n$, $s$), ($s$, $n$) are possible for its arguments.

In the example above, the combination of lists $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2], $M_4$=[3,4] (for *because*), and $M_5$=[2], $M_6$=[3,4] (for *and*) do not obey this rule, because for *edu*s 3 and 4 there is no marker with 3 in its left sub-tree and 4 in its right sub-tree.

**The "unique-root" rule**:

There is one and only one marker that covers the sequence of all *edu*s.

In the example above, the combination of lists $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2], $M_4$=[3,4] (for *because*), and $M_5$=[1,2], $M_6$=[3,4] (for *and*) do not obey this rule, because both *so* and *and* do cover the whole range of *edu*s.

**The "one-parent" rule**:

There are no two lists $M_i = M_j$ with $i \neq j$.

This rule asserts the obvious condition in trees that is impossible to have one text span which is an argument to two distinct markers.

For instance, the combination $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2], $M_4$=[3,4] (for *because*), and $M_5$=[2], $M_6$=[3,4] (for *and*) contradicts twice this rule because of the lists $M_3$=$M_5$ and $M_4$=$M_6$.

Among the 24 possible combinations of lists of the above example, only one obeys all four rules: $M_1$=[1], $M_2$=[2,3,4] (for *so*), $M_3$=[2,3], $M_4$=[4] (for *because*), and $M_5$=[2], $M_6$=[3] (for *and*), which is also the expected one, displaying the sentence-level tree of Figure 2.

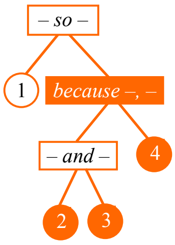


Figure 2: The elementary discourse tree of Example 1 (nuclear nodes filled-in)

The above rules applied to sentence-initial markers yields the integration of adjacent sentences into larger elementary tree.

## 6 The processing model

This section describes how pieces can be sewed together. The idea is to integrate elementary trees into a global one by taking into consideration references discovered by the AR-engine and arranging the *edu*s the references belong to such that they lay mostly along unit's vein expression.

Processing follows the following three phases:

1. determination of co-referential chains;

2. building sentence level discourse trees (*sdt*s) based on intra-sentence markers;

3. integration of *sdt*s up to a global discourse tree.

During the first phase, AR-engine is run over the POS-tagged, FDG-analysed, and NP-tagged text, as explained in section 4. The result is a set of co-referential chains of *re*s. All *re*s belonging to each such chain point to a unique *de*.

During the second phase, the syntactic constrains encumbered by cue-phrases at (mainly) sentence level are applied, in order to arrive to a sequence of *sdt*s, as explained in section 5. Then, during the third phase the sentence level trees are combined in sequence with the aim to obtain one complete tree of the whole discourse.

Let's note that the model we describe is opened for both an incremental as well as a pipe-line type of processing. In incremental processing, suppose a partial discourse tree (*pdt*) is obtained from processing the text up to (and excluding) the current *edu*. The current sentence *s* is submitted to the first phase, as described above, resulting in a set of co-reference relations from all anaphors contained in *s*. Then, as a result of running the second phase, an *sdt t*, is obtained from *s*. Finally, the third phase will integrate *t* into *pdt*, leaving a larger *pdt*. In a pipe-line type of processing, the first phase is run over the whole document, leaving a set co-referential expressions. In parallel with this phase or following it, the second phase will be run over the whole text, leaving a sequence of *sdt*s. Finally, the third phase will be run, in order to integrate the sequence of *sdt*s into a global discourse tree by taking into consideration the set of co-references.

The following discussion applies to both processing models. We will suppose the parser is in a state when the first *i sdt*s have been combined into a partial discourse tree structure, $pdt_i$, and the next *sdt* under operation is $sdt_{i+1}$. This tree has to be combined with the developing tree $pdt_i$ by adjoining an auxiliary tree obtained from this one on the right frontier of the developing tree (Cristea&Webber, 1997). An auxiliary tree of an *sdt t* consists of a relation node, with a dummy (foot) node as its left child and *t* as its right child. Figure 3 displays the adjoining operation. If the name of the relation rooting the auxiliary tree is ignored, still two other problems are to be dealt with: to what node of the right frontier of the developing tree should the adjunction be directed, and what should be the nuclearity pattern of the two descendents of the relation node in the auxiliary tree (the foot node and $sdt_{i+1}$)?

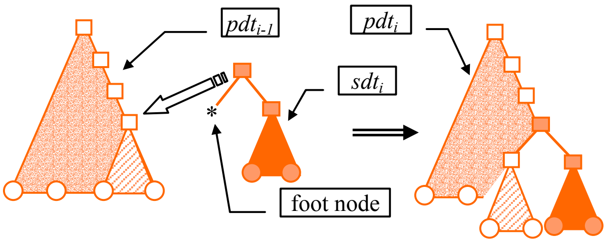


Figure 3: The adjoining operation

We define a function that records the number of co-references between *edu*s belonging to different *sdt*s, as follows.

If $SDT^D$ is the ordered set of all *sdt*s over discourse *D* (indexed from left to right, as the text unfolds), and $U^D$ is the set of all *edu*s in *D*, then:

$f: SDT^D \text{ x } \mathsf{P}(U^D) \text{ x } SDT^D \text{ x } \mathsf{P}(U^D) \rightarrow \mathsf{N}$

(where, if *x* is a set, $\mathsf{P}(x)$ is the power set of *x*, and $\mathsf{N}$ is the set of natural numbers), defined as follows: if $u_k$ is an *edu* on the terminal frontier of *sdt* $t_i$, and $u_l$ is an *edu* on the terminal frontier of *sdt* $t_j$ (* represents all *edu*s on the terminal frontier of $SDT\ \ t_i$, respectively $SDT\ \ t_j$) then:

$f(t_i, u_k, t_j, u_l)$ = number of antecedents belonging to unit $u_k$ of $t_i$ directly or indirectly referred by anaphors belonging to unit $u_l$ of $t_j$;

$f(t_i, *, t_j, u_l)$ = number of antecedents belonging to all units of $t_i$ directly or indirectly referred by anaphors belonging to unit $u_l$ of $t_j$ (if two or more anaphors in $u_l$ refer the same antecedent belonging to $t_i$ then *f* will count all of them);

$f(t_i, u_k, t_j, *)$ = number of antecedents belonging to the unit $u_k$ of $t_i$ directly or indirectly referred by anaphors belonging to all units of $t_j$;

$f(t_i, *, t_j, *)$ = number of antecedents belonging to the all units of $t_i$ directly or indirectly referred by anaphors belonging to all units of $t_j$;

$f(t_i, u_k, t_j, head(root(t_j)))$ = number of antecedents belonging to the unit $u_k$ of $t_i$ directly or indirectly referred by anaphors belonging to those units of $t_j$ that are contained in the head expression of the root of $t_j$.

The following rules give decision criteria with respect to the node of the right frontier of the developing tree $pdt_{i-1}$ where adjoining is to be operated:

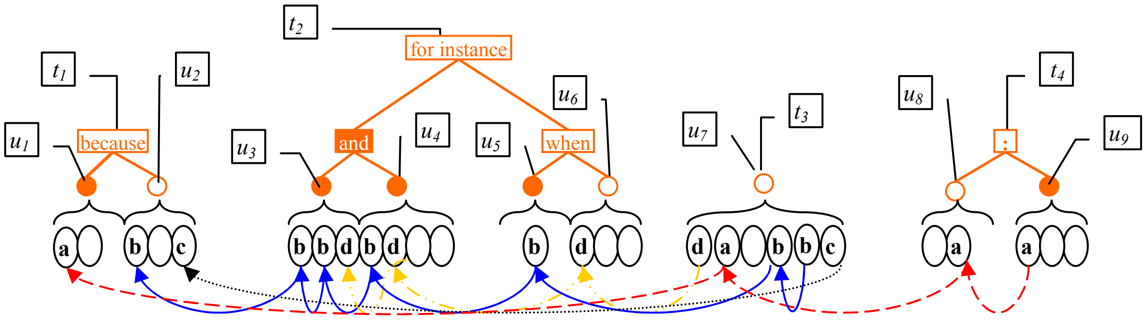


Figure 4: *Sdt*s and references for Example 2
(a = *Maria*, b = *Simon*, c = *the child*, d = *I*, empty = any other REs)

**Rule 1**:

If $f(t_p, u_k, t_i, *) > f(t_q, u_l, t_i, *)$, where $1 \leq p \neq q < i$ (meaning: the elementary tree $t_i$ comes in sequence after the elementary trees $t_p$ and $t_q$), $u_k$ is any unit of $t_p$ and $u_l$ is any unit of $t_q$, then if $n_k$ is the right frontier node of $pdt_{i-1}$ (that contains both $t_p$ and $t_q$) covering $u_k$ or the lowest node on the right frontier of $t_p$ that contains $u_k$ in its head expression, then the auxiliary tree stemmed out of $t_i$ is adjoined onto the node $n_k$.

**Rule 2**:

If $f(t_p, u_k, t_i, *) < f(t_q, u_l, t_i, *)$, with $1 \leq p \neq q < i$, but $u_l$ is not visible on the right frontier of $pdt_{i-1}$, then $u_l$ is ignored.

**Rule 3**:

If $f(t_p, u_k, t_i, *) = f(t_p, u_l, t_i, *)$, with $1 \leq p \neq q < i$ and $l < k$, then $u_l$ is ignored.

**Rule 4**:

If $f(t_p, u_k, t_i, *) = f(t_q, u_l, t_i, *)$, with $1 \leq p \neq q < i$, but $f(t_p, u_k, t_i, head(root(t_i))) < f(t_q, u_l, t_i, head(root(t_i)))$ then $t_i$ is adjoined into the node $n_l$, even if $p > q$;

**Rule 5**:

If there is no $p < i$ such that $f(t_p, u_k, t_i, *) > 0$, with $u_k$ any unit of $t_p$, or if such a $p$ exists but none of its $u_k$ are visible on the right frontier of $pdt_{i-1}$, then $t_i$ is adjoined onto the lowest most node of the right frontier of $pdt_{i-1}$ as a satellite of it.

The root of the auxiliary tree being adjoined always remains with the nuclearity of the node where the adjoining is being made. What still has to be decided is the nuclearity of the foot node (which will give the nuclearity of the node onto which the adjunction is being made, let's call it $u_k$) and of its sibling (the current *sdt* $t_i$):

**Rule 6**:

The node $u_k$, where the adjoining is being made will always be nuclear.

**Rule 7**:

If $f(t_p, head(root(u_k)), t_i, *) > 0$ then $t_i$ will be nuclear, otherwise it will be satellite.

We will display how the model works on the following example:

**Example 2**

*[Maria went alone to the market* $^{1}$*]* *because* *[Simon had to stay at home with the baby.* $^{2}$*] [Simon is a good friend of mine* $^{3}$*] and [he also helped me in a number of situations.* $^{4}$*] For instance [he was very helpful* $^{5}$*] when [I had the problem with the car.* $^{6}$*] [I think she has a lot of trust in him to let him alone with the child.* $^{7}$*] [You know how Maria is* $^{8}$*] : [she is not very hurried to give credit to anybody.* $^{9}$*]*

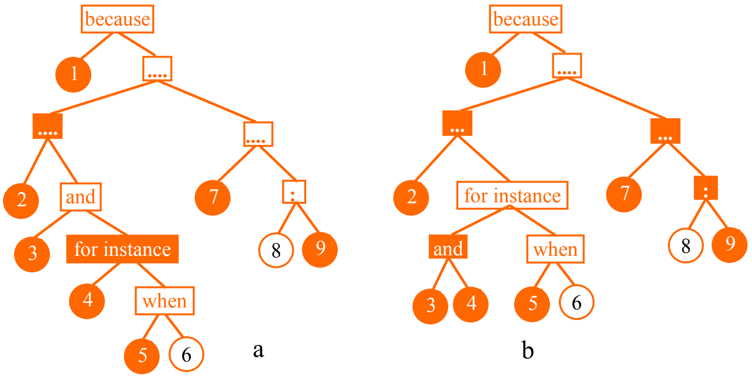


Figure 5: Correct (a) and computed (b) tree

Figure 4 shows the sequence of *sdt*s and the co-reference chains after the first two phases (in a pipe-line processing). The four *sdt*s are grouped together by three adjoining operations. At each step $i$, the function $f(t_p, u_k, t_{i+1}, *)$ is computed for each $p < i$, and any $u_k$ belonging to $t_p$, then the rules described above are applied. Only the first step is detailed here: the *sdt* $t_2$ has to be adjoined onto the first *pdt*, which is $t_1$: $f(t_1, u_1, t_2, *) = 0$; $f(t_1, u_2, t_2, *) = 3$; (corresponding to the 3 references for *Simon* in $u_2$, from $u_3$ and $u_4$). Applying rule 1 $t_2$ must be adjoined onto $t_1$ at the level of $u_2$. According to rule 6, $u_2$ will be nuclear, while $t_2$ will be satellite, according to rule 7 (the root of $u_2$ is the relation node because, whose head expression is $u_1$, and $f(t_1, u_1, t_2,*) = 0$). Figure 5 shows the correct tree, drawn by hand, compared with the computed one. The table below displays the vein expressions of the correct tree compared with the computed one.

| | golden | computed |
|---|---|---|
| 1 | 1 | 1 |
| 2 | 1 2 | 1 2 7 9 |
| 3 | 1 2 3 4 | 1 2 3 4 7 9 |
| 4 | 1 2 3 4 | 1 2 3 4 7 9 |
| 5 | 1 2 3 4 5 | 1 2 3 4 5 7 9 |
| 6 | 1 2 3 4 5 6 | 1 2 3 4 5 6 7 9 |
| 7 | 1 2 7 | 1 2 7 9 |
| 8 | 1 2 7 8 9 | 1 2 7 8 9 |
| 9 | 1 2 7 8 9 | 1 2 7 8 9 |

Let's try focused summaries for *Maria*, and *the child*. *Maria* is referred in *edu*s 1, 7, 8, and 9 (see example 2 and figure 4). The longest vein expression of these *edu*s (1 2 7 8 9) is the same in both golden and computed tree. Therefore, the summary focused on Maria will be:

*Maria went alone to the market because Simon had to stay at home with the baby. I think she has a lot of trust in him to let him alone with the child. You know how Maria is: she is not very hurried to give credit to anybody.*

*The child* is referred in *edu*s 2 and 7. The longest vein expression of these units is 1 2 7 in the golden tree and 1 2 7 9, in the computed tree. The summary focused on *the child* will be:

*Maria went alone to the market because Simon had to stay at home with the baby. I think she has a lot of trust in him to let him alone with the child. She is not very hurried to give credit to anybody.*

## 7 Data and experiments

The assumption on the correlation of vein structure with co-references was based on earlier experiments reported in (Cristea *et al*. 1998). An average, the results of experiments on Romanian and English texts revealed that in 99.1% references obey this conjecture.

Around 50 manually discovered cue-phrase patterns were used in the sentence-level tree construction, described in section 5. In order to validate the approach we developed two experiments. In the first experiment, *sdt*s were built based on the information given by an FDG parser, and in the second, *sdts* were generated based on our approach. We used a two pages excerpt from the original English version of G. Orwell's "1984" which contained 45 sentences out of which 19 were one-clause sentences, our attention being focused on the remaining 26 complex sentences. In the first experiment we used the FDG output both for extracting the units (the clauses) and for building the tree. It turned out that only 7 sentences (27%) could be resolved correctly while another 6 were only partially correct. In the second experiment for 20 sentences (76%) the method correctly indicated a unique *sdt*, while for the remaining 6 sentences more than one tree could be generated.

To validate the focused summarisation method guided by veins, a one-page text from "The Legends of Mount Olympus" of Al. Mitru, consisting of 62 *edu*s was used. 57 students, participants of the EUROLAN'01 summer school, were asked to extract a summary of the text, focused on *Hefaistos*, a secondary character in the extract. We then built a golden summary composed of units voted by more than a half of judges - 28 out of 57. For each judge, the recall and precision values were calculated. In the following table, the average of these values and the VT results are presented:

| | Judges' results | VT's results |
|---|---|---|
| Precision | 74.26% | 73.33% |
| Recall | 72.92% | 64.71% |

## 8 Discussions and further work

We are aware that errors can intervene in all processing steps of the described summarisation method (segmentation in *edu*s, detection of *sdt*s, anaphoric links detection). Further investigation will have to identify the overall trust in the method proposed.

An earlier investigation (Ide and Cristea, 2000) showed a correlation between the type of the anaphor (pronominal, proper nouns, definite or indefinite noun) and the percentage on which the antecedent is found along veins of the discourse structure. This suggests that the method of building the global structure of the discourse guided by references could be further sophisticated by using scores to account for type of antecedents.

The described method of inferring the discourse structure is deterministic in the sense that only one tree is obtained. Further development would have to transform it into a beam-search type of processing, close to the one described in (Cristea, 2000), in order to combine contribution from cue-phrases, and references with that given by centering. This way, the problem itself of partial trees proliferation caused by cue-phrases with multiple patterns, presently ignored, could also be tackled.

As demonstrated by the example in the previous section, the computed vein expressions have a tendency to be larger than needed, this yielding to longer summaries. More sophisticated integration rules, automatically discovered from a discourse structure annotated corpus by learning, could fix this problem.

Finally, it is to note that the structure of a discourse as a complete tree gives more information than properly needed (at least for summarization purposes). An underspecified type of representation, keeping, for instance, only vein expressions not the whole tree, could be a better solution.

## References

Ait-Mohtar, S. and Chanod, J.-P. 1997. Incremental Finite-State Parsing. *Proceedings of ANLP'97*, Washington.

Cole, R.A., Mariani, J., Uszkoreit, H, Zaenen, A. and Zue, V. 1995. Survey of the State of the Art in Human Language Technology.

Cristea, D. and Webber B.L. (1997). Expectations in Incremental Discourse Processing. *Proceedings ofACL/EACL'97*, Madrid.

Cristea, D., Ide, N. and Romary, L. (1998). Veins Theory: A Model of Global Discourse Cohesion and Coherence, *Proceedings of Coling/ACL'98*, Montreal.

Cristea, D., Ide, N., Marcu, D. and Tablan, V. 2000. Discourse Structure and Co-Reference: An Empirical Study. *Proceedings of The 18th International Conference on Computational Linguistics COLING'2000*, Luxembourg.

Cristea, D. and Dima, G.-E. 2001. An Integrating Framework for Anaphora Resolution. *Information Science and Technology*, Romanian Academy Publishing House, Bucharest, 4(3).

Cristea, D., Postolache, O.-D., Dima, G.-E. and Barbu, C. 2002a. AR-Engine – a framework for unrestricted co-reference resolution. *Proceedings of Language Resources and Evaluation Conference - LREC 2002*, Las Palmas, vol. VI: 2000-2007.

Cristea, D., Dima, G.E., Postolache, O.-D. and Mitkov, R. 2002b. Handling complex anaphora resolution cases, *Proceedings of the Discourse Anaphora and Anaphor Resolution Colloquium*, Lisbon.

Cristea,D. (2000): An Incremental Discourse Parser Architecture, D. Christodoulakis (Ed.) *Proceedings of the Second International Conference - Natural Language Processing - NLP 2000*, Patras, Greece, June 2000. Lecture Notes in Artificial Intelligence 1835, Springer.

Fox, B. 1987. Discourse Structure and Anaphora, Cambridge University Press.

Grosz, Barbara J., Aravind K. Joshi, Scott Weinstein. 1995. Centering: a Framework for Modelling the Local Coherence of Discourse. Computational Linguistics, 21(2).

Ide,N., Cristea,D. (2000): A Hierarchical Account of Referential Accessibility. *Proceedings of The 38th Annual Meeting of the Association for Computational Linguistics, ACL'2000*, Hong Kong.

Knott, A. and Dale, R. 1992. Using Linguistic Phenomena to Motivate a Set of Coherence Relations. Discourse Processes 18(1).

Mani, I. 2001. Automatic Summarization. Natural Language Processing series. John Benjamins Publishing Co., Amsterdam.

Marcu, D. 2000. *The Theory and Practice of Discourse Parsing and Summarization.* The MIT Press.

Puscasu, G. forthcoming. Elementary discourse unit segmentation. Dissertation thesis. “Al.I.Cuza” University of Iasi.

Soricutu and Marcu (2003) Sentence Level Discourse Paring using Syntactic and Lexical Information, *Proceedings of HLT/NAACL – 2003*, Edmonton.

Tufiş, Dan 1999. Tiered Tagging and Combined Classifiers. F. Jelinek, E. Nöth (Eds) *Text, Speech and Dialogue, Lecture Notes in Artificial Intelligence, 1692*, Springer.

Vonk, W., Hustinx, L. and Simons, W. 1992. The Use of Referential Expressions in Structuring Discourse. *Language and Cognitive Processing.* 7 (3-4).

# From manual to automatic annotation of coreference

**Renata Vieira*, Caroline Gasperin*, Rodrigo Goulart***
PIPCA - Unisinos
São Leopoldo, Brazil
{renata,caroline,rodrigo}@exatas.unisinos.br

## Abstract

We present experiments on manual annotation of coreference in Portuguese texts motivated by the goal of developing and evaluating a coreference resolution tool. The tool architecture, designed to deal with multi-level annotation and to allow easy combination of heuristics and configuration of parameters, is also described in the paper.

## 1 Introduction

We are studying coreference in romance languages with the goal of developing a multi-lingual tool for coreference resolution. In this paper we present results in corpora annotation of coreference in Portuguese texts[1]. These experiments serve as background for the development of our tool. First, the tasks to be performed by the tool should be tasks that subjects can perform themselves and furthermore, there must be agreement in the analyses. Secondly, our experiments are developed to raise relevant features to be taken into account in the process of coreference resolution. The tool we are designing handles multi-level annotation encoded according to recently proposed standards.

In the next section, our experiments on Portuguese coreference annotation are detailed. In section 3, we present the designing principles of our tool. Conclusions of this work are presented in section 4.

## 2 Manual annotation of coreference

Coreference on natural language texts consists on two or more expressions referring to a same discourse entity. When they follow one another in the text we refer to the previous expression in the sequence as antecedent. We refer to terms under analysis in our studies as coreferent terms. Coreferent terms can be of different types, for example:

- pronominal: the coreferent term is a pronoun. Ex: *The boy read the book, but* ***he*** *didn't like it.*
- definite description: the coreferent term is a noun phrase preceded by a definite article. Ex: *I bought a house.* ***The house*** *is far from here.*
- demonstrative descriptions: the coreferent term is a noun phrase preceded by a demonstrative pronoun. Ex: *I bought a house.* ***This house*** *will be mine forever.*

In our work we have undertaken three experiments on manual annotation, considering two types of coreferent expressions (definite descriptions and demonstratives) and different annotation methodologies (as described later in the paper). We studied definite descriptions and demonstrative noun phrases separately, observing different features for each one. The annotation methodology has evolved by considering questions related to inter-annotator agreement. Table 1 presents an overview of the Portuguese corpora we studied.

The first two experiments presented next, were first presented in detail in (Salmon-Alt and Vieira, 2002; Vieira et al., 2002b; Vieira et al., 2002a).

[1] Both Brazilian and European Portuguese.

* Research Grant CNPq- Brazil.

Table 1: Language resources.

| Experiment | Size (words) | Number of cases |
|---|---|---|
| Exp 1 | 5000 | 541 definites |
| Exp 2 | 50000 | 880 demonstratives |
| Exp 3 | 6795 | 730 definites |

These previous works refer to both Portuguese and French languages.

We used MMAX (Müller and Strube, 2001) tool for manual annotation of coreference in the three experiments. MMAX was suitable for annotating the resources in a format as close as possible to the MATE recommendations, especially concerning the use of XML, the stand-off annotation principle and the compatibility with proposed coreference encoding guidelines (Poesio, 2000).

### 2.1 First experiment

The corpus used in the first experiment was composed of European Portuguese written question-answer pairs published in the Official Journal of the European Commission. Our classes of analyses were based on the analyses of English texts presented in (Poesio and Vieira, 1998), with the difference that we divided the *Bridging* class of their analyses into two different classes, separating coreferent (*Indirect Anaphora*) and non-coreferent (*Other Anaphora*) cases. The study aimed to verify if we could get a similar distribution of types of definite descriptions for Portuguese and English, which would serve as an indication that the same heuristics tested for English (Vieira and Poesio, 2000) could apply for Portuguese. The main annotation task in this experiment was identify antecedents and classify each definite description (d) into one of the following four classes:

- *Direct Coreference* d corefers with a previous expression a; d and a have the same nominal head:

  a. A Comissão tem conhecimento **do livro**... (the Commission knows *the book*)

  d. a Comissão constata ainda que **o livro** não se debruça sobre a actividade das várias ... (the Commission remarks that *the book* ignores the activity of various)

- *Indirect Coreference* d corefers with a previous expression a; d and a have different nominal heads:

  a. a circulação **dos cidadões** que dirigem-se (...) (the flow of *the citizens* heading to...)

  d. do controle **das pessoas** nas fronteiras (the control of *the people* in the borders)

- *Other Anaphora* d does not corefer with a previous expression a, but depends for its interpretation on a:

  a. **o recrutamento de pessoal científico e técnico**... (*the recruitment of scientific and technical employees*)

  d. **as condições de acesso à carreira científica** (*the conditions of employment for scientific jobs*)

- *Discourse New* the interpretation of d does not depend on any previous expression:

  d. o livro não se debruça sobre **a actividade das várias organizações internacionais**... (the book ignores *the activity of various international organisation...*)

#### 2.1.1 Results

Table 2 presents the distribution of the annotation among the classes presented above. The distribution of uses is similar to previous studies for English texts.

Around 40% of the cases are discourse new, for these cases resolution does not apply. This indicates that the heuristics to identify discourse new descriptions proposed for English should be tested for Portuguese. These heuristics are mainly based on syntactic complexity. The syntactic structure of discourse new descriptions was analyzed for Portuguese. It was found that they were modified (by adjectives, prepositional phrases and relative

Table 2: Experiment 1.

| Classification | Ann 1 | Ann 2 | Average |
|---|---|---|---|
| Direct coreference | 96 | 179 | 25.4 |
| Indirect coreference | 51 | 45 | 8.9 |
| Other anaphora | 46 | 77 | 11.4 |
| Discourse new | 266 | 198 | 42.9 |
| Not classified | 82 | 42 | 11.5 |
| TOTAL | 541 | 541 | 100.0 |

clauses) in approximately 50% of the cases. These findings indicate that the heuristics developed for English may be applied to Portuguese.

Those classes that rely on common sense knowledge (indirect and other anaphora), and are, therefore, of difficult computational treatment, account for 20% of the total.

The agreement (given by Kappa) among the annotators was low, K = 0.44. This was worse agreement than the experiments made with English corpus. This could be related to the inclusion of a fourth class in the analysis (the splitting of bridging into indirect coreference and other anaphora). The motivation for introducing this class was to distinguish coreferent from non-coreferent (associative) uses like in *house* - *the door*. At first we considered that a better specification of the classes could improve the agreement results, but the results lead us to conclude that it could be more difficult to the annotators to deal with a greater number of choices.

### 2.2 Second experiment

In this experiment we analysed demonstrative noun phrases. Our classes here, similar to those used for definite descriptions, serve to estimate the frequency of antecedents that are noun phrases, the frequency of coreferential and other relations between demonstratives and their NP antecedents. We were also verifying the frequency in which the NP antecedent has the same head noun of the demonstrative. The reason why we are isolating nominal antecedents from other expressions such as verb phrases, sentences or paragraphs is that this can give us an idea of how well a system for coreference resolution of demonstratives can perform on the basis of nominal expression relations only.To gather this sort of knowledge, each demonstrative description (d) was classified into one of the following classes:

- *Direct coreference*, as before:

  a. e prestar **às autoridades gregas** (*to the greek authorities*)

  d. para **essas autoridades** (*these authorities*)

- *Indirect coreference*, as before:

  a. **À Albânia**

  d. ajudar **este país** a atingir (*this country*)

- *Other kind of anaphora* the antecedent is not a nominal expression or the relation between demonstrative and its antecedent is not a coreference relation.

  a. **o ano de 1993 será essencialmente consagrado ao apoio de experiências-piloto de informação dos jovens na Europa** (*the year of 1993 will be important to the experiments ...*)

  d. **nesse contexto** *in this context*

  a. **adoptar medidas de âmbito nacional** (*to adopt measures...*)

  d. **essa adopção** *this adoption*

#### 2.2.1 Results

Table 3 shows the results obtained in the second experiment. The results show that demonstratives are context dependent, with nearly half of them being coreferent to previous NPs. The other half is either coreferent with antecedents that are not NP or not coreferent. Demonstratives whose antecedents were not explicitly marked by the annotator were included in other anaphora class; most of these cases were antecedents corresponding to more than one paragraph in the text.

Table 3: Experiment 2.

| Classification | Ann 1 | Ann 2 | Average |
|---|---|---|---|
| Direct coreference | 80 | 74 | 31.7 |
| Indirect coreference | 60 | 49 | 22.4 |
| Other anaphora | 77 | 66 | 29.4 |
| Discourse new | 0 | 0 | 0 |
| Not marked | 26 | 54 | 16.5 |
| TOTAL | 243 | 243 | 100.0 |

We calculated Kappa for three classes (direct coreference, indirect coreference, other). We found K = 0.65 for Portuguese demonstratives. These results show better agreement than for previous experiments related to four different classes for definite descriptions. The improvement might be related to the reduced number of classes and the kind of distinction involved.

## 2.3 Third experiment

In the third experiment we analysed again definite descriptions but adopting a different annotation methodology. The change in annotation methodology is related to the low agreement observed in the first experiment with definite descriptions. This could be due to some difficulties in the annotation process. There, all the annotation had to be done in a single step. In order to simplify annotators' tasks, we decided to split the annotation process in 4 steps (the first on is done by just one annotator and the others by two):

1. selecting coreferent terms;

2. identifying the antecedent of coreferent terms selected in sep 1, if there is one;

3. classifying coreferent terms: if there is an antecedent and it is coreferent, their relation should be classified (direct or indirect);

4. classifying non coreferent terms: if the expression doesn't have an antecedent or if it has a non coreferent antecedent classify the non coreferent relation (discourse new or other anaphora).

### 2.3.1 Results

The results here show the similar distribution of the previous experiments for definite descriptions. Compared to the total (730 cases) we have about half of them classified as discourse new descriptions, which account for about 70% of non-coreferent cases. Among the coreferent cases the number of direct coreference is twice the number of indirect coreference. Regarding the agreement among annotators, we see that after dividing our experiments in steps, the agreement of our annotators increased a little compared to experiment 1. Considering 4 classes (direct, indirect, discourse new, and other anaphora) we have K = 0.52. Considering Kappa for each step in the annotation task we have for step 2 (coreferent X non coreferent) K = 0.76, for step 3 (direct X indirect) K = 0.57 and for step 4 (other anaphora X dicourse new) K = 0.29. Clearly, the difficult class to analyse is the non coreferent class, that is the distinction between those cases introduced in text by an associate entity and cases based on subject's previous world knowledge. This confirms previous work done for English. Regarding the development of a tool for coreference resolution in texts, we can only hope to be able to identify coreferent from non-coferent terms, since this is the task that can be performed by speakers.

## 3 Automatic annotation of coreference

Based on the corpus studies presented in the previous section, we are developing a tool that is able to identify automatically the antecedents of coreferent expressions. The tool is designed on the basis of standard encoding of linguistics resources (Ide and Romary, 2002). It deals with multi-level annotated resources combining POS, syntactic and coreference

Table 4: Experiment 3.

| Step 2 | All cases | Ann 1 | Ann 2 | Average |
|---|---|---|---|---|
| | Coreferent | 218 | 218 | 29.9 |
| | Non coreferent | 512 | 508 | 69.9 |
| | None | 0 | 4 | 0.2 |
| | TOTAL | 730 | 730 | 100.0 |
| Step 3 | Coreferent | Ann 1 | Ann 2 | Average |
| | Direct | 125 | 151 | 63.3 |
| | Indirect | 93 | 67 | 36.7 |
| | Other | 0 | 0 | 0.0 |
| | None | 0 | 0 | 0.0 |
| | TOTAL | 218 | 218 | 100.0 |
| Step 4 | Non coreferent | Ann 1 | Ann 2 | Average |
| | Discourse new | 354 | 382 | 72.2 |
| | Other anaphora | 147 | 114 | 25.6 |
| | Other | 11 | 12 | 2.2 |
| | None | 0 | 0 | 0.0 |
| | TOTAL | 512 | 508 | 100.0 |

```
 ...
<word id="word_92">as</word>
<word id="word_93">autoridades</word>
<word id="word_94">gregas</word>
 ...
<word id="word_135">essas</word>
<word id="word_136">autoridades</word>
 ...
```

Figure 1: Words file

information.

### 3.1 Input Data Format

Input data for our tool follows MMAX´s word and markable file formats. MMAX formats are shown on Figure 1.

The coreference annotation produced by manual analysis is encoded as shown on Figure 2, where the attribute *span* indicates the words that form each <markable>, the attribute *pointer* indicates the antecedent identifier, and the attribute *classification* corresponds to the classes presented earlier.

We also produce compatible POS and syntactic information[2] used for coreference resolution from the Portuguese parser PALAVRAS (Bick, 2000). PALAVRAS output is converted into a set of XML

```
 ...
<markable id="markable_3"
           span="word_92..word_94"
           pointer=""
           np_form="defNP"
           classification=""/>
 ...
<markable id="markable_5"
           span="word_135..word_136"
           pointer="markable_3"
           np_form="demNP"
           classification="indirect"/>
 ...
```

Figure 2: Markables file

```
 ...
<word  id="word_92">
  <art canon="o"
       gender="F"
       number="P">
    <secondary_art tag="artd"/>
  </art>
</word>
 ...
<word  id="word_93">
  <n   canon="autoridade"
       gender="F"
       number="P">
</word>
 ...
```

Figure 3: POS file

[2] Presented in detail in (Gasperin et al., 2003)

```
...
<chunk id="chunk_15" ext="np" span="word_92..word_94">
  <chunk id="chunk_16" ext="h" span="word_93"/>

</chunk>
...
<chunk id="chunk_27" ext="np" span="word_135..word_136">
   <chunk id="chunk_28" ext="h" span="word_136"/>
</chunk>
...
```

Figure 4: Chunks file

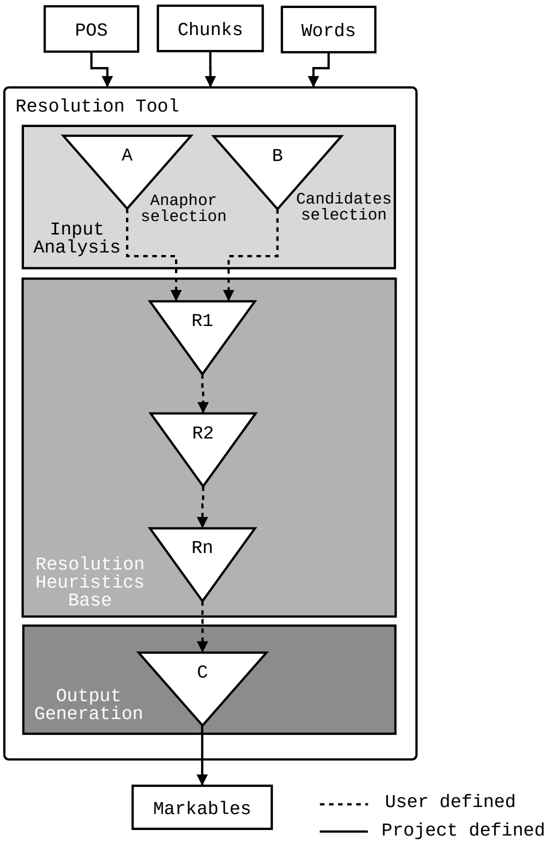


Figure 5: General architecture

files: the words file; a file with the part-of-speech (POS) categories for each word, an example is given in Figure 3, where the element *n* indicates a noun; and a file with the sentences syntactic structure, represented by <chunk> elements. A <chunk> represents a structure inside the sentence that can contain other child chunks, an example is shown in Figure 4, where the parent <chunk> (attribute @id =”chunk_7”) is a noun phrase (attribute ext=”np”) and the child <chunk> 8 is the header of the parent <chunk> (attribute ext=”h”).

### 3.2 General arquitecture

The design of the tool is based on “Pipes & Filters” proposed by Gamma (Gamma et al., 1995). The tool is composed by a set of filters that transform the data flowing through it. Each filter corresponds to a XSL script (which are simple and flexible) and implements heuristics for coreference resolution or parsers input and output of elements. The filters are combined into layers as shown in Figure 5.The first layer, “Input Analysis”, uses input data to generate two new data elements <anaphor> and <candidate> (Figure 6 and 7).

The <anaphor> elements are expressions to be resolved (definite, demonstratives or other noun phrases), which are extracted from corpus. This extraction is made by looking for certain attribute values in <chunk> elements. The presence of the definite articles in NPs, for instance, identifies definite descriptions.

The <candidate> elements are antecedent candidates for coreference resolution, different choices may also apply according to the heuristics adopted.

We have considered NPs as candidates, for instance.

These new elements add one annotation level to our resources. They are used together with the

```
...
<anaphor span="word_92..word_94"
         pointer=""/>
...
<anaphor span="word_135..word_136"
         pointer="word_92..word_94"/>
...
```

Figure 6: Anaphor elements

```
...
<candidate span="word_92..word_94"/>
...
<candidate span="word_135..word_136"/>
...
```

Figure 7: Candidate elements

other annotation levels in the layer called “Resolution Heuristics Base” (RHB). RHB layer is composed by a set of filters corresponding to coreference resolution heuristics. The user of the tool can define the combination of heuristics and parameters to be taken into account at each execution.

Some resolution heuristics may be based on the comparison of the head nouns of <anaphor> and <candidate>. Their span values point to the corresponding <chunk> nodes. The heads are the childs of these chunks, having ext attribute value equal to "h" (chunk_16 and chunk_28). They are accessible through words and POS files ("autoridades"). When a suitable antecedent is found the pointer attribute of the <anaphor> will take the span attribute from the matching <candidate>. Other heuristics may combine information from POS and Chunks regarding the candidates and anaphors. The use of linguistic information changes according to the rules to be applied by the tool, as specified by the user.

The last layer, “Output Generation”, takes RHB´s output and adapts it according to the required format. In our case we generate MMAX markables, so we can evaluate our output with manually annotated corpus and also visualize the coreference chains in MMAX. However, other formats can be generated, for instance, the virtual annotation language proposed in (Ide and Romary, 2003; Ide and Romary, 2001). The combination of heuristics (different rules, different sequences and paramenters) may be defined and tested, since the user defines conections of RHB filters.

## 4 Conclusion

We have presented our work towards the designing of a coreference annotation tool on the basis of manual annotation experiments. Our experiments show that definite descriptions are commonly used to introduce new discourse elements in Portuguese texts, confirming previous findings for English. We also compared the use of definite descriptions to another type of noun phrases, commonly considered as anaphoric: demonstrative noun phrases. As opposite to definite descriptions, they are mainly text dependent for their interpretation (coreferent or anaphoric); antecedent NPs account for at least 50% of the cases (direct and indirect coreference). Therefore, the same heuristics applied to direct coreference of definite descriptions may be used for demonstratives, by the distribution we have an idea that the heuristics may account for 30% of the cases.

Following our corpus studies, we presented the general architecture of a tool for automatic coreference resolution. This tool process parsed corpus encoded in XML according to recommendations of the standards under development for corpora annotation (XCES (Ide and Romary, 2002), ISO TC37 SC4). The advantage of having data encoded in XML is the possibility of using the existing tools for handling XML data, as well as existing tools for manual coreference annotation (MMAX).

Regarding the tasks to be done automatically by the tool, we will concentrate in distinguishing between coreferent from non-coferent terms, and resolve co-referent terms, we will not try to resolve other anaphora, since we will not be able to evaluate this task.

Although we concentrated this paper on the study of Portuguese text, the tool is being conceived to treat multilingual corpora, and we are first considering romance languages. Studies on French corpora (Salmon-Alt and Vieira, 2002; Vieira et al., 2002b; Vieira et al., 2002a), is being conducted along with our experiments for Portuguese.

## Acknowledgements

We are grateful to CNPq, INRIA and FAPERGS for financial support. We would like to thank the help of Eckhard Bick, Christoph Müller, Michael Strube, Paulo Quaresma and Susanne Salmon-Alt.

## References

Eckhard Bick. 2000. *The Parsing System PALAVRAS: Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework*. Ph.D. thesis, Århus University, Århus.

Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. 1995. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Addison-Wesley Publishing Company, New York.

Caroline Gasperin, Renata Vieira, Rodrigo Goulart, and Paulo Quaresma. 2003. Extracting xml syntactic chunks from portuguese corpora. In *Traitement automatique des langues minoritaires - TALN 2003*, Btaz-sur-mer, France.

Nancy Ide and Laurent Romary. 2001. Common framework for syntactic annotation. In *Proceedings of ACL'2001*, pages 298–305, Toulouse.

Nancy Ide and Laurent Romary. 2002. Standards for language resources. In *Proceedings of the LREC 2002*, pages 839–844, Las Palmas de Gran Canaria.

Nancy Ide and Laurent Romary. 2003. Encoding syntactic annotation. In Anne Abeillé, editor, *Building and Using Syntactically Annotated Corpora (in press)*. Kluwer, Dordrecht.

Christoph Müller and Michael Strube. 2001. MMAX: A tool for the annotation of multi-modal corpora. In *Proceedings of the IJCAI 2001*, pages 45–50, Seattle.

Massimo Poesio and Renata Vieira. 1998. A corpus-based investigation of definite description use. *Computational Linguistics*, 24(2):183–216.

Massimo Poesio. 2000. Coreference.mate dialogue annotation-deliverable d2.1. Technical report, http://www.ims.uni.stuttgart.de/projekte/mate/mdag, Jan.

Susanne Salmon-Alt and Renata Vieira. 2002. Nominal expressions in multilingual corpora: Definites and demonstratives. In *Proceedings of the LREC 2002*, Las Palmas de Gran Canaria.

Renata Vieira and Massimo Poesio. 2000. An empirically-based system for processing definite descriptions. *Computational Linguistics*, 26(4):525–579.

Renata Vieira, Susanne Salmon-Alt, Caroline Gasperin, Emmanuel Schang, and Gabriel Othero. 2002a. Coreference and anaphoric relations of demonstrative noun phrases in multilingual corpus. In *Proceedings of the DAARC 2002*, Estoril.

Renata Vieira, Susanne Salmon-Alt, and Emmanuel Schang. 2002b. Multilingual corpora annotation for processing definite descriptions. In *Proceedings of the PorTAL 2002*, Faro.

# The Semantic Web Needs Anaphora Resolution

**Rodolfo Delmonte**
Dipartimento Scienze del Linguaggio
Università Ca' Foscari
Ca' Garzoni-Moro - San Marco 3417 - 30124 VENEZIA
e-mail: delmont@unive.it website - http://project.cgm.unive.it

## 1. Introduction

This paper will address the importance of using anaphora resolution in the creation of knowledge databases or ontologies for the Semantic Web. In the current debate existing between promoters of the Semantic Web [1] and NLP oriented practitioners of the web like the creators of the START QA system [6;13], the former believe that by simply establishing a standard in the format of the information to be published on the web they will allow natural language communication, i.e. question/answering, to ensue. On the contrary, NLP-oriented practitioners of the web find it totally misleading and insufficient in itself: they assume that natural language facilities must come first. This debate is reminiscent of a similar debate existing between people working in the NLP paradigm who have witnessed an increasing gap dividing on the one side, Knowledge Representation oriented researchers – usually engineers, their implementations being some variant of expert systems; on the other side, NLP oriented computational linguists who have continued working within the domain of a syntactic-semantic approach, not disregarding the relevance of knowledge of the world, but trying to reduce its impact on the overall architecture of a Text/Speech Understanding System.

## 2. Semantic Web and RDF

The Semantic Web is an extension of the World Wide Web that facilitates the exchange of machine-readable information [1]. At the heart of the Semantic Web is a technology known as the Resource Description Framework (RDF) [2], a portable XML-based representation of semantic networks or labeled directed graphs. RDF serves as the lingua franca of the Semantic Web, making it possible for programs to exchange ontologically encoded information, such as authorship, annotations, topic labels, content and customer satisfaction ratings, etc. over the Internet using a standard format.

The basic RDF data model consists of a series of nodes in a graph, which represent objects, and arcs connecting nodes, which represent relationships between objects. An arc (also called a predicate) in conjunction with the two nodes it connects is collectively termed a statement in RDF parlance and is the unit of information in the RDF model. Furthermore, nodes and predicates are named by uniform resource identifiers (URIs), which in conjunction with the XML Namespace standard [3], allow object identifiers to be globally unique. These standards also allow predicate vocabularies to be defined, which give standard names to relationships such as "has name", "published by", etc.

RDF is the lingua franca of the Semantic Web, providing a standardized data model for allowing interchange of metadata across the Internet. In short, it is a portable representation of a semantic network, a labeled directed graph. The basic unit of information in RDF is the statement, consisting of a triple of subject (a resource), predicate (an arc in the graph), and object (another resource or a literal). In its original form, RDF was meant for consumption by computers, not humans.

Most of the Web's content today is designed for humans to read, not for computer programs to manipulate meaningfully. Computers can adeptly parse Web pages for layout and routine processing—here a header, there a link to another page—but in general, computers have no reliable way to process the semantics, as [1] assumes. The authors continue by noting that the task is complicated,

> For the semantic web to function, computers must have access to structured collections of information and sets of inference rules that they can use to conduct automated reasoning. Artificial-intelligence researchers have studied such systems since long before the Web was developed. Knowledge representation, as this technology is often called, is currently in a state comparable to that of hypertext before the advent of the Web: it is clearly a good idea, and some very nice demonstrations exist, but it has not yet changed the world. It contains the seeds of important applications, but to realize its full potential it must be linked into a single global system.
> … The challenge of the Semantic Web, therefore, is to provide a language that expresses both data and rules for reasoning about the data and that allows rules from any existing knowledge-representation system to be exported onto the Web. Adding logic to the Web—the means to use rules to make inferences, choose courses

> of action and answer questions—is the task before the Semantic Web community at the moment. A mixture of mathematical and engineering decisions complicate this task.

Their conclusions are a mixture of hope and lack of realism: agents should be endowed with NLP capabilities which are far beyond current technology,

> The real power of the Semantic Web will be realized when people create many programs that collect Web content from diverse sources, process the information and exchange the results with other programs. The effectiveness of such software agents will increase exponentially as more machine-readable Web content and automated services (including other agents) become available.

Our central idea for bridging this gap between the core Semantic Web data model and natural language revolves around the suggestion offered by the authors of the START system, i.e. the application of the natural language annotations technology they employed in Start. In essence, they propose to "tag" fragments of RDF with language to facilitate access. In fact, annotation support has already been explored in the context of the Semantic Web. Projects such as Annotea [4] and CREAM [5] are developing frameworks for creating and exchanging RDF-encoded annotations between Semantic Web clients. RDF also forms the basis of Haystack's data model [6], meaning that annotations created from within the Haystack environment are usable by other RDF-enabled software packages. Furthermore, natural language technology enables users to query information stores using everyday language without resorting to specialized and often unintuitive query languages.

We believe that natural language annotations are not only an intuitive and helpful extension to the Semantic Web, but will also assist in the deployment and adoption of the Semantic Web itself. As the authors comment, the primary barrier to the success of the Semantic Web is a classic chicken-and-egg problem: people will not spend extra time marking up their data unless they perceive a value for their efforts, and metadata will not be useful until a "critical mass" has been achieved. Although researchers have been focusing on ontology editors to reduce barriers to entry, such initiatives are not sufficient to overcome the hurdles.

## 3. Large-scale Syntactic-Semantic Indexing

Although full syntactic and semantic analysis of open-domain natural language text is beyond current technology, we believe that it is possible to augment RDF's manual-annotation-based approach with automatically built annotations by extracting a limited subset of relations from unstructured text; in short, shallow/partial text understanding on the level of semantic relations, an extended label including Predicate-Argument Structures and other syntactically and semantically derivable head modifiers and adjuncts. This approach is promising because it attempts to address the well-known shortcomings of standard "bag-of-words" information retrieval/extraction techniques without requiring manual intervention: it develops current NLP technologies which make heavy use of statistically and FSA based approaches to syntactic parsing.

To this end, we have developed a robust version of GETARUNS [10], a prototype question answering system based on matching semantic relations derived from the question with those derived from the corpus (for START see Lin, 2001, 2003). These relations are simplified versions of RDF's ternary expressions (see also Katz, 1997), but can be generated automatically and indexed on a large scale.

Currently, START's system answers millions of natural language questions about places (e.g., cities, countries, lakes, coordinates, weather, maps, demographics, political and economic systems), movies (e.g., titles, actors, directors), people (e.g., birth dates, biographies), dictionary definitions, and much, much more. Because START performs sophisticated syntactic and semantic processing of questions to pinpoint the exact information need of a user, questions can be answered with remarkable precision. The authors report that in the period from January, 2001 to March, 2002, Start and Omnibase replied to over 326 thousand queries from users all over the world. Of those, 67% were answered successfully by the system (59% of the questions answered were handled by Omnibase) [7]. Table 1. here below shows the ternary structure induced for a sample of typical questions in START:

Table 1: Some sample questions that can be handled by an object–property–value model of Web data.

| Question | Object | Property | Value |
|---|---|---|---|
| Who wrote the music for Star Wars? | Star Wars | composer | John Williams |
| Who invented dynamite? | dynamite | inventor | Alfred Nobel |

| | | | |
|---|---|---|---|
| How big is Costa Rica? | Costa Rica | area | 51,100 sq. km |
| How many people live in Kiribati? | Kiribati | population | 94,149 |
| What languages are spoken in Guernsey? | Guernsey | languages | English, French |
| Show me paintings by Monet | Monet | works | [images] |

## 4. Ternary Expressions as Predicate-Argument Structures

As a result, researchers like Lin, Katz and Litkowski have started to work in the direction of using NLP to populate a database of RDFs, thus creating the premises for the automatic creation of ontologies to be used in the SW. People have come to believe that the problem of NLP might be reduced to that of creating ternary expressions; in turn the problem of ontologies has also been reduced to that of having ternary expressions available. This reduction is in our opinion absolutely misleading and not to further: we want to make it clear that in no way RDFs and ternary expressions may constitute a formal tool sufficient to express the complexity of natural language texts.

RDFs are assertions about the things (people, Webpages and whatever) they predicate about by asserting that they have certain properties with certain values. So, on the one side, we may agree with the fact that this is a natural way of dealing with data handled by computers most frequently; however, it also a fact that this is not equivalent to being useful for natural language understanding. The misconception seems to be deeply embedded in the nature of RDFs: they are directly comparable to attribute-value pairs and DAGs which are also the formalism used by most recent linguistic unification-based grammars. From the logical and semantic point of view RDFs also resemble very closely first order predicate logic constructs: but we must remember that FOPL is as such insufficient to describe natural language texts. In its basic abstract representation, a ternary expression may be represented as follows:

Ternary expressions(T-expressions), <subject relation object>.

Certain other parameters or relational valuies (adjectives, possessive nouns, prepositional phrases, etc.) can be used to create additional T-expressions in which prepositions and several special words may serve as relations. For instance, the following simple sentence

(1) Bill surprised Hillary with his answer

will produce two T-expressions:

(2)i. <<Bill surprise Hillary> with answer>

□□i. <answer related-to Bill>

In Litkowski's system the key step in their question-answering prototype was the analysis of the parse trees to extract semantic relation triples and populate the databases used to answer the question. A semantic relation triple consists of a discourse entity, a semantic relation which characterizes the entity's role in the sentence, and a governing word to which the entity stands in the semantic relation. The semantic relations in which entities participate are intended to capture the semantic roles of the entities, as generally understood in linguistics. This includes such roles as agent, theme, location, manner, modifier, purpose, and time. Surrogate place holders included are "SUBJ," "OBJ", "TIME," "NUM," "ADJMOD," and the prepositions heading prepositional phrases. The governing word was generally the word in the sentence that the discourse entity stood in relation to. For "SUBJ," "OBJ," and "TIME," this was generally the main verb of the sentence. For prepositions, the governing word was generally the noun or verb that the prepositional phrase modified. For the adjectives and numbers, the governing word was generally the noun that was modified.

### 4.1 Ternary Expressions are better than the BOWs approach, but…

People working advocating the supremacy of the TEs approach were reacting against the Bag of Words approach of IR/IE in which words were wrongly regarded to be entertaining a meaningful relation simply on the basis of topological criteria: normally the distance criteria or the more or less proximity between the words to be related. Intervening words might have already been discarded from the input text on the basis of stopword filtering. Stopwords list include all grammatical or close type words of the language: these are considered useless for the main purpose of IR/IE practitioners seen that they cannot be used to denote concepts. Stopwords constitute what is usually regarded the noisy part of the channel in information theory. However, it is just because the redundancy of the information channel is guaranteed by the presence of grammatical words that the message gets appropriately computed by the subject of the communication process, i.e. human beings. Besides, entropy should not to be computed in terms of number of words or letters of the alphabet, but in number of semantic and syntactic relation entertained by open class words (nouns, verbs, adjectives, adverbials) basically by virtue of closed class words. Redundancy should then be computed on the basis of the ambiguity intervening when enumerating those relations, a very hard task to accomplish which has never been attemped yet, at least to my knowledge.

What people working with TEs noted was just the problem of encoding relations appropriated, at least some of these relations. The IR/IE BOWs approach suffers (at least) from Reversible Arguments Problem (Katz & Lin)

- What do frogs eat? vs  What eats frogs?

The verb "eat" entertains asymmetrical relations with its SUBJect and its OBJect: in one case we talk of the "eater", the SUBJect and in another case of the "eatee", the OBJect. Other similar problems occur with TEs when the two elements of the relation have the same head, as in:

-The president of Russia visited the president of
China. Who visited the president?

The question will not be properly answered in lack of some clarification dialogue intervening, but the corresponding TEs should have more structure to be able to represent the internal relations of the two presidents. The asymmetry of relation in transitive constructions involving verbs of accomplishments and achievements (or simply world-changing events) is however further complicated by a number of structural problems which are typically found in most languages of the world, the first one and most common being Passive constructions:

i.John killed Tom.
ii.Tom was killed by a man.
Who killed the man?

Answer to the question would be answered by "John" in case the information available was represented by sentence in i., but it would be answered by "Tom" in case the information available was represented by sentence ii. Obviously this would happen only in lack of sufficient NLP elaboration: a too shallow approach would not be able to capture presence of a passive structure. We are here referring to "Chunking"-based approaches those in which the object of computation is constituted by the creation of Noun Phrases and no attempt is made to compute clause-level structure.

There is a certain number of other similar structure in texts which must be regarded as inducing into the same type of miscomputation: i.e. taking the surface order of NPs as indicating the deep intended meaning. In all of the following constructions the surface subject is on the contrary the deep object thus the Affected Theme or argument that suffers the effects of the action expressed by the governing verb rather than the Agent:

**Inchoatized structures; Ergativized structures; Impersonal structures**

Other important and typical structures which constitute problematic cases for a surface chunks based TEs approach to text computation are the following ones in which one of the arguments is missing and Control should be applied by a governing NP, they are called in one definition Open Predicative structures and they are

**Relative clauses; Fronted Adjectival adjunct clauses; Infinitive clauses; Fronted Participial clauses,; Gerundive Clauses; Elliptical Clauses; Coordinate constructions**

In addition to that there is one further problem and is definable as the Factuality Prejudice: by collecting keywords and TEs people apply a Factuality Presupposition to the text they are mining: they believe that all terms being recovered by the search represent real facts. This is however not true and the problem is related to the possibility to detect in texts the presence of such semantic indicators as those listed here below:

**Negation; Quantification; Opaque contexts (wish, want); Future, Subjunctive Mode; Modality; Conditionals**

Finally, there is a discourse related problem and is the **Anaphora Resolution** problem which is the hardest to be tackled by NLP: it is a fact that anaphoric relations are the building blocks of cohesiveness and coherence in texts. Whenever an anaphoric link is missed one relation will be assigned to a wrong referring expression thus presumably jeopardising the possibility to answer a related question appropriately.

## 5. GETARUNS Complete and Robust

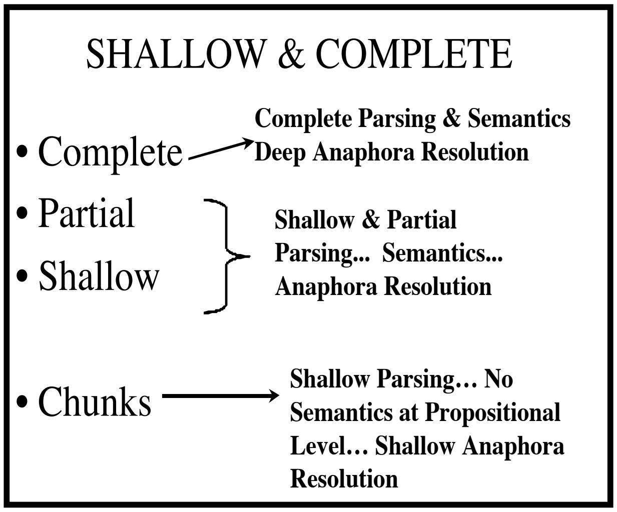
Consider now a simple sentence like the following:

- John went into a restaurant

This might be represented by TEs as follows:
<John go restaurant>
<GO <SUBJ John>, <OBLrestaurant>>

GETARUNS represents the same sentence in different manners according to whether it is operating in Complete or in Shallow-Robust or Chunks modality. In turn the operating modality is determined by its ability to compute the current text: in case of failure the system will switch automatically from Complete to Partial/Shallow modality; and in case of failure again it will switch to Chunks modality.

**Fig.1 GETARUNS system Robust and Complete**



SHALLOW & COMPLETE

• Complete → Complete Parsing & Semantics Deep Anaphora Resolution

• Partial

• Shallow

Shallow & Partial Parsing... Semantics... Anaphora Resolution

• Chunks → Shallow Parsing… No Semantics at Propositional Level… Shallow Anaphora Resolution

The system will produce the following representations:

```
loc(infon2, id1, [arg:main_tloc, arg:tr(f1_r01)])
loc(infon3, id2, [arg:main_sloc, arg:restaurant])
ind(infon4, id3)
fact(infon5, inst_of, [ind:id3, class:man], 1, univ, univ)
fact(infon6, name, [john, id3], 1, univ, univ)
ind(infon7, id4)
fact(infon8, isa, [ind:id4, class:restaurant], 1, id1, id2)
fact(infon9, inst_of, [ind:id4, class:place], 1, univ, univ)
fact(id5, go, [agent:id3, locat:id4], 1, tes(f1_r01), id2)
fact(infon12, isa, [arg:id5, arg:ev], 1, tes(f1_r01), id2)
fact(infon13, isa, [arg:id6, arg:tloc], 1, tes(f1_r01), id2)
fact(infon14, past, [arg:id6], 1, tes(f1_r01), id2)
fact(infon15, time, [arg:id5, arg:id6], 1, tes(f1_r01), id2)
```

This first representation is inspired by Situation Semantics where reality is represented in Situations which are collections of Facts: in turn facts are made up of Infons which information units characterised as follows:

```
Infon(Index,
        Relation(Property),
        List of Arguments - with Semantic Roles,
        Polarity - 1 affirmative, 0 negation,
        Temporal Location Index,
        Spatial Location Index)
```

In addition Arguments have each a semantic identifier which is unique in the Discourse Model and is used to individuate the entity uniquely. Also propositional facts have semantic identifiers assigned thus constituting second level ontological objects. They may be "quantified" over by temporal representations but also by discourse level operators, like subordinating conjunctions. Negation on the contrary is expressed in each fact. So in case of failure at the Complete level, the system will switch to Partial and the representation will be deprived of its temporal and spatial location information as follows:

```
ind(infon4, id3)
fact(infon5, inst_of, [ind:id3, class:man], 1, univ, univ)
fact(infon6, name, [john, id3], 1, univ, univ)
ind(infon7, id4)
fact(infon8, isa, [ind:id4, class:restaurant], 1, id1, id2)
fact(infon9, inst_of, [ind:id4, class:place], 1, univ, univ)
fact(id5, go, [agent:id3, locat:id4], 1, univ, id2)
```

Finally, the Shallow and the Chunks modality will limit the DM representation to entities, as follows:

```
ind(infon4, id3)
fact(infon5, inst_of, [ind:id3, class:man], 1, univ, univ)
fact(infon6, name, [john, id3], 1, univ, univ)
ind(infon7, id4)
fact(infon8, isa, [ind:id4, class:restaurant], 1, id1, id2)
fact(infon9, inst_of, [ind:id4, class:place], 1, univ, univ)
```

In addition, all system modalities will separately build a less abstract representation which is intended to capture the real words being used in the linguistic descriptions: this representation is exploited in Summarization and in the generation of canned-type answers. Suppose we have a sentence as the following one:

"This unit has been manufactured to assure your personal safety, but improper use can result in potential electrical shock or fire hazards."

which has been taken from one of the texts we have used for our Evaluation Test reported below, the representation at the concrete level is as follows: it is constituted by a set of referring expressions with an index, a semantic identifier (the same of the DM ) and a Grammatical Function,

```
refs(r0007, id5, subj, unit, [this, unit])
refs(r0008, id8, mod, electrical_shock, [electrical, shock,
electrical_shock])
refs(r0009, id4, obj, safety, [your, personal, safety])
refs(r0010, id6, subj, use, [improper, use])
```

Multiwords are decomposed and appear twice in the final representation. Then there is the list of relations,

```
refs(r0011, id7, obl, fire, [in, potential, electrical_shock, or, fire])
refs(r0012, 4-4, id6, subj, [can, result], [improper, use])
refs(r0013, 4-4, id7, obl, [can, result], [in, potential,
electrical_shock, or, fire])
refs(r0016, 4-1, id5, subj, [has, been, manufactured], [this, unit])
```

refs(r0017, 4-1, id9, vcomp, [has, been, manufactured], [to, assure, your, personal, safety])
refs(r0015, 4-2, id9, obj, [to,assure], [your,personal,safety])

## 6. The Experiment

We downloaded the only freely available corpus annotated with anaphoric relations, i.e. Wolverhampton's Manual Corpus made available by Prof. Ruslan Mitkov on his website. The corpus contains text from Manuals at the following address, http://clg.wlv.ac.uk/resources/corpus.html

| Text Type | Referring Exps | Coreferring Exps | Total Words |
|---|---|---|---|
| AIWA | 1629 | 716 | 6818 |
| ACCESS | 1862 | 513 | 9381 |
| PANASONIC | 1263 | 537 | 4829 |
| HINARI | 673 | 292 | 2878 |
| URBAN | 453 | 81 | 2222 |
| WINHELP | 672 | 206 | 2935 |
| CDROM | 1944 | 279 | 10568 |
| Totals | 8496 | 2624 | 39631 |

Table 2. General data of Worlverhampton's coreference annotated corpora

We reported in Tab. 2 the general data of the Coreference Corpus. As can be easily noted, there is no direct relationship existing between the number of referring expressions and the number of coreferring expressions. We assume that the higher the number of coreferring expressions in a text the higher is the cohesion achieved. Thus the text identified as CDROM has a very small number of coreferring expressions if compared to the total number of referring expressions. The proportion of referring expressions to words and of coreferring expressions to referring expressions is reported in percent value in table 3. where the most highly cohesive texts are highlighted in italics; highly non cohesive texts are highlighted in bold:

| Text Type | Referring Exps % W | Coreferring Exps % RE |
|---|---|---|
| AIWA | 23.89 | *43.21* |
| ACCESS | 19.84 | 27.01 |
| PANASONIC | 26.15 | *42.51* |
| HINARI | 23.38 | 29,22 |
| URBAN | 20.38 | **17.88** |
| WINHELP | 22.89 | 27.14 |
| CDROM | 18.39 | **14.24** |
| Means | 21.43 | 30.88 |

Table 3. Proportion of coreferential expressions to referring expressions

To compare our results with the SGML documents we created a Perl script that extracted all referring expressions and wrote the output into a separate file. The new representation of the SGML files looked now like a list of records each one denoted by an index a dash and the text of the referring expression. In case of complex referring expressions we had more than one index available and so we translated the complex referring expression into a couple or a triple of records each one denoted by its index.
The first comparison results were very disappointing. So we looked into the files and we found out that in general, there were mapping problems in the way in which complex referring expressions had been manually encoded. For instance, in such cases as the following ones,
Example 1.

- ASCII text

from the Royal National Institute for the Blind's helpsheets.

- SGML text

<COREF ID="1833"><COREF ID="1832">the Royal National Institute for the Blind's</COREF> helpsheets</COREF> .

- output records

1832- the Royal National Institute for the Blind-s_
1833- the Royal National Institute for the Blind-s_ helpsheets

- our output

1753 - the Royal National Institute
1754 - the Blind-s_ helpsheets
Example 2.

- ASCII text

many of the features available with the CombiBraille

- SGML text

<COREF ID="1747">many of the features available with <COREF ID="1748" TYPE="IDENT" REF="1664">the CombiBraille</COREF></COREF>

- output records

1747 - many of the features available with the CombiBraille
1664 - the CombiBraille

- our output

1666 - many of the features available
In both examples the governing referring expression is not captured because the choice of the annotators (or the program that helped in the annotation) has been dishomogeneous and misleading. Perhaps a better way to encode the internal modifying relation could have been the following one,

Example 3. SGML text
<COREF ID="1833"><COREF ID="1832">the Royal National Institute</COREF> for the Blind's helpsheets</COREF> .

Example 4. SGML text
<COREF ID="1747">many of the features available<COREF ID="1748" TYPE="IDENT" REF="1664"> with the CombiBraille</COREF></COREF>

In addition to these problems we found other more serious problems: modifiers of the nominal head were systematically left off in case they were quantifiers, as in

Example 4.

- ASCII text

Much of this document

- SGML text

Much of <COREF ID="1823" TYPE="IDENT" REF="0">this document</COREF>

Or in case a nominal head was modified by an an adjunct predicate relative clause this was treated as a single referring expression, as in

Example 5.

- ASCII text

every useful feature that I know about

- SGML text

<COREF ID="1789">every useful feature that I know about</COREF>

- our output

1702 - every useful feature

This treatment was not applied to complementizerless relative clauses predicate adjuncts, as in

Example 5.

- ASCII text

a normal terminal screen of the kind Linux supports

- SGML text

<COREF ID="481">a normal terminal screen</COREF> of the kind <COREF ID="482" TYPE="IDENT" REF="1">Linux</COREF> supports

where we see that the modifier "of the kind" is left off and the relative adjunct is thus cut off from its governing head "a normal terminal screen".

In addition to this, the annotators chose not to treat as referring expression all section numbers, money amounts numbers, numbers related to software versions. Also email addresses and webpage addresses have been left off. Since they constitute regular referring expressions in our system we decided to include them in the final count.

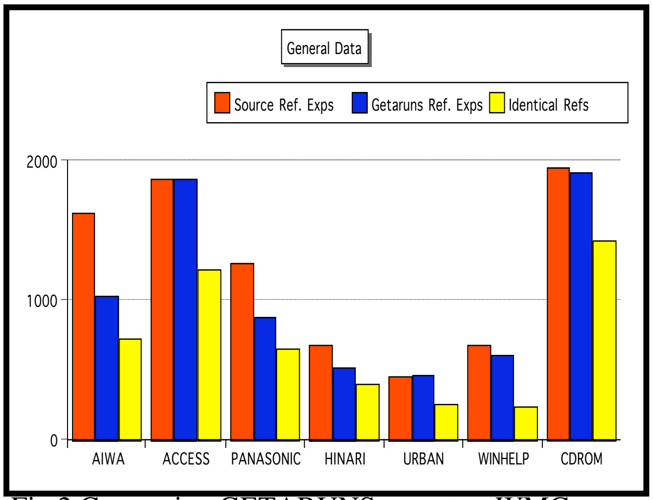


Fig.2 Comparing GETARUNS output to WMC

The final results are reported in the following figure where we plot Precision and Recall for each text and then the comprehensive values. The F measure is computed according to the following formula

$$F= \frac{(W + 1)RP}{(WR) + P}$$

where W represents the relative weight of recall to precision and typically has the value of 1.

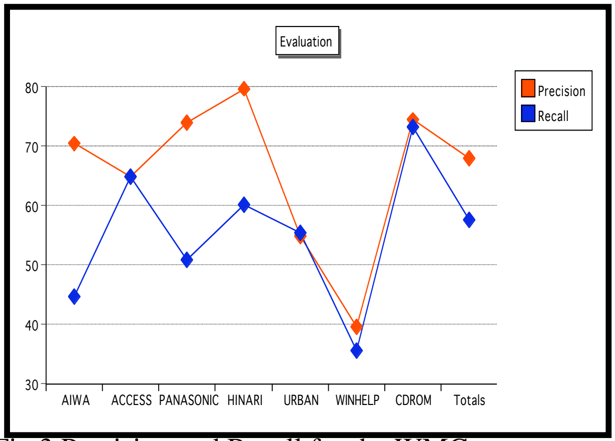


Fig.3 Precision and Recall for the WMC

## 7. Conclusions

Results reported in the experiment above have been limited to the ability of the system to cope with what has always been regarded as the toughest task for an NLP system to cope with. We have not addressed the problem of question answering for lack of space.

Would it be possible for computers the recognize the layout of a Web page, much in the same manner as a human? Much like the development of the Semantic Web itself, early efforts to integrate natural language technology with the Semantic Web will no doubt be slow and incremental. By weaving natural language into the basic fabric of the Semantic Web, we can begin to create an enormous network of knowledge easily accessible by both machines and humans alike. Furthermore, we believe that natural language querying capabilities will be a key component of any future Semantic Web system. By providing "natural" means for creating and accessing information on the Semantic Web, we can dramatically lower the barrier of entry to the Semantic Web. Natural language support gives users a whole new way of interacting with any information system, and from a knowledge engineering point of view, natural language technology divorces the majority of users from the need to understand formal ontologies. As we have tried to show in the paper, this calls for better NLP tools where a lot of effort has to be put in order to allow for complete and shallow/robust techniques to coalesce smoothly into one single system. GETARUNS represents such a hybrid system and its performance is steadily improving.

In the future we intend to address the problem of using the database of TEs created by our system in asnswering a more extended set of natural language queries than what has been tried sofar [8].

## 7. References

[1] Berners-Lee, T., Hendler, J., and Lassila, O. The Semantic Web. Scientific American (May 2001).

[2] Lassila, O. and Swick, R. (eds.). Resource Description Framework (RDF) model and syntax specification. Available at http://www.w3.org/TR/1999/REC-rdf-syntax-19990222.

[3] Bray, T., Hollander, D., and Layman, A. (eds.) Namespaces in XML. Available at http://www.w3.org/TR/REC-xmlnames/.

[4] Kahan, J. and Koivunen, M. Annotea: an open RDF infrastructure for shared web annotations, in Proceedings of WWW10 (May 2001).

[5] Handschuh, S., Staab, S., and Maedche, A. CREAM—creating relational metadata with a component-based ontology-driven annotation framework, in Proceedings of K-CAP 2001 (October 2001).

[6] David R. Karger, Boris Katz, Jimmy Lin, Dennis Quan, Sticky Notes for the Semantic Web, IUI'03, January 12–15, 2003, Miami, Florida, USA, ACM 1-58113-586-6/03/0001.

[7] Boris Katz Jimmy J. Lin Sue Felshin, The START Multimedia Information System: Current Technology and Future Directions, In Proceedings of the International Workshop on Multimedia Information Systems (MIS 2002).

[8] R.Delmonte, 2003. Getaruns: a Hybrid System for Summarization and Question Answering. In Proc. Natural Language Processing (NLP) for Question-Answering, EACL, Budapest.

[9] J. Lin. 2001. Indexing and retrieving natural language using ternary expressions. Master's thesis, Massachusetts Institute of Technology.

[10] B. Katz. 1997. Annotating the World Wide Web using natural language. In RIAO '97.

[11] Delmonte R. 2000d. Generating from a Discourse Model, *Proc. MT-2000*, BCS, Exeter, pp.25-1/10.

[12] Delmonte R., D. Bianchi. 2002. From Deep to Partial Understanding with GETARUNS, *Proc. ROMAND 2002*, Università Roma2, Roma, pp.57-71.

[13] Litkowski, K. C. (2001). Syntactic Clues and Lexical Resources in Question-Answering. In E. M. Voorhees & D. K. Harman (eds.), The Ninth Text Retrieval Conference (TREC-9). NIST Special Publication 500-249. Gaithersburg, MD., 157-166.

[14] Litkowski, K. C. (2002a). CL Research Experiments in TREC-10 Question-Answering. In E. M. Voorhees & D. K. Harman (eds.), The Tenth Text Retrieval Conference (TREC 2001). NIST Special Publication 500-250. Gaithersburg, MD., 122-131.

[15] Litkowski, K. C. (2002b). Digraph Analysis of Dictionary Preposition Definitions. Proceedings of the ACL SIGLEX Workshop: Word Sense Disambiguation. Philadelphia, PA., 9-16.

[16] Ravichandran, D. & E. Hovy. (2002). Learning Surface Text Patterns for a Question Answering System. Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia, PA., 41-7.

[17] Harabagiu, S., Moldovan, D., Pasca, M., Surdeanu, M., Mihalcea, R., Girju, R., Rus, V., Lacatusu, F., Morarescu, P., & Bunescu, R. (2002). Answering complex, list, and context questions with LCCs Question-Answering Server. In TREC-10 Question-Answering. In E. M. Voorhees & D. K. Harman (eds.), The Tenth Text Retrieval Conference (TREC 2001). NIST Special Publication 500-250. Gaithersburg, MD., 355-361.

[18] Hovy, E., U. Hermjakob, & C. Lin. (2002a). The Use of External Knowledge in Factoid QA. In E. M. Voorhees & D. K. Harman (eds.), The Tenth Text Retrieval Conference (TREC 2001). NIST Special Publication 500-250. Gaithersburg, MD., 644-652.

# Coreference-Based Summarization and Question Answering: a Case for High Precision Anaphor Resolution

**Roland Stuckardt**
Johann Wolfgang Goethe University Frankfurt
Im Mellsig 25
D-60433 Frankfurt am Main, Germany
roland@stuckardt.de

## Abstract

Approaches to Text Summarization and Question Answering are known to benefit from the availability of coreference information. Based on an analysis of its contributions, a more detailed look at coreference processing for these applications will be proposed: it should be considered as a task of anaphor resolution rather than coreference resolution. It will be further argued that high precision approaches to anaphor resolution optimally match the specific requirements. Three such approaches will be described and empirically evaluated, and the implications for Text Summarization and Question Answering will be discussed.

## 1 Introduction

Text Summarization (TS) and Question Answering (QA) are generic applications that highly benefit from the availability of an enhanced software technology for the content-oriented analysis of potentially noisy textual data. Recent research has shown that these applications particularly profit from the availability of robust, knowledge-poor solutions to the coreference resolution task as defined at the Message Understanding Conferences MUC-6 and MUC-7.[1] Various research projects have investigated how coreference information can be employed to determine the topics of a text (as relevant for TS), or to determine the contexts that contribute potentially relevant information about entities mentioned in a user query (as relevant for QA).[2]

Beside the work on coreference resolution as fostered by the MUCs, a line of related research addresses the problem of anaphor resolution.[3] These problems are in so far closely related as solutions to the second-mentioned task contribute to performing the first-mentioned task. Importantly, however, they differ with respect to the perspective from which the coreference processing issue is discussed, which is reflected by the different methods that are employed to evaluate software technology for the two tasks: regarding coreference resolution, the scoring procedure refers to the *classes* of coreferring linguistic expressions, whereas the anaphor resolution output is typically assessed by determining the accuracy with which the *antecedent selection* for certain types of anaphoric expressions is performed.

In this paper, the role of coreference information and coreference processing for TS and QA will be explored. Beginning with a brief survey of recent work, the potential contributions of coreference information to QA and TS are identified. Based on the identification of different ways of employing coreference information for these applications, it will be studied in detail which kind of coreference processing is needed, and which requirements an algorithm should meet in order to optimally support the solution of these tasks. This involves a theoretical analysis as well as a look at empirical data. According to

[1] cf. Hirschman (1998)

[2] e.g., cf. Baldwin and Morton (1998), Azzam, Humphreys, and Gaizauskas (1999), Breck et al (1999), Morton (1999)

[3] cf. the monograph of Mitkov (2002)

the results of these investigations, coreference processing for TS and QA should be looked at in more detail: it should be considered as a task of anaphor resolution rather than coreference resolution. Moreover, approaches to anaphor resolution should be biased towards high precision in order to match the specific requirements. Three such approaches will be described and empirically evaluated, and the implications for TS and QA will be discussed.

## 2 Coreference Information for Text Summarization and Question Answering

Various research projects have explored coreference-based approaches to TS and QA. A brief survey of some representative approaches will be given.

### 2.1 Text Summarization

Baldwin and Morton (1998) investigated coreference-based TS in an information retrieval (IR) scenario in which automatically generated document summaries are used to support relevance judgments of the IR user. Basically, coreference analysis is employed in two processing stages: (1) retrieving referential relations between the terms of the original IR query and the terms of the documents that are considered, with respect to the query, to be of highest relevance by the IR engine; (2) generation of the document summary by identifying the sentences in which entities of the query that have been identified at stage (1) occur. For solving the second-mentioned problem, the system follows the coreference chains and heuristically selects a subsequence of sentences that, according to further criteria, are judged to be of highest relevance; at this stage, the approach further takes care to provide lexically informative substitute expressions for anaphors (in particular pronouns) that may, out of their original context, become incomprehensible.[4]

The approach of Azzam, Humphreys, and Gaizauskas (1999), too, employs coreference resolution for deriving text summaries. They considered the scenario of *generic* summarization, in which there is no user query that prescribes relevant entities on which the summary should focus. Their algorithm tries to identify a *single* coreference chain pertaining to the central entity the text is about. They further investigated the contribution of a focus mechanism to identify the subsequence of sentences in which this entity is salient.[5]

### 2.2 Question Answering

QA, too, benefits from the availability of coreference information since it renders possible the identification of contexts in which information regarding the entities a question is about is contributed. A formal definition of a QA scenario has been provided and investigated at the TREC evaluation conferences. According to Breck et al (1999) and Morton (1999), whose systems participated at TREC-8, this problem, too, can be solved by employing coreference information in the two stages of (1) relating entities mentioned in the query to the retrieved documents, and (2) looking at the relevant coreference classes and searching the contexts in which these entities occur for information that may contribute to answer the question.[6] As regarding TS, the coreference information is further employed to supply maximally informative substitutes for anaphoric realizations of entities mentioned in contexts that contribute to answering the question.

### 2.3 Contribution of Coreference to TS and QA

According to the above survey of some representative approaches, the tasks of TS and QA are closely related. Coreference information is employed in different processing stages. For QA and user-focused TS, the first stage consists in relating some terms of the query to coreferring occurrences in the document pool over which the application runs; this may be considered as a special case of the cross-document coreference resolution problem that has been investigated elsewhere.[7] At the second processing stage, three different cases of using exclusively document-local coreference information may be distinguished:

[4] According to Baldwin and Morton (1998), their approach deals with object coreference and event coreference. They further consider the issue of referential relations beyond the identity relation; this, however, merely seems to cover a few domain-specific special cases.

[5] A plethora of further approaches to automatic text summarization has been investigated (cf., e.g., (Mani, 2002)). Recent research has in particular been fostered by the TIPSTER SUMMAC evaluation exercise, cf. (Mani et al., 1998).

[6] An analysis of the type of the expected answer typically supports this process.

[7] cf., e.g., (Bagga and Baldwin, 1998; Ravin and Kazi, 1999)

1. looking at the coreference *classes* of relevant entities in order to retrieve contexts that contribute information potentially relevant for QA;
2. following a coreference *chain* in order to select a sub*sequence* of sentences that constitute, or contribute to, a document summary;
3. identifying coreferring *antecedents for anaphoric occurrences* in order to provide maximally informative substitute expressions (for QA as well as TS).

According to case 1, solutions to the QA task refer to unordered classes, i.e. sets of coreferring occurrences. This seems to indicate that document-local coreference resolution for TS and QA should be addressed by an approach with high empirical performance in the coreference task as formally defined for MUC-6 and MUC-7.[8] The other two cases, however, emphasize asymmetric aspects of coreference, viz. (surface-topologically ordered) chains of coreferring occurrences, or certain types of anaphoric expressions to be substituted by non-anaphoric antecedent expressions. Moreover, as illustrated by figure 1, TS and QA typically employ lexical information in order to identify relevant coreference classes: *lexically informative* occurrences are the typical points of access. This indicates that looking at the cofererence class level only as done by the MUC scoring scheme of Vilain et al. (1996) falls short of capturing certain aspects that are crucial with regard to the applications TS and QA.
It will now be shown that these requirements can be complied with by considering the document-local coreference processing task as a problem of anaphor resolution rather than reference resolution.

## 3 Coreference processing for TS and QA

### 3.1 Towards anaphor resolution

Choosing a coreference processing module that optimally supports TS and QA requires appropriate evaluation measures that are expressive with respect to the type of performance that, according to section 2.3, is essential for these applications. To discuss this issue, an example that illustrates how coreference resolution errors are counted by different evaluation measures proves to be helpful. Figure 2 shows the typical case of a coreference processing error as generated by an employed anaphor resolution system.[9] The coreference classes specified by the key are represented by the dashed boxes. The anaphor resolution system response consists of a set of instances of anaphoric resumption that are represented by arrows pointing from the anaphor to the resumed antecedent; the respective response coreference classes are obtained by computing the reflexive-transitive closure over the individual resumptions and determining the equivalence classes of it. In the configuration shown in figure 2, to an occurrence *he* that belongs to the key coreference class *[Don Giovanni]* an incorrect antecedent has been assigned, which belongs to key class *[Leporello]*.

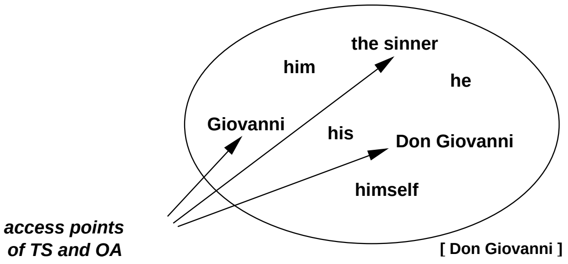


Figure 1: accessing a coreference class via lexically informative occurrences

If the task to be accomplished is considered a problem of coreference resolution the goal of which consists in the computation of the *classes* of coreferring occurrences, the model-theoretic scoring scheme of Vilain et al. (1996) may be employed. According to this scheme, in the configuration of figure 2, there is a single *recall error*, since there is one subclass of the *[Don Giovanni]* key class (represented by the dotted box) that is not connected to the rest of the key class. There is also one single *precision error*: one response class contains a subclass (again, the occurrences in the dotted box) that, according to the key, should have been kept apart from the other occurrences, which belong to class *[Leporello]*.
This scheme, however, does not take into account the crucial issues that have been pointed out in

[8] cf. (Hirschman, 1998) and the respective coreference-class-oriented scoring scheme of Vilain et al. (1996)

[9] As stated above, it can be assumed that even approaches to the coreference resolution task refer to the output of an anaphor resolution stage in order to compute the coreference classes.

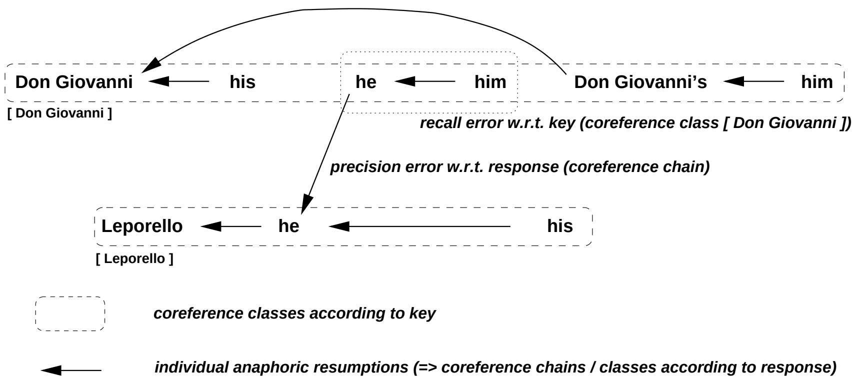

Don Giovanni
his
he
him
Don Giovanni's
him
[ Don Giovanni ]
recall error w.r.t. key (coreference class [ Don Giovanni ])
precision error w.r.t. response (coreference chain)
Leporello
he
his
[ Leporello ]
coreference classes according to key
individual anaphoric resumptions (=> coreference chains / classes according to response)

Figure 2: coreference processing error from the perspectives of anaphor vs. coreference resolution

section 2.3. Essentially, it considers the following two cases as equal (the arrows represent antecedent choices, "$+$" denotes "*correct*", "$-$" stands for "*wrong*"):

$$\textit{(1) Leporello} \xleftarrow{-} \textit{he} \xleftarrow{+} \textit{him} \xleftarrow{+} \textit{his}$$
$$\textit{(2) Leporello} \xleftarrow{+} \textit{he} \xleftarrow{+} \textit{him} \xleftarrow{-} \textit{his}$$

Regarding TS and QA, however, case (1) should be considered worse than case (2). As discussed above and illustrated in figure 1, TS and QA typically access coreference classes or chains via occurrences that are lexically informative. Hence, for these applications, in case (1), only one out of four coreferring occurrences would be found, whereas in case (2), three out of four coreferring occurrences would be retrieved. A similar argument holds with respect to the subtask of providing maximally informative substitute expressions for anaphors in the output of TS and QA. Things are even worse since there is focus-theoretic as well as empirical evidence that the problem of identifying a content-carrying non-pronominal antecedent is considerably harder than identifying an arbitrary antecedent.[10] This implies that the model-theoretic scoring scheme (Vilain et al., 1996) yields results that are, in general, not sufficiently expressive with respect to the contribution of coreference resolution to TS and QA.

The refinement of model-theoretic coreference scoring that is suggested by Bagga and Baldwin (1998) (B-CUBED scoring algorithm) weights errors by taking into account the number of occurrences of the affected class and the relative sizes of the induced subclasses. It meets the specific requirements of evaluating cross-document coreference resolution systems, whereas it does not comply with the above identified requirements.

To achieve the required sensitivity, the problem should be looked at in more detail. Coreference processing for TS and QA should be considered as a task of anaphor resolution rather than coreference resolution, and one should depart from evaluation schemes merely grounded on coreference classes. Formal measures for the evaluation of anaphor resolution systems should be employed. Let $(\alpha, \gamma)$ be a pair consisting of an anaphoric occurrence $\alpha$ and an antecedent occurrence $\gamma$ determined by the anaphor resolution system. (If, for $\alpha$, no antecedent has been determined, then $\gamma$ is empty.) The scoring is based on a disjoint partition of the pairs $(\alpha, \gamma)$ output by the anaphor resolution system into the following sets: $o_{++}$ ($\alpha$ and $\gamma$ corefer), $o_{+-}$ ($\alpha$ and $\gamma$ do not corefer), $o_{+\_}$ ($\gamma$ empty, no antecedent assigned), $o_{+?}$ ($\gamma$ denotes a spurius occurrence). By distinguishing between measures of precision and recall, the second-mentioned of which takes into account the cases with empty antecedent $\gamma$ as well, one obtains the definitions:[11]

$$P := \frac{|o_{++}|}{|o_{++}|+|o_{+-}|+|o_{+?}|}$$
$$R := \frac{|o_{++}|}{|o_{++}|+|o_{+-}|+|o_{+?}|+|o_{+\_}|}$$

These measures will be employed to determine the performance regarding the identification of arbitrary coreferring antecedents (immediate antecedency *(ia)* discipline). Now, essentially, to take

[10] Pronouns typically refer to focused entities. Hence, they are, with higher probability, correct antecedents (cf. (Stuckardt, 2001)).

[11] For further details cf. (Stuckardt, 2001).

into account the central issue of anchoring pronouns in information-carrying occurrences, which constitute the potential access points of TS and QA, the evaluation discipline of *non-pronominal anchoring (na)* will be considered. It employs the same P and R measures; however, only pronominal anaphors $\alpha$ are taken into consideration, and antecedents $\gamma$ are required to be lexically informative.[12]

It has to be emphasized that it is not proposed to reduce coreference processing for TS and QA to the mere task of providing lexically informative substitutes for pronouns or to pronominal anaphor resolution. Obviously, *general* coreference information has to be provided in order to comply with the requirements of TS and QA identified in section 2.3. According to the above analysis, however, technology assessment by model-theoretic coreference scoring falls short of adequately capturing the important case of lexically anchoring pronouns, which is a harder problem than coreference class determination according to the MUC task.[13] Moreover, the provision of lexically informative substitutes for pronouns is known to significantly enhance the performance of QA systems, as has been shown by the empirical investigation of Vicedo and Ferrández (2000a).[14] Clearly, the actual contribution of pronominal anaphor resolution depends on the density of pronoun occurrences relevant to the specific QA task; as shown by Vicedo and Ferrández (2000b), it may be moderate in certain cases.

### 3.2 The case for high prec anaphor resolution

By further reflecting upon how, according to section 2.3, TS and QA employ referential information, additional evidence regarding the specific anaphor resolution strategy that optimally supports these applications can be obtained.

With respect to the coreference chains sought by TS, *recall errors* can be expected to affect the quality of the summarization output only weakly since these errors tend to have local impact only. Typically, regarding a particular coreference chain, as illustrated by figure 2, there will be local sequences of *pronouns* that, due to single anaphor resolution errors, are not connected to the chain. However, subsequent occurrences that carry more referentially discriminative information will be correctly resolved; thus, the interrupted chain will be resumed. This is further supported by empirical data that will be presented below. Since the spread of the chain can thus be expected to still cover the whole document, and since the summary is typically constructed by selecting a *subsequence* of occurrences, the loss should not be too big. *Precision errors*, on the other hand, potentially affect the output quality: if occurrences are erroneously identified as coreferring, the summary may contain irrelevant sentences; the reader may be further misled if incorrect substitute expressions for pronouns are provided.

Regarding the coreference classes sought by QA, *recall errors* do have potential impact since information contributed by a context of a not-found coreferring occurrence gets lost. However, the document pool over which QA is performed may exhibit redundancy and the sought information may be retrieved from elsewhere; in fact, this has been empirically observed by Vicedo and Ferrández (2000b) during the TREC-9 evaluation. *Precision errors* are critical since they can lead to a wrong answer derived from a context of a non-coreferring occurrence, including an incorrect substitute expression.

Thus, precision errors can be expected to cause more potential damage with respect to the applications TS and QA than recall errors. Hence, strategies to high precision anaphor resolution shall be explored.

## 4 Three approaches to robust high precision anaphor resolution

In order to see which level of performance can be reached, three approaches to high precision anaphor resolution will be investigated. The subsequent discussion focuses on the subproblem of third-person pronominal anaphora, the interpretation of which is known to be of particular importance to TS and QA

[12] This proposal can be rendered even more generally: the performance with respect to relating *lexically less informative* (typically anaphoric) occurrences to *lexically more informative* occurrences should be measured. Pronouns are the most important and easy-to-recognize special case: they are lexically less informative than any common NP or name occurrence.

[13] A detailed explanation is given in (Stuckardt, 2001).

[14] Vicedo and Ferrández (2000a) focus on the case of QA by text snippet (viz., sentence) extraction. Corresponding to the two different ways of employing document-local coreference information for QA identified in section 2.3, they prove the positive effects of substituting pronouns that refer to entities (1) mentioned in the query, and (2) to be mentioned in the answer.

(cf., e.g., (Vicedo and Ferrández, 2000a)).[15] The approaches should work robustly on texts of arbitrary domains, i.e. under the side condition of knowledge-poor processing of potentially noisy data. The robust syntactic salience-based anaphor resolution system ROSANA and its machine-learning-based descendant ROSANA-ML of Stuckardt (2001; 2002) are taken as the starting points.[16]

### 4.1 ROSANA with CogNIAC high prec ruleset

The first approach consists in the partial reimplementation of the CogNIAC system of Baldwin (1997), which is designed to achieve high precision pronoun resolution. CogNIAC combines the morphological agreement and syntactic disjoint reference filters with six antecedent selection rules, each of which covers one specific situation in which there seems to be little or no ambiguity regarding the antecedent choice. The antecedent filters are employed prior to the six high precision rules, which are applied in order of increasing ambiguity: if a rule applies, the respective candidate will be chosen; if no rule applies, the anaphor remains unresolved.

The CogNIAC system of Baldwin (1997) requires full parses, whereas its reimplementation ROSANA-CogNIAC, which combines the *robust* antecedent filters of ROSANA with the high precision ruleset of CogNIAC, works on partial parses, and, hence, meets the robustness requirements.

### 4.2 ROSANA with salience threshold

A second approach to high precision pronoun resolution consists in an even more immediate adaptation of the antecedent selection phase of classical, salience-based anaphor resolution algorithms:

> *Given a salience threshold $\theta$, only such candidates are considered the salience of which exceeds the threshold $\theta$.*

The rationale behind this strategy is that salience does not only constitute a base for heuristically comparing the relative plausibility of the candidates (and choosing the one with highest salience); in addition, it can be employed as an heuristic estimate of the probability that an individual candidate is a correct antecedent, thus allowing to decline candidates with low salience in order to avoid risky decisions.

By accordingly modifying the antecedent selection step of ROSANA, the system ROSANA-$\theta$ is obtained.

### 4.3 ROSANA-ML towards high precision

Another approach to high precision pronoun resolution has been investigated as part of the research on the machine-learning-based approach ROSANA-ML, which employs C4.5 decision tree classifiers for selecting among antecedent candidates fulfilling the filtering criteria.[17] Basically, the decision trees represent classifier functions which map pairs of anaphors and antecedent candidates (represented as feature vectors) to a prediction $\in \{COREF, NON\_COREF\}$. Beside the primary classification result, the leaves of the decision trees contain additional quantitative information: each leaf provides the total number $\mu$ of training cases that match the respective decision path, and the number $\varepsilon \leq \mu$ of these cases that are, through the category prediction of the leaf, wrongly classified. By computing the quotient $\frac{\varepsilon}{\mu}$, it should thus be possible to derive an estimate of the classification error probability of the particular leaf.

This information can be employed to gradually bias ROSANA-ML towards high precision. For this end, the preference criterion of ROSANA-ML, which refers to the (heuristical) decision tree predictions and employs surface-topological distance as the secondary criterion, has been modified by adding a threshold $\theta$ that imposes bounds on the admissible classification error probability estimates $\frac{\varepsilon}{\mu}$.[18] This yields the system ROSANA-ML-$\theta$, the degree of inclination towards precision of which depends on $\theta$.

### 4.4 Evaluation

Figure 3 displays evaluation results of the above approaches on a corpus of 35 news agency press

[15] As argued above, since general coreference information is required, pronoun resolution has to be supplemented by strategies dealing with other types of referring expressions, in particular names and common NPs.

[16] ROSANA and ROSANA-ML interpret names and definite NPs as well, and perform general coreference resolution. Only the discussion focuses on pronoun resolution issues.

[17] ROSANA-ML has been trained and thoroughly evaluated (including intrinsic 10-fold and extrinsic 6-fold cross-validation of the learned classifiers) on a corpus of 66 press releases (cf. section 4.4). Full details are given in (Stuckardt, 2002).

[18] Details are given in (Stuckardt, 2002).

|  | antecedents $(P_{ia}, R_{ia})$ |  | anchors $(P_{na}, R_{na})$ |  |
|---|---|---|---|---|
| experiment | PER3 | POS3 | PER3 | POS3 |
| (0) ROSANA (salience-based) | $(0.71, 0.71)$ | $(0.76, 0.76)$ | $(0.68, 0.67)$ | $(0.66, 0.66)$ |
| (1) ROSANA-CogNIAC | $(0.66, 0.49)$ | $(0.82, 0.53)$ | $(0.62, 0.42)$ | $(0.79, 0.45)$ |
| (2) ROSANA-CogNIAC, (R6)' | $(0.74, 0.59)$ | $(0.82, 0.53)$ | $(0.71, 0.53)$ | $(0.77, 0.45)$ |
| (3) ROSANA-$\theta$ ($\theta$ = 90) | $(0.75, 0.67)$ | $(0.79, 0.74)$ | $(0.74, 0.62)$ | $(0.72, 0.63)$ |
| (4) ROSANA-$\theta$ ($\theta$ = 110) | $(0.79, 0.62)$ | $(0.81, 0.50)$ | $(0.77, 0.56)$ | $(0.74, 0.38)$ |
| (5) ROSANA-ML-$\theta$, $p$ | $(0.79, 0.51)$ | $(0.86, 0.60)$ | $(0.75, 0.45)$ | $(0.83, 0.54)$ |
| (6) ROSANA-ML-$\theta$, $p^-$ | $(0.74, 0.56)$ | $(0.78, 0.63)$ | $(0.71, 0.52)$ | $(0.76, 0.59)$ |
| (7) ROSANA-ML-$\theta$, $p^+$ | $(0.81, 0.45)$ | $(0.89, 0.50)$ | $(0.74, 0.36)$ | $(0.67, 0.30)$ |
| (8) ROSANA-ML-$\theta$, $p^{++}$ | $(0.83, 0.31)$ | $(1.00, 0.17)$ | $(0.80, 0.08)$ | $(1.00, 0.12)$ |

Figure 3: evaluation results of the high precision approaches on a corpus of *News Agency Press Releases*

releases, comprising 12,904 words, 204 third-person non-possessives, and 131 third-person possessives.[19] In row (0), the results of the original version of ROSANA are shown. Arbitrary immediate antecedents (*ia* task) are chosen with an accuracy (P=R) of $0.71$ for non-possessives (PER3), and with an accuracy (P=R) of $0.76$ for possessives (POS3). If the more difficult *na* task of identifying nonpronominal antecedents is considered, results deteriorate to $(0.68, 0.67)$ and $(0.66, 0.66)$, respectively. This provides empirical support for the argument of section 3.1 according to which the problem of identifying information-carrying antecedents is considerably harder than the problem of identifying an arbitrary coreferring antecedent.

The subsequent rows display results for the three high precision approaches. ROSANA-CogNIAC's scores are given in rows (1) and (2). Two versions of ROSANA-CogNIAC are considered since it turned out that one of the original rules of CogNIAC (rule 6, dealing with intersentential subject preference) should be modified in order to achieve better results on the Press Releases texts.[20] The cumulated performance with respect to the *ia* discipline (covering third-person non-possessive, possessive, and relative pronouns)[21] amounts to 0.78 precision at 0.60 recall. It thus lags behind original CogNIAC's performance of 0.92 precision at 0.64 recall, which was determined by Baldwin (1997). This might be attributed to the harder conditions of robust processing under which ROSANA-CogNIAC has been run.

ROSANA-$\theta$ has been evaluated with a lower and a higher salience threshold. According to the results in rows (3) and (4), precision biasing works; it results in a higher precision at the expense of a lower recall when employing the higher threshold. The same holds with respect to the precision biasing strategy of ROSANA-ML-$\theta$; results for four different threshold settings are displayed in rows (5) to (8).

Obviously, there is no unambiguous winner. Which strategy performs best depends on the targeted (P,R) tradeoff level and on the type of pronoun. E.g., regarding nonpossessives, (4) ROSANA-$\theta$ ($\theta$ = 110) can be considered to be superior to (2) ROSANA-CogNIAC, (R6)'; regarding possessives, however, empirical evidence is to the contrary. ROSANA-ML-$\theta$ (particularly, the $p$ setting) seems to produce the best results on possessives.

According to figure 3, there is a strong correlation between the results in the *ia* discipline and the results in the *na* discipline. Approaches that score high in the first-mentioned discipline typically score high, too, in the second-mentioned discipline. (5) ROSANA-ML-$\theta$, $p$, e.g., achieves a nonpronominal anchoring performance of $0.83$ precision at $0.54$ recall which is considerably higher than the figures for (2) ROSANA-CogNIAC, (R6)', which amount to $(0.77, 0.45)$.[22]

ROSANA-CogNIAC and ROSANA-$\theta$ have been further evaluated on a corpus of a different genre (plot descriptions of Mozart Operas).[23] This has

[19] For the system development, a separate training corpus of 31 press releases (11,808 words, 202 non-possessives, 115 possessives) has been employed.

[20] The *previous sentence* notion was slightly weakened to cover *intra*sentential candidates that occur in a previous *clause*.

[21] Relative pronouns are not covered by the results shown in figure 3. They are included here for comparison purposes since they are covered, too, by the performance figures of CogNIAC.

[22] Interestingly, it is even higher than the immediate antecedency figures of the second-mentioned approach.

[23] Evaluation figures for these experiments are not included. Since the corpus is quite small, it proved to be impossible to evaluate ROSANA-ML-$\theta$ on it, since this approach requires a reasonable amount of training data.

given evidence that the relative performance of the approaches varies across text genres. Regarding nonpossessives, ROSANA-$\theta$ is no longer superior to ROSANA-CogNIAC. Moreover, ROSANA-CogNIAC with the original version of rule 6 now clearly outperforms ROSANA-CogNIAC, (R6)'. This might be due to the resemblance of the genre of the Mozart Operas corpus to the genre of the texts on which the original CogNIAC system was run, which were stories about two persons of different gender.

### 4.5 Implications for TS and QA

According to section 3.1, since lexically informative occurrences constitute the access points of TS and QA to coreference chains and classes, the discipline of nonpronominal anchoring will be considered here. As displayed in figure 3, regarding nonpossessive pronouns, one can achieve a precision of $0.77$ at a recall rate of $0.56$ ((4) ROSANA-$\theta$ ($\theta$ = 110)); compared to the non-biased system ((0) ROSANA (salience-based)), this amounts to a gain of 9% precision at the expense of 11% recall. Regarding possessives, by employing the approach (5) ROSANA-ML-$\theta, p$, a precision of $0.83$ at $0.54$ recall is reached, which means a gain of 17% precision at the expense of 12% recall.

Concerning TS and QA, this implies that one could expect to reduce the amount of wrongly anchored pronouns from about 33% to about 20% while still retrieving more than 50% of the pronoun occurrences. A further in-depth study has provided empirical support for a specific argument of section 3.2: while high precision anaphor resolution strategies provide an effective means to avoid wrongly anchored subchains of pronouns, they typically do not affect the overall spread of a coference chain. Specifically, the outputs of the non-biased system (0) ROSANA (salience-based) and the high precision approach (1) ROSANA-CogNIAC on the Mozart Operas corpus have been compared. The analysis shows that the spread of the two systems' result coreference chains with respect to the 5 biggest coreference classes[24] of each text is nearly identical.[25] A study of the coreference classes generated by ROSANA-CogNIAC reveals that incorrect antecedent choices are avoided in case of 12 third-person pronouns, which, due to chaining effects as described above, results in a total of 25 third-person pronouns that are no longer anchored to an incorrect lexically informative antecedent. Hence, the high precision strategy can be expected to enhance the quality of the TS output.

Regarding QA, as indicated by the empirical results by Vicedo and Ferrández (2000b), much depends on the document pool over which the application runs (cf. section 3.2). If it exhibits redundancy, it may be reasonable to employ an anaphor resolution strategy with a high degree of inclination towards precision; otherwise, a lower precision bias may yield best results. QA thus seems to be best supported by the threshold-based approaches, which render possible different degrees of biasing. This issue should be studied further by performing respective extrinsic (application-level) evaluation runs.

## 5 Conclusion and further research

Because of the specific requirements, coreference processing for TS and QA should be looked at in more detail: it should be considered as a task of anaphor resolution rather than coreference resolution. To support the choice of the most appropriate approach, formal evaluation should employ anaphor resolution evaluation measures; in particular, the performance regarding the determination of lexically informative anchors for pronouns should be assessed. In order to optimally contribute to TS and QA, solutions to anaphor resolution should be inclined towards high precision. Three approaches have been investigated. According to formal evaluation, these approaches successfully reduce the amount of wrongly anchored pronouns, while still yielding coreference chains that spread the document as required by TS. QA is expected to benefit from threshold-based approaches, which render possible different degrees of precision bias.

Further research should address the contribution of high precision anaphor resolution at the application (TS, QA) level; with respect to QA, this amounts to continuing the empirical work of Vicedo and

[24] as marked up in an intellectually gathered key

[25] There is a single case in which ROSANA-CogNIAC performs worse, which, however, proved to be not attributable to to the high precision strategy proper. Interestingly, in another case, due to complex processing interdependencies, ROSANA-CogNIAC generated a coreference chain with higher coverage.

Ferrández (2000a), who do not provide a detailed analysis of the impact of pronoun interpretation errors. Regarding the high precision strategies, the issue of genre dependency should be paid attention to. Moreover, the contributions of high precision strategies to sequenced models of anaphor resolution, which employ a series of competence modules of increasing complexity, should be investigated.

## Acknowledgements

Thanks to the anonymous reviewer number two for providing a valuable hint at the empirical investigations by Vicedo and Ferrández (2000a; 2000b) on the importance of pronominal anaphor resolution to QA.

## References

Saliha Azzam, Kevin Humphreys, and Robert Gaizauskas. 1999. Using coreference chains for text summarization. In *Proceedings of the ACL'99 Workshop on Conference and its Applications, Baltimore*, pages 77–84.

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreference using the vector space model. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING'98/ACL'98), Montreal*, pages 79–85.

Breck Baldwin and Thomas S. Morton. 1998. Dynamic coreference-based summarization. In *Proceedings of the 3rd International Conference on Empirical Methods in Natural Language Processing (EMNLP-3), Granada*, pages 1–6.

Breck Baldwin. 1997. Cogniac: High precision coreference with limited knowledge and linguistic resources. In Ruslan Mitkov and Branimir Boguraev, editors, *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphor Resolution for Unrestricted Texts, Madrid*, pages 38–45.

Eric Breck, John Burger, Lisa Ferro, David House, Marc Light, and Inderjeet Mani. 1999. A sys called qanda. In *Proceedings of the 8th Text Retrieval Conference (TREC-8), Gaithersburgh*, pages 499–506.

Lynette Hirschman. 1998. Muc-7 coreference task definition, version 3.0. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Published online, available (June 7, 2003) at http://www.itl.nist.gov/iaui/894.02/.

Inderjeet Mani, David House, Gary Klein, Lynette Hirschman, Leo Obrst, Therese Firmin, Michael Chrzanowski, and Beth Sundheim. 1998. The tipster summac text summarization evaluation, final report. Technical Report MTR 98W0000138, MITRE.

Inderjeet Mani. 2002. *Automatic Summarization*. John Benjamins, Amsterdam/Philadelphia.

Ruslan Mitkov. 2002. *Anaphora Resolution*. Longman, London.

Thomas S. Morton. 1999. Using coreference in question answering. In *Proceedings of the ACL'99 Workshop on Conference and its Applications, Baltimore*, pages 85–89.

Yael Ravin and Zunaid Kazi. 1999. Is hillary rodham clinton the president? disambiguating names across documents. In *Proceedings of the ACL'99 Workshop on Conference and its Applications, Baltimore*.

Roland Stuckardt. 2001. Design and enhanced evaluation of a robust anaphor resolution algorithm. *Computational Linguistics*, 27(4):479–506.

Roland Stuckardt. 2002. Machine-learning-based vs. manually designed approaches to anaphor resolution: the best of two worlds. In *Proceedings of the 4th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 02)*, pages 211–216.

José L. Vicedo and Antonio Ferrández. 2000a. Importance of pronominal anaphora resolution in question answering systems. In *Proceedings of the 38th Annual Meeting of the Association of Computational Lingustics (ACL'00), Hongkong*, pages 555–562.

José L. Vicedo and Antonio Ferrández. 2000b. A semantic approach to question answering systems. In *Proceedings of the 9th Text Retrieval Conference (TREC-9), Gaithersburgh*, pages 511–516.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1996. A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 45–52. Morgan Kaufmann.

# Fuzzy Coreference Resolution for Summarization

**René Witte** and **Sabine Bergler**
Department of Computer Science
Concordia University
Montréal, Canada
[rwitte|bergler]@cs.concordia.ca

## Abstract

We present a fuzzy-theory based approach to coreference resolution and its application to text summarization.

Automatic determination of coreference between noun phrases is fraught with *uncertainty*. We show how fuzzy sets can be used to design a new coreference algorithm which captures this uncertainty in an *explicit* way and allows us to define varying *degrees* of coreference.

The algorithm is evaluated within a system that participated in the 10-word summary task of the DUC 2003 competition.

## 1 Introduction

Imagine the following task: of a set of texts on a particular topic you need to select which one(s) to read based on 10-word indicative summaries of the texts. Summaries can be of any form.

This describes Task 1 of the NIST sponsored DUC 2003 competition. Our approach to this task is simple: we order the entities[1] in the text by importance to the text and output representative NPs until we reach the limit.

We approximate the importance of an entity to a text by the number of times it is referred to in that text, that is by the length of its corresponding coreference chain.

In addition, we prefixed our summaries with a text category, generated using the classification tool *Bow* (McCallum, 1996) to supply some contextual information:

```
People:  construction project,
Schulz's work, voices, a
repository, his ``Peanuts'' strip
```

While the idea of using the length of coreference chains is not novel to the summarization community (see (Brunn et al., 2001; Lal and Rüger, 2002) just for the last two DUC competitions), our approach is distinguished by its purity: no other technique is used to identify material for the summary. Evaluations so far show a surprising success of this single summarization principle: In a set of 15 systems manually evaluated for "usefulness" by external evaluators, our system placed above average.

The core engine behind the summarizer is a knowledge-poor noun phrase coreference system called Fuzzy-ERS,[2] based on ERS (Bergler, 1997), which is similar in spirit, but simpler than (Baldwin, 1997). Knowledge-poor heuristics by nature are less reliable and we chose to model the certainty of their results explicitly, using *fuzzy set theory* (Zadeh, 1987; Witte, 2002a).

Using fuzzy theory allows Fuzzy-ERS to simultaneously consider all coreference possibilities, even if this temporarily assigns a NP to more than one coreference chain (albeit with different coreference certainties). This means greater flexibility, because the same coreference heuristics can lead to a strict or lenient system based simply on the choice of cut-off threshold, which can vary for different uses.

[1] Events are part of the output if they are referred to by NPs, but since they frequently corefer to predicates, they do not usually achieve their proper place in this system.

[2] ERS stands for Experimental Resolution System.

We describe our fuzzy coreference resolution algorithm in detail below and evaluate its usefulness on the summarization task outlined above.

## 2 ERSS — Summarization System

Input to ERSS is a tagged text (using Mark Hepple's Brill-style POS tagger (Hepple, 2000)). The major components used are:

**NPE** a noun phrase chunker that performs above 85%

**Fuzzy-ERS** a coreference resolution system using fuzzy logic

**Classifier** a naive Bayes classifier for multi-dimensional text categorization

**ERSS** the summarization system

ERSS is implemented in the GATE architecture (Cunningham, 2002) and uses some of the ANNIE components and resources provided with GATE as well as a classifier built with the *Bow* toolkit (McCallum, 1996) and WordNet (Fellbaum, 1998).

**Noun Phrase Extractor.** NPE uses a context-free NP grammar and an Earley-type chart parser to extract minimal noun phrases. Minimal noun phrases do not carry attachments, relative clauses, appositions, etc. Thus in our system *the president of the United States of America* generates three NPs, namely *the president*, *the United States*, and *America*.[3] The obvious setback of losing the semantics of this NP is offset by the fact that we avoid dealing with the ambiguity of PP attachment and have not compiled word lists for NPE.

The performance of NPE, evaluated with the GATE *Corpus Annotation Diff Tool* against a set of manually annotated texts, is shown in Table 1. Here, the *strict* measure considers all partially correct responses as incorrect, *lenient* regards all partially correct (overlapping) responses as correct, and the third column gives an *average* of both. The F-measure is computed with $\beta = 0.5$.

| **Precision** | min. | max. | average |
|---|---|---|---|
| *strict* | 52.23% | 72.15% | 62.85% |
| *average* | 64.33% | 83.12% | 75.00% |
| *lenient* | 75.95% | 94.09% | 87.15% |
| **Recall** | min. | max. | average |
| *strict* | 56.00% | 80.00% | 71.40% |
| *average* | 74.00% | 90.00% | 85.20% |
| *lenient* | 92.00% | 100.0% | 99.00% |
| **F-measure** | min. | max. | average |
| *strict* | 57.44% | 73.71% | 66.85% |
| *average* | 71.89% | 84.91% | 79.78% |
| *lenient* | 85.41% | 96.12% | 92.70% |

Table 1: Performance of the noun phrase extractor

Parsing errors are mostly due to tagging errors or the insufficiency of our context-free grammar.

**Fuzzy Coreferencer.** Fuzzy-ERS groups the NPs extracted by NPE into *coreference chains*, ordered sets of NPs that refer to the same entity. ERS was initially conceived as a baseline system, operating with almost no knowledge sources. It considers definite and indefinite NPs, dates, amounts, and third person pronouns.[4] It is based on a few shallow heuristics which operate on the ordered set of NPs produced by NPE. The different heuristics are distinguished by their likelihood to produce a valid result: string equality is more likely to indicate correct coreference than matching only by head noun. In (Bergler, 1997) this was addressed implicitly by a specific ordering of the heuristics. Using fuzzy values now allows us an explicit representation of the certainty of each stipulated coreference: a NP is assigned to a coreference chain with a certain likelihood. To determine the final coreference chains, the system can now be biased: setting a threshold of 1 for chain membership essentially removes the fuzzy component from the system and results in very short, accurate coreference chains. Setting a more lenient threshold allows more NPs into the chain, risking false positives.

We describe the design and influence of the fuzzy values below.

**Classifier.** The classifier is a naive Bayes model trained on a number of small, focused ontologies (which we call *Micro-Ontologies*), implemented with the *Bow* toolkit (McCallum, 1996). Each of these ontologies focuses on a particular topical categorization (e.g., disasters and their subtypes); together, they give a multi-dimensional categorization of a text. For example, using three of these ontolo-

[3] We repair some of this by using the named entity recognition component from ANNIE, which resolves *the United States of America* to a single named entity before it is fed to NPE.

[4] Pronoun resolution is inspired by (Hobbs, 1978; Lappin and Leass, 1994) but since we do not parse the entire sentence our algorithm is much cruder.

gies, a news article could be classified as $\{$*Politics, People, Single-Event*$\}$ within a three-dimensional space.

**Summarizer.** The summarizer is based on the simple idea that a 10-word summary should mention the most important entities of the text. We stipulate that the most important entities of a newspaper text are usually the ones corresponding to the longest coreference chains. Thus, for the summarization, all chains are *ranked*. The longest chain usually receives the highest rank, but the ordering is additionally influenced by a *boosting factor* that promotes chains with NPs that also occur in the first two sentences. Currently, we choose the longest NPs as representatives for the longest chains.

Thus, our summarization strategy can be summarized as follows:

1. output the most salient text classification with a simple decision-tree algorithm to provide come context
2. sort the coreference chains according to their ranking
3. select the longest noun phrase from each chain
4. output NPs as long as the length limit (10 words for the DUC 2003 Task 1) has not yet been reached.

## 3 Fuzzy Noun Phrase Coreference Resolution

The core idea for using a fuzzy-theory based resolution algorithm is the realization that coreference between noun phrases can neither be established nor excluded with absolute certainty. While statistical methods employed in natural language processing already model this *uncertainty* through probabilities, non-statistical methods that have been used so far had no systematic, formal representation for such imperfections. Instead, weights or biases are derived experimentally or through learning algorithms (Cardie and Wagstaff, 1999). Here, uncertainty is implicitly and opaquely dealt with in the system and changing it requires rebuilding the system or training set.

Our approach is to examine *explicit* representation and processing models for uncertainty based on fuzzy set theory (Zadeh, 1987; Klir and Folger, 1988; Cox, 1999). There are several advantages in explicitly modelling uncertainty: we do not have to choose arbitrary cut-off points when deciding between "corefering" and "not corefering", like for the semantic distance between words. Instead of such an a priori decision to be lenient or restrictive, we can dynamically decide on certainty thresholds to suit different processing contexts and this value itself can become part of the system deliberations.

As a consequence, we have more information available when building coreference chains, improving overall performance. Moreover, it is now possible to use the same result in different contexts by requesting a specific coreference certainty: a summarizer, for example, can decide to select only coreferences with a high certainty, while a full-text search engine might allow a user to retrieve information based on a more lenient certainty degree.

Our fuzzy noun phrase coreference resolution algorithm is based on the system described in (Bergler, 1997), but has been completely rewritten with the fuzzy-theory based representation model presented in (Witte, 2002a; Witte, 2002b). We now describe the fuzzy resolution algorithm in detail; we start with the representation model for fuzzy coreference chains, then describe the fuzzy resolution algorithm and its resources, and finally show how the computed fuzzy coreference chains can be converted into classical, crisp chains.

### 3.1 Modeling Fuzzy Coreferences

Fuzzy coreference chains are the basic representational unit within our fuzzy resolution algorithm. A single *fuzzy chain* $\mathscr{C}$ is represented by a fuzzy set $\mu_{\mathscr{C}}$, which maps the domain of all noun phrases in a text to the $[0,1]$-interval. Thus, each noun phrase $\mathrm{np}_i$ has a membership degree $\mu_{\mathscr{C}}(\mathrm{np}_i)$, indicating how certain this NP is a member of chain $\mathscr{C}$. The membership degree is interpreted in a possibilistic fashion: a value of 0.0 *("impossible")* indicates that the NP cannot be a member of the chain, a value of 1.0 *("certain")* means that none of the available information opposes the NP from being a member of the chain (*not* that it must be a member!), and values in between indicate varying degrees of compatibility of a noun phrase with the chain.

**Example (Fuzzy Coreference Chain)** Figure 1 shows an example for a fuzzy coreference chain. Here, the noun phrases $\mathrm{np}_3$ and $\mathrm{np}_6$ have a very high certainty for belonging to the chain, $\mathrm{np}_1$ only a medium certainty, and the remaining NPs are most likely not chain members.

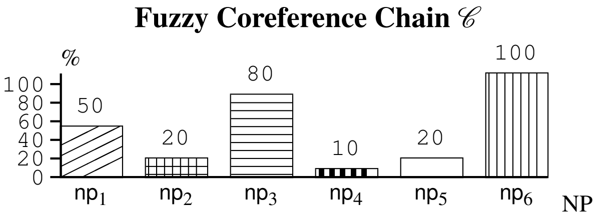


Figure 1: Example for a fuzzy chain showing the membership grades for each noun phrase

The output of our coreference algorithm is a set of fuzzy coreference chains, similar to classical resolution systems. Each chain holds all noun phrases that refer to the same conceptual entity. However, unlike for classical, crisp chains, we do not have to reject inconsistent information out of hand, so we can admit a noun phrase as a member of more than one chain, with a varying degree of certainty for each. This will be discussed later in more detail. We first show how fuzzy chains are constructed through *fuzzy heuristics*.

### 3.2 Fuzzy Heuristics

The fuzzy resolution system contains a number of heuristics for establishing coreference, each focusing on a particular linguistic phenomenon. Examples for fuzzy heuristics are pronominal coreference, synonym/hypernym-coreference, or substring coreference.

Formally, a fuzzy heuristic $\mathscr{H}_i$ takes as input a noun phrase pair $(\mathrm{np}_j, \mathrm{np}_k)$ and returns a fuzzy set $\mu^{\mathscr{H}_i}_{(\mathrm{np}_j,\mathrm{np}_k)}$ that indicates the certainty of coreference for the noun phrase arguments.

Such a certainty degree can be intuitively determined for almost all heuristics: an example is the synonym/hypernym heuristic, which has been implemented with WordNet (Fellbaum, 1998). Here, we assume two NPs that are synonyms corefer *certainly*, hence they are assigned a degree of 1.0. For hypernyms, our certainty decreases linearly with increasing semantic distance (we are currently evaluating different measures for semantic distance).

Heuristics[5] currently in use include:

**Synonym/Hypernym** the WordNet-based semantic distance heuristic mentioned above;

**Substring** a simple string comparison, assigning a 1.0 certainty for identical NP strings and a linearly decreasing coreference for substrings depending on their overlap;

**Acronym** a heuristic comparing NPs with their acronyms and abbreviations;

**Pronoun** a pronoun resolution algorithm, assigning lower coreference degrees for certain types of gender mismatches without degrading to the *impossible* certainty of 0.0; and

**Common Head** a comparison of the head noun of two NPs. We currently assign a coreference degree of *likely* (0.8) if two NPs match in their head noun.

New heuristics can easily be added to the system by placing them into the fuzzy coreferencer framework. This modular approach leads to a very robust system, since the result never depends solely on the performance of a single heuristic. For example, if a word cannot be located in the WordNet dictionary, the Synonym/Hypernym will not be able to establish a coreference degree, but one or multiple other heuristics have the chance to compensate for its failure.

The *design* of fuzzy heuristics brings new challenges to the system developer, however, since the uncertainty of a coreference must now be modeled explicitly. Our experiences show that this requires an additional initial effort, as uncertainty management and fuzzy set theory are not commonly used tools in computational linguistics. The start-up effort is worthwhile, though, since fuzzy heuristics turned out to be easier to design (no impedance mismatch between uncertain reality and computer model) and more powerful (retaining more information) than their classical, non-fuzzy counterparts.

### 3.3 Building Fuzzy Chains

The first step in the fuzzy coreference algorithm is the construction of *fuzzy chains*, holding the possi-

[5] For a motivation of these heuristics see (Bergler, 1997).

bilities of coreference represented by certainty degrees as described above. This is achieved by applying all fuzzy heuristics to each noun phrase pair.

More formally, given a text with $n$ noun phrases $\langle \mathrm{np}_1, \ldots, \mathrm{np}_n \rangle$ and $m$ fuzzy heuristics $\mathscr{H}_1, \ldots, \mathscr{H}_m$ we initialize for each $\mathrm{np}_j$ a fuzzy coreference chain $\mathscr{C}_j$ by collecting the coreference possibilities of $\mathrm{np}_j$ with all other NPs, represented by a fuzzy set $\mu_{\mathscr{C}_j}$:

$$\begin{aligned}\mu_{\mathscr{C}_j} \;:=\; & \mu^{\mathscr{H}_1}_{(\mathrm{np}_j,\mathrm{np}_1)} \cup \mu^{\mathscr{H}_1}_{(\mathrm{np}_j,\mathrm{np}_2)} \cup \ldots \cup \mu^{\mathscr{H}_1}_{(\mathrm{np}_j,\mathrm{np}_n)} \cup \\ & \mu^{\mathscr{H}_2}_{(\mathrm{np}_j,\mathrm{np}_1)} \cup \mu^{\mathscr{H}_2}_{(\mathrm{np}_j,\mathrm{np}_2)} \cup \ldots \cup \mu^{\mathscr{H}_2}_{(\mathrm{np}_j,\mathrm{np}_n)} \cup \\ & \ldots \cup \\ & \mu^{\mathscr{H}_m}_{(\mathrm{np}_j,\mathrm{np}_1)} \cup \mu^{\mathscr{H}_m}_{(\mathrm{np}_j,\mathrm{np}_2)} \cup \ldots \cup \mu^{\mathscr{H}_m}_{(\mathrm{np}_j,\mathrm{np}_n)}\end{aligned}$$

Thus, in this step we build as many fuzzy chains as there are noun phrases in a text. Each noun phrase is a member of each chain, but usually with varying degrees of certainty.

For the final result, however, we are interested in compiling all possible coreferences concerning a given NP into a single coreference chain. The next section describes a merging algorithm assuming that coreference is symmetric and transitive.

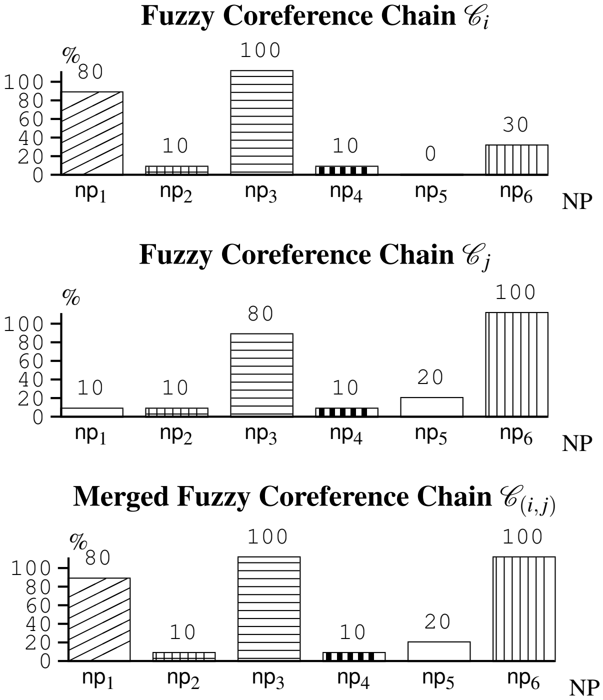
### 3.3.1 Merging Fuzzy Chains

All coreference possibilities concerning a noun phrase $\mathrm{np}_i$ are described in the fuzzy set $\mu_{\mathscr{C}_i}$, which constitutes an incomplete fuzzy coreference chain. Since the coreference relation is symmetric and transitive, if $\mathscr{C}_1$ establishes a coreference of e.g. $\mathrm{np}_1$ and $\mathrm{np}_3$ (with some certainty) and likewise $\mathscr{C}_2$ for $\mathrm{np}_3$ and $\mathrm{np}_5$, we expect the final result to also show a coreference for $\mathrm{np}_1$ and $\mathrm{np}_5$ in the same chain.

This is achieved by the process of *merging* the incomplete fuzzy chains into a set of complete chains where each chain holds all references to a single entity with a given certainty, prescribed by a *consistency* parameter $\gamma$, which is a threshold value for inclusion of a coreference possibility into the merged chain. The consistency of a fuzzy coreference chain $\mathscr{C}$ is defined as the consistency (maximum value) of its corresponding fuzzy set $\mu_{\mathscr{C}}$, denoted by $C(\mu_{\mathscr{C}})$. In order for a reference chain $\mathscr{C}_i$ to reach a consistency degree of at least $\gamma$, there has to be at least one noun phrase $\mathrm{np}_j$ in this chain with $\mu_{\mathscr{C}_i}(\mathrm{np}_j) \geq \gamma$ (note that every noun phrase corefers with itself to a degree of 1.0, so all initial chains $\mu_{\mathscr{C}_i}$ created by the algorithm above also have a consistency degree of 1.0). Thus, two chains are merged if their fuzzy set intersection[6] reaches at least the requested consistency degree $\gamma$:

$$\text{if } C(\mu_{\mathscr{C}_i} \cap \mu_{\mathscr{C}_j}) \geq \gamma, \text{ then } \mu_{\mathscr{C}_{(i,j)}} := \mu_{\mathscr{C}_i} \cup \mu_{\mathscr{C}_j}$$

A simple chain merging algorithm examines all possible chain combinations given a degree $\gamma$ and returns a list of merged fuzzy chains.


Fuzzy Coreference Chain $\mathscr{C}_i$

Fuzzy Coreference Chain $\mathscr{C}_j$

Merged Fuzzy Coreference Chain $\mathscr{C}_{(i,j)}$

Figure 2: Merging two fuzzy coreference chains with $\gamma = 0.75$

**Example (Chain Merging)** An example for the merging of two chains is shown in Figure 2. Here, a single new chain $\mathscr{C}_{(i,j)}$ (bottom) has been formed out of the two chains $\mathscr{C}_i$ and $\mathscr{C}_j$ (top) given a degree of $\gamma = 0.75$. If we had asked for a consistency degree of $\gamma = 1.0$, however, the chains would not have been merged since the consistency degree of both fuzzy sets' intersection is only 0.8.

With this algorithm, we can directly influence the result by changing the required consistency degree for an output chain; a degree of 1.0 corefers only 100% certain[7] NP pairs, a degree of 0.0 would core-

[6] We use the standard functions for possibilistic fuzzy sets, that is *min* for intersection, *max* for union, and $1 - \mu$ for computing the complement.

[7] Under a closed world assumption the degree of consistency corresponds to a degree of certainty.

fer all NPs into a single chain, and degrees in between result in chains of varying NP clusters according to their coreference certainty. The cut-off value $\gamma$ influences the results of ERSS directly (for the DUC 2003 ten word summary, we used the empirically chosen consistency degree of 0.6).

### 3.3.2 Defuzzification of Fuzzy Chains

Most of our existing processing resources have not yet been "fuzzified", hence, they still expect classical, crisp coreference chains. For these components we have to *defuzzify* our fuzzy chains.

We chose a simple defuzzification function: a crisp reference chain contains exactly the noun phrases having a membership degree of at least $\gamma$.

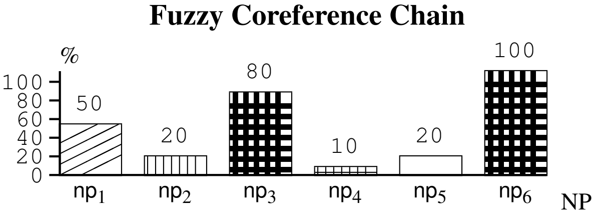


Figure 3: Defuzzification Example

**Example (Defuzzification)** An example is shown in Figure 3. With a certainty degree of $\gamma = 0.8$ we get the crisp result set $c = \{\text{np}_3, \text{np}_6\}$.

### 3.4 Performance of the fuzzy coreference resolution algorithm

The performance of the fuzzy coreference algorithm depends largely on two factors: the quality of the implemented heuristics (and their available resources) and the properties and settings of the fuzzy algorithm itself. Within this paper, we only analyze the second component, assuming a given set of fuzzy heuristics.[8]

The fuzzy coreference algorithm described above produces a similar result to its non-fuzzy counterpart when run with a consistency degree of 1.0.[9] However, with this algorithm we now gained the ability to explicitly request coreference results with different degrees of certainty.

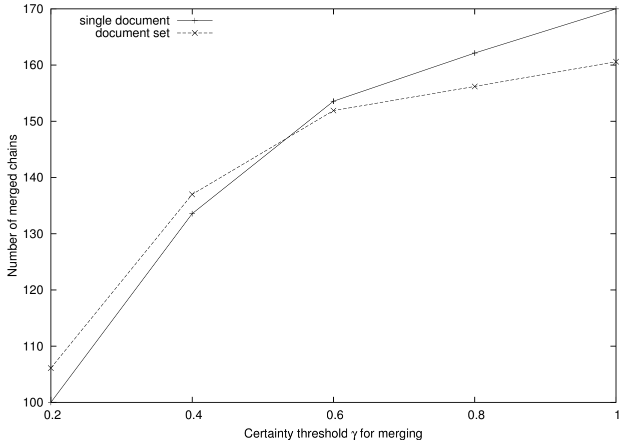


Figure 4: Number of resulting (merged) chains depends on the fuzzy value $\gamma$

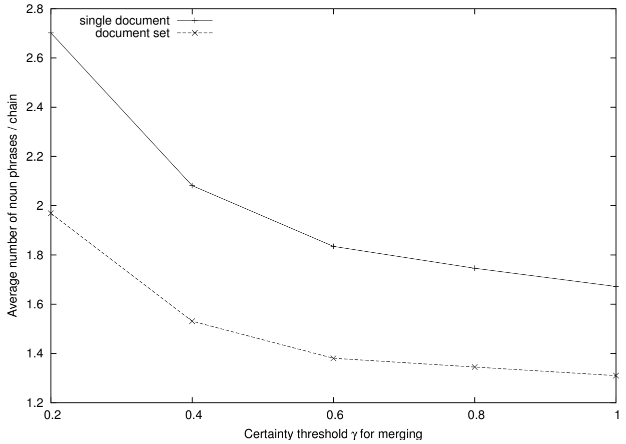


Figure 5: Different fuzzy values $\gamma$ result in chains of different lengths

The decisive parameter here is the consistency parameter $\gamma$ used for merging, effectively determining how certain a coreference must be to be admitted in a chain. Higher $\gamma$-values lead to a greater number of shorter chains that have a higher certainty of coreference between its NPs at the expense of completeness. Lower $\gamma$-values, in turn, result in fewer and longer chains, but might contain wrongly coreferred NPs.

This intuitive understanding of the fuzzy algorithm's behaviour has been experimentally confirmed during our evaluations for DUC 2003. Figure 4 shows how different settings for the certainty threshold $\gamma$ used in the merging phase of the algorithm influence the resulting chains: the lower the

[8] For alternative sets of heuristics see (Baldwin, 1997; Kameyama, 1997; Harabagiu and Maiorano, 1999).

[9] Of course, if we didn't want to exploit fuzzy theory we would have written the algorithm differently and thus the comparison is only illustrative.

requested certainty, the more chains are merged, resulting in fewer output chains (shown here are values for a single document containing 433 recognized noun phrases and values that were averaged over a 10-document set). Likewise, Figure 5 shows how the average number of NPs in a chain increases with a decreasing certainty threshold.

As can be seen, a fuzzy value of 0.2 results in comparatively long chains containing a higher average number of NPs. An empirical evaluation showed that these chains are not very useful, however, since they contain many wrong coreferences (after all, a certainty of 20% is not very high). Likewise, coreference chains with a certainty of 1.0 tended to be too fragmented for our intended application, automatic summarization. Intermediate fuzzy values lead to good coreference chains that produce useful results, as we will show below.

## 4 Evaluation for Summarization

Fuzzy-ERS works with very knowledge-poor techniques depending solely on isolated minimal NPs. It is thus much less sophisticated than other NP coreference systems. Because of the direct influence of $\gamma$ on precision and its inverse relationship with recall, we chose to evaluate the usefulness of fuzzy theory for coreference resolution on the summarization task.

We evaluate Fuzzy-ERS on 10 word summaries. With $\gamma = 0.6$ we include some very inaccurate NPs in chains, especially the WordNet derived distance measure is very permissive at that value. Yet the benefit of overcoming the chain fragmentation of higher thresholds still outweighs the imprecision of some chains.

NIST assessors evaluated ERSS summaries against manually constructed target summaries of different styles: Some were single sentences, some multiple sentences, some resembled our output very closely and some mixed the other styles. This was a feature of this year's target summaries: not to penalize a system too much for stylistic differences, NIST had four summaries prepared for each text and selected one at random for the target summary.

ERSS was judged to give relevant summaries in 83% of the cases. Coverage overall was judged at 29%. Usefulness was judged average at 1.82 over a scale from 0 (bad) to 4 (excellent).

| | Documents | | | Directories | | |
|---|---|---|---|---|---|---|
| | min | max. | avg. | min. | max. | avg. |
| *Recall* | 0 | 100 | 44.7 | 26 | 71 | 48.5 |
| *Precision* | 0 | 100 | 44.5 | 26 | 69 | 47.5 |
| *F-measure* | 0 | 100 | 42 | 26 | 62 | 44 |

Table 2: Performance of ERSS over 264 Documents in 60 Directories

We manually evaluated the ERSS on the same target summaries. To compare our output with the target summary, we choose to split the target into *concept-tokens* (CTs), where tokens could be single nouns, noun phrases and possibly verbs. CTs are thus similar to and comparable with ERSS's output.

Any CT that matches against an output NP counts as one hit. We do not count or compare with the output of the classifier, since the document type information given by our classifier is not present in the target summaries.

The match can be partial, 'Asian Games' and 'Second Asian Games' count as a hit, as does 'drug trade' and 'China's major drug problem', where we have a common drug-problem concept.

Once concept-tokens are matched against ERSS's NPs, recall and precision are measured, and consequently the F-measure. Table 2 shows the maximum, minimum, and average values for recall, precision, and the F-measure for the summary comparison.

No hits happen when either ERSS returns the general event such as 'International Human Rights Treaty', while the manual summary goes more into details and is about an 'arrest', or ERSS and the manual summary each cover a distinct idea in the text, and we get 'Bad weather' vs. 'No Satellite Damage', or for the SwissAir Flight 111 example, 'the dead' vs. 'the plane's wreckage'. On the other hand, we score 4% of maximum recall over 14 different directories. The manual summaries in this case are short and headline-like. When it's the other way round, i.e. ERSS returns 2 to 3 NPs, precision is at its best. This occurs 3.5% of the time over 7 directories.

We feel that this performance validates our approach: coreference resolution is part of the known toolkit for summarization. Yet a system that uses as its single summarization strategy the length of NP coreference chains performs average. This argues convincingly that Fuzzy-ERS succeeds in highlighting important text entities and we will now refine its algorithm and embed it into a more sophisticated environment.

## 5 Conclusions and further work

We showed how the uncertainty arising in non-probabilistic natural language processing can be modelled explicitly with a fuzzy-theory based representation formalism. This allows for an interesting new approach to noun phrase coreference resolution, using fuzzy heuristics and fuzzy coreference chains that can adapt dynamically to different certainty requirements. It has been successfully integrated into an automatic summarization system built for the DUC 2003 competition.

Currently, we are continuing the evaluation of our fuzzy coreferencer and are in the process of refining and adding more heuristics.

Additionally, we started work on multi-lingual fuzzy coreference resolution for the French and German language. First tests suggest that the approach used for English translates well into other languages, since most resources are relatively knowledge-poor.

A remaining challenge, however, is to rewrite more of our existing processing components for the fuzzy model, allowing them to take full advantage of the augmented information representation and processing capabilities that are now available.

**Acknowledgements.** We would like to thank our students *Michelle Khalife*, *Zhuoyan "Robert" Li*, and *Frank Rudzicz* for their tireless work during the implementation and evaluation of our system.

## References

Breck Baldwin. 1997. Cogniac: High precision coreference with limited knowledge and linguistic resources. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, pages 38–45, Madrid, Spain, July.

Sabine Bergler. 1997. Towards reliable partial anaphora resolution. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, Madrid, Spain, July.

M. Brunn, Y. Chali, and C.J. Pincha. 2001. Text summarization using lexical chains. In *Document Understanding Conference (DUC)*, New Orleans, Louisiana USA, September 13-14, 2001.

Claire Cardie and Kiri Wagstaff. 1999. Noun phrase coreference as clustering. In *Proceedings of the Joint Conference on Empirical Methods in NLP and Very Large Corpora*, Maryland.

Earl Cox. 1999. *The Fuzzy Systems Handbook*. AP Professional, 2nd edition.

H. Cunningham. 2002. Gate, a general architecture for text engineering. *Computers and the Humanities*, 36:223–254. `http://gate.ac.uk`.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Sanda Harabagiu and Steven Maiorano. 1999. Knowledge-lean coreference resolution and its relation to textual cohesion and coherence. In *Proceedings of the ACL'99 Workshop on the Relation of Discourse/Dialogue Structure and Reference*, pages 29–38, University of Maryland, June.

Mark Hepple. 2000. Independence and commitment: Assumptions for rapid training and execution of rule-based pos taggers. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL-2000)*, Hong Kong, October.

Jerry R. Hobbs. 1978. Resolving pronoun references. *Lingua*, 44:311–338.

Megumi Kameyama. 1997. Recognizing referential links: An information extraction perspective. In *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, Madrid, Spain, July.

George J. Klir and Tina A. Folger. 1988. *Fuzzy Sets, Uncertainty, and Information*. Prentice-Hall.

P. Lal and S. Rüger. 2002. Extract-based summarization with simplification. In NIST, 2002.

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

Andrew Kachites McCallum. 1996. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. `http://www.cs.cmu.edu/~mccallum/bow`.

NIST. 2002. *DUC 2002 Workshop on Text Summarization*, Philadelphia, Pennsylvania, USA, July 11-12.

René Witte. 2002a. *Architektur von Fuzzy-Informationssystemen*. BoD. ISBN 3-8311-4149-5.

René Witte. 2002b. Fuzzy belief revision. In *9th Intl. Workshop on Non-Monotonic Reasoning (NMR'02)*, pages 311–320, Toulouse, France, April 19–21. `http://rene-witte.net`.

L.A. Zadeh. 1987. Fuzzy sets. In R.R. Yager, S. Ovchinnikov, R.M. Tong, and H.T. Nguyen, editors, *Fuzzy Sets and Applications: Selected Papers by L.A. Zadeh*, pages 29–44. Wiley&Sons. Originally published in *Information and Control*, Vol. 8, New York: Academic Press, 1965, pp. 338–353.

# A Differential Representation of Predicates for Extensional Reference Resolution

**Guillaume Pitel**
LIMSI-CNRS
BP 133
F-91403 Orsay Cedex
pitel@limsi.fr

**Jean-Paul Sansonnet**
LIMSI-CNRS
BP 133
F-91403 Orsay Cedex
jps@limsi.fr

## Abstract

In this paper, we focus on a method in practical dialogue for resolving extensional descriptions containing vague or relational predicates as well as predicates on intrinsic properties. It is shown that different kinds of predicates can be handled within a unified approach. This method is built upon the work of Salmon-Alt and is intended to be included into her general resolution model, extending her account for differentiation criteria. We indeed argue about the lack of expressiveness of logical predicates when dealing with vague or context-sensitive linguistic predicates. Our solution is a differential, function-oriented approach to reference resolution, based on the idea that predicates' meaning may be represented with a comparison function and two partitioning functions. In order to address these problems, we make two propositions. The first is that reference resolution process should primarily rely on the use of a comparison function, not on predicates indicating absolute properties of entities. The second is that functions we use to represent the semantics of referential extractors should be able to take context as an argument.

## 1 Extensional Reference Resolution in Practical Dialogue

In the framework of practical dialogue, extensional reference resolution is defined as the process of searching for entities referred to by a linguistic description (Byron and Allen, 2001). We have particularly examined the case of descriptions containing so-called vague predicates such as *tall* or *heavy*, but also relational predicates such as *on the left to* and purely intrinsic predicates such as *red* or *square*. While there is a large literature about vagueness in the field of linguistics, its application to reference resolution in dialogue systems is almost non-existent. The main reason is that literature about vague predicates focuses on a descriptive in statements, while reference resolution rather requires an extensional interpretative account. Because of this, most extensional reference resolution methods in dialogue systems use a basic logical approach. Within a logical predicative approach, reference resolution is handled by considering all the entities of a model *M* satisfying the predicates derived from the description. For instance, when a user says "*the big red square*", a predicative approach would result in searching for entities satisfying the formula: $big(x) \wedge red(x) \wedge square(x)$. When one wants to cope with more complex properties, such as spatial relations, it becomes necessary to make use of specific heuristics (Winograd, 1973).

This paper presents an extension of the reference resolution frame defined in (Salmon-Alt, 2001b). Contrary to Salmon-Alt's position about differentiation criteria, we adopt a function-oriented view about the meaning of referential extractors, and examine the idea of using differentiation procedure in place of differentiation criteria. This approach allows suppressing the need for absolute valuation of entities' properties, and thus can be used for strong qualitative predicates, such as *true/false* or *masculine/feminine* as well as quantitative predicates. of vague predicates. Splitting the semantic representation of vague predicates into several functions moreover allows dispatching the account for context and helps explaining the processing order of the predicates.

## 2 Mental Representations and Referential Extractors

Salmon-Alt (2001) examines a model for recruiting contextual entities referred to by referential expressions. Her model is based on the idea of mental representations (Reboul, 1999) used to stand for the dialogue's context and more specifically reference domains. Reference domains are local contextual sets of entities. They are structured in order to reflect the way they were built, and consequently can be used to predict the distribution of different reference markers. In this model, mental representations introduced by a referential expression can be recursively used as reference domains.

This model however fails in representing relational constraints on entities, such as *'big'*. Indeed, in our opinion, Salmon-Alt fails to distinguish between referential extractors and entities properties. For instance, when processing *"the big circle"*, she proposes to partition the initial reference domain between *big* and *¬big*. In other words, it separates entities for which property *size* is *big* from others.

*Referential extractors* are the parts of a referential description that constrain properties of the expression's extension, i.e. the objects[1] of the world referred to by the expression. Typically, in a sentence like *"Take the blue box"*, the word 'blue' serves as an extractor for objects having a color that can be characterized as blue. In this sentence, *'box'* is also a referential extractor, but *'the'* plays another role. The role of determiners is best described by the term *meta-extractor*, since they control the way extractors are used in the resolution process. It is also the case of degree words and modifiers. Salmon-Alt's model has been primarily designed to account for the behavior of different kinds of referential expressions for French, such as demonstrative, definite, indefinite and pronominal forms. In our model, referential extractors are explicitly distinguished from other roles. For instance, in the following sentences, '*blue*' is not a referential extractor: *"Is your car blue?"*, *"Blue squares are beautiful"*, *"Add a blue square on the grid"*.

### 2.1 On the vagueness of properties

In practical dialogue systems, semantics of size adjectives like *big* or *small* are often reduced to predicates. However as (Dale and Reiter, 1995) notice it seems obvious that the entity referred to by an expression containing an adjective such as *'big'* or *'small'* is not supposed to hold a property asserting whether it is objectively big or small.

While it is almost not explored in practical dialogue, vagueness of properties is thoroughly studied in linguistics domain. Various approaches have been proposed, either qualitative (Kamei and Muraki, 1994) or quantitative (Simmons, 1993) or hybrid (Staab and Hahn, 1997). These studies primarily focus on the problems raised by vague predicates in positive, comparative, equative or superlative assertions (e.g. *"Paul is slightly taller than Susan"*). An issue that seems more important to us for practical dialogue is the problem of resolving reference containing vague predicates.

Whereas literature about vagueness proposes interesting solutions for the logical representation of vague predicates, and for the effect of degree words used in conjunction, it doesn't help to find applicable solutions for the specific problem of extensional reference resolution.

Pateras et al. (1995) have proposed to deal with reference resolution containing weight predicates with a method based on fuzzy logic. The authors propose to use functions based on fuzzy sets to choose the right referent. However, these functions are defined relatively to a particular task, and are not designed to take ontological or operating context into account,

[1] *Objects* are internal representations of the software with which the user want to talk, while *entities* denote underspecified elements manipulated by the natural language understanding system.

because they have no influence on the task presented in their article. According to us this does not help much for solving our problem, since dynamic context is still not considered. The same problem can be found in (Lammens and Shapiro, 1993) where the authors construct color categorization functions with a learning algorithm, but do not address influences from the context.

An interesting approach is drawn in (Staab and Hahn, 1997), which serves to compute the comparison class with which the vague predicate is applicable. The proposed algorithm finds the concept (a subpart of the ontological context) within which the predicate must be applied. The problem with this approach is that the comparison class can not be defined with the dynamic knowledge from the operating context but only with the ontological knowledge.

In order to address these problems, we make two propositions. The first is that reference resolution process should primarily rely on the use of a comparison function, not on predicates indicating absolute properties of entities. The second is that functions we use to represent the semantics of referential extractors should be able to take context as an argument.

### 2.2 Properties, Relations and Types

Dale and Reiter (1995) distinguish three kinds of referential extractors: those involving typological properties (e.g. being a figure, being a square…), those involving intrinsic properties (e.g. size, color) and those involving relational properties (e.g. spatial position, temporal relations).

Most dialogue system's models assume access to an adequate representation of world entities, coherent with the way the user refers to them in natural language. This can be observed even in recent work such as (Salmon-Alt, 2001a:149), (Byron et al., 2001) or (Dale et Reiter, 1995) with usage of predicates like *(color x RED)* or *(size x SMALL)* for selecting red and small objects. Such models assume that referential extractors purely rely on intrinsic properties of objects to designate entities. Consequently, dialogue systems cannot propose a unified account for relational constraints, leading to ad hoc approaches to relational references resolution.

We make the hypothesis that these three kinds of extractor can be represented in a unified way using a comparative or differential approach. Indeed, many intrinsic properties are actually relational, which is shown by their use in comparative clauses. For instance, one can say "*Your shirt is redder than mine",* or "*Your boyfriend is more masculine than mine*". The main hypothesis sustaining our proposal is that even strong predicates are actually relational predicates applied on the target entity against a default, prototypical entity. For instance a prototypical male for *masculine*, or a prototypical color for *red*.

## 3 Differential Representation

We propose to represent referential extractors with a structure of functions that is to be used by a generic resolution algorithm. For the purpose of illustrating our view, we define a situation described in FIG. 1. In order to focus the discussion on the issue of resolving intrinsic references, we will consider that elements of the mediated software are *objects* (in the object-oriented algorithmic sense). These objects are defined by a type and two attributes: color (a 3-uple of values in (0, 1) and a size (a value in (0,∞)).

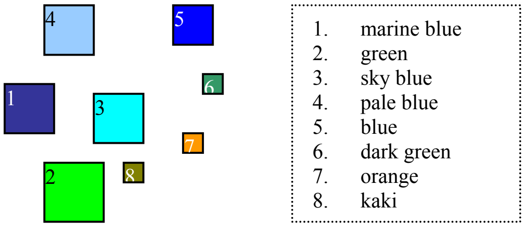


FIG. 1: Experiment case for procedural representation of size and color.

### 3.1 Functional Representation of Referential extractors

We represent referential extractors as a structure of three functions. These functions are detailed in FIG. 2. The *fSimil* comparison function is used to sort a set of objects among a given property (say, *blueness* or *bigness*). For instance, the *fSimil* function of the referential extractor *'blue'* give the similarity ratio between the colors of two objects projected along the blue axis[2]. If *a* is bluer than *b*, then the returned value is above 1, while the contrary returns a value between 0 and 1 (exclusive), and equality returns 1. If one of the two colors is outside the area considered to be the limit of blue colors, returned value is ⊥.

The *fExcl* function serves to select in the sorted list (produced by the resolution algorithm using *fSimil* function) objects that will be excluded from the candidates list that is going to be submitted to the next extractor. All extractors do not select the same way, for instance extractor *big* will select objects of the same size as the littlest one (that is, the last one in the list). For any extractor for *color*, this function selects all objects with similarity ratio ⊥.

The *fPref* function selects the preferred objects for a given extractor. Here again, there is no general rule for the function. The extractor *big* could select it based on any heuristic (either the 30% upper sizes, or all the objects from the beginning to the first point where the secondary differential coefficient is over a given threshold).

Referential Extractor

- *fSimil*(entityType1 a, entityType1 b, [RefDomain⟨entityType1⟩ d]) → (0, ∞) ∪ ⊥
- *fExl*(RefDomain⟨entityType1⟩ d, ) → RefDomain⟨entityType1⟩ partitionedDomain
- *fPref*(RefDomain⟨entityType1⟩ d) → RefDomain⟨entityType1⟩ partitionedDomain

FIG. 2 : Referential extractor representation

### 3.2 Resolution Algorithm

The resolution algorithm is triggered during the natural language analysis process when an extensional referring expression has been isolated. The rules leading the algorithm follow the guidelines defined in (Salmon-Alt, 2001b), so that the type (definite, indefinite, demonstrative, pronominal) of the referring expression determines the way the initial reference domain is built, and how the extractors are used to return the final result of the reference resolution in the *restructuring* phase.

We consider each referential extractor to be a sub-process of the following form[3]:

1. Transform each entity of the original reference domain through[4] a point of view that would make the extractor able to process the domain. The initial reference domain may be sorted, partitioned and focused by a preliminary step.
2. Use the *fSimil* comparison function of the extractor in order to sort the domain by the adequate criteria.
3. If the produced reference domain is to be passed to another extractor, or the referring expression is in plural form, use the *fExcl* function to partition the domain among possible and impossible candidates.
4. Use the *fPref* function to extract best candidates if the referring expression is in plural form.

Compared to Salmon-Alt's differentiation criteria, our algorithm makes use of two different partition functions (*fExcl* and *fPref*) in order to produce a ternary partition instead of a binary one. Indeed, as we introduce a total order relation in the differentiation process, we are faced with an issue that was hidden when using simple predicates: when referring to several entities, like in "*remove blue squares*", the user refers to several objects at the top of the list sorted by blueness; however, squares less blue but still blue must be distinguished from non-blue squares, because if the user asks for "*remove big blue squares*", and big squares are not at the top of the list sorted by blueness, they must be passed from *'blue'* extractor to *'big'* extractor in the list of possible candidates for the referring expression. So it is necessary to distinguish between best candidates, needed to answer to group references, and possible and impossible candidates in order to respond "nothing matches" to ambiguous referential expressions.

[2] Do not mistake it for the blue component of a red, green and blue decomposition of color. The *blue axis* or *dimension* represents the distance (in a cognitive view) of any color compared to blue. For instance, sky blue or indigo are farer from blue than marine blue. Note that there is no measurable distance between green and yellow on the blue axis, and thus not all colors can be projected on this axis.

[3] The full algorithm is not detailed here, because of its length. Moreover its details still have to be checked against results of not yet finished psychological experiments.

[4] The transformation through a point of view changes all objects of the domain into objects of a class suitable for the fSimil comparison function. This point is crucial, but out the scope of this paper, and will not be discussed further.

### 3.3 A practical situation

From the situation presented in FIG. 1, we illustrate how to use differentiation functions in the case of resolving the referring sub-expression *"big blue squares"*.

We take as a starting point for reference resolution the referential domain $\mathrm{RD}_{\mathrm{initial}}$ of all the squares of the scene. The referential domain contains labels of objects. In order to shorten the example, we will not detail the *'squares'* extractor, since all objects of the scene are squares. We first apply the referential extractor for *'blue'*[5]. Theoretically, this function will produce when applied to a set of $n$ objects, a matrix of $n{\times}n$ elements. For all practical purpose, one can consider that in many cases, $fSimil_i(x,z) = fSimil_i(x,y) \times fSimil_i(y,z)$, providing a full order relation. This point could be subject to discussion, but we will consider that it is a reasonable assumption. Algorithms for sorting sets with a full order relation have complexity of order $O(n\log n)$. Using the $fSimil_{blue}$ function, we construct an ordered list ($\mathrm{Ord}_{\mathrm{blue}}$) whose links between objects are annotated with similarity ratios ($\mathrm{Sim}_{\mathrm{blue}}$), and partition it with the two functions $fExcl_{blue}$ and $fPref_{blue}$.

$\mathrm{RD}_{\mathrm{initial}} = \{1, 2, 3, 4, 5, 6, 7, 8\}$

$\mathrm{Sim}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}) = \{\langle(5, \mathrm{proto}_{\mathrm{blue}}), 0.98\rangle, \langle(1, \mathrm{proto}_{\mathrm{blue}}), 0.9\rangle, \langle(3, \mathrm{proto}_{\mathrm{blue}}), 0.6\rangle, \langle(4, \mathrm{proto}_{\mathrm{blue}}), 0.55\rangle, \langle(6, \mathrm{proto}_{\mathrm{blue}}), \perp\rangle, \langle(7, \mathrm{proto}_{\mathrm{blue}}), \perp\rangle, \langle(8, \mathrm{proto}_{\mathrm{blue}}), \perp\rangle\}$

$\mathrm{Ord}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}) = 5 > 1 > 3 > 4 > x\ \forall x \in \{2, 6, 7, 8\}$

$\mathrm{Pref}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}) = \{5 > 1\}$

$\mathrm{Poss}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}) = \{5 > 1 > 3 > 4\}$

$\mathrm{Excl}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}) = \{2, 6, 7, 8\}$

As $\mathrm{card}(\mathrm{Pref}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}}))$ is only 2, and the referring expression is in plural form, *big* must at least take the context from $\mathrm{Poss}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}})$ for argument (because it must manage to make a partition).

$\mathrm{Sim}_{\mathrm{big}}(\mathrm{Poss}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}})) = \{\langle(3,4), 1.\rangle, \langle(4,1), 1.\rangle, \langle(1,5), 0.75\rangle\}$

Since the similarity ratio is too high between square 1 and square 5, the partitioning is impossible. The context must be enlarged. The extraction is then applied on the whole original domain.

$\mathrm{Sim}_{\mathrm{big}}(\mathrm{RD}_{\mathrm{initial}}) = \{\langle(3,4), 1.\rangle, \langle(4,1), 1.\rangle, \langle(1,5), 0.75\rangle, \langle(2,3), 0.9\rangle, \langle(5,6), 0.25\rangle, \langle(6,7), 1.\rangle, \langle(7,8), 1.\rangle\}$

$\mathrm{Poss}_{\mathrm{big}}(\mathrm{RD}_{\mathrm{initial}}) = \{2 > 1 > 3 > 4 > 5\}$

$\mathrm{Excl}_{\mathrm{big}}(\mathrm{RD}_{\mathrm{initial}}) = \{6, 7, 8\}$

As the partitioning succeeded, one can compute the intersection between $\mathrm{Pref}_{\mathrm{blue}}(\mathrm{RD}_{\mathrm{initial}})$ and $\mathrm{Poss}_{\mathrm{big}}(\mathrm{RD}_{\mathrm{initial}})$. The result of reference resolution for *"big blue squares"*, in the situation of FIG. 1 produces squares number 1 and 5 as the best candidates. Same method applied to "small blue square" produces the square number 5. In these two cases, the result seems to corroborate the basic intuition and the human validation[6].

### 3.4 Arranging extraction operations

The processing order of referential extractors is of great importance in extensional resolution. Salmon-Alt's model rely on the propositions that (Dale and Reiter, 1995) have made about the relative importance of objects' properties in the framework of referential expressions generation. The authors mainly base their research on the implication of Grice's maxims (Grice, 1975) on reference generation, following the principle of maximum economy in language generation. Authors propose to take properties into account in the following order: type > intrinsic property > relational property. That means that, when a user refers to an entity, he first uses the type of the entity to describe it, then if it is not sufficient to distinguish the entity from others, he use an intrinsic property, and finally a relational property. For instance, if (1) "*The red square*" and (2) "*The square on the right*" refers to the same object, (1) will be used, since it contains only an intrinsic property.

Adapting this approach from generation to interpretation leads to consider that in a referring expression, extractors (mainly nouns and adjectives) must be processed in the same order, first the nouns, then intrinsic adjectives, then relational properties. As a consequence, the interpretation of *"the big blue square"* begins with the selection of squares giving a set of entities $S_1$, then blue objects from the set $S_1$ giving the set

[5] We address in section 3.4 the issue of extractors' processing order.

[6] We have made a first experiment with a dozen of volunteers. Participants have been faced with several situations like the one described in FIG. 1. They were then asked to select on paper the objects denoted by a given sentence, for instance "the big squares" or "the blue squares".

$S_2$, and finally the big objects from the set $S_2$, giving the set $S_3$ used to find the best candidate for the reference. It is obvious that choosing another order will lead to different results, incompatible with the natural interpretation.

Our model of context handling provides a rationale for the empirical observation of resolution order. Indeed, while the calculation of the subset referred to by a type is computationally cheap, the calculation of the subset of entities referred to by a relational adjective is very expensive. Moreover, the more the predicate's meaning is potentially influenced by context, the more probable is the fact that context must be included in the computation in order to ensure that the reference domain is correct.

The cost of processing an extractor is related to the quantity of contextual information that is necessary to take into consideration. As the context is restricted by each extractor, low-cost extractors must be processed first. We propose the following principle:

> *The more the operating context is important to process an extractor, the later this extractor should be put in the resolution chain.*

As the perception of colors is only a little dependent on the situation, whereas the perception of the size is highly related to the other objects present in the scene, the arrangement of extractor execution is *color* then *size*. This approach offer the advantage to explain also why the type (the noun) is always the most important of the properties and is always considered before other ones. So the expression "*the big blue square*" is not disambiguated as: *"the squares among the blue objects among the big objects"*, but as: "*the big ones among the blue ones among the squares*".

## 4 Conclusion

We have made two propositions:

- That all kinds of referential extractors should be accounted within a differential approach.
- That functions used to represent referential extractors take the context as arguments.

We also have proposed the principle that extractors processing order in a referential chain should be computed from the amount of contextual information needed to process each extractor. Those propositions are made to be integrated into a resolution model which follows Salmon-Alt's guidelines. This approach aims to cover relational and typological extractors as well, but there is still some important work to be done in order to reach this goal. One of the issues raised by this extension of the preliminary model is the need for a mechanism to measure the distance between two objects in any dimension. For typological properties, this implies to perform a metaphorical transformation of reference domains from one dimension to another.

Our team is currently developing this mechanism inside the InterViews project (Sansonnet et al., 2002). This project is built around the concept of conversational agents for assistance to ordinary people. We aim to provide a platform for helping design of natural language interface to common software. The formalism presented in this paper, as well as the overall model of natural language analysis integrating it, is under development in this framework. There is consequently no evaluation of this system so far, but we plan to reach a working platform in order to be able to compare results from our reference resolution method with the results of psychological experimentations.

## References

Donna K. Byron and James F. Allen. 2002. What's a Reference Resolution Module to do? Redefining the Role of Reference in Language Understanding Systems. In *Proc. of the 4th DAARC.*

Robert Dale and Ehud Reiter. 1995. Computational Interpretation of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science*, 18, pp. 233-265.

Herbert Paul Grice. 1975. Logic and conversation. In P. Cole and J. Morgan, editors, *Syntax and Semantics*, 3, Speech Acts, pp. 43-58. New York, Academic Press.

Shin-Ichiro Kamei and Kazunori Muraki. 1994. A discrete model of degree concept in natural language. In *Proc. Of COLING-94*, 2, pp. 775-781.

Johan M. Lammens and Stuart C. Shapiro. 1993. Learning Symbolic names for Perceived Colors, in *Machine Learning in Computer Vision: What, Why and How?* AAAI-TR FSS93-04.

Claudia Pateras, Gregory Dudek, Renato DeMori. 1995. *Understanding Referring Expressions in a Person- Machine Spoken Dialogue*. In *Proc. of the ICASSP'95*, Detroit, MI.

Anne Reboul. 1999. Reference, agreement, evolving reference and the theory of mental representation, in Coene, M., De Mulder, W., Dendale, P. & D'Hulst, Y. (eds), *Studia Linguisticae in honorem Lilianae Tasmowski*, pp. 601-616, Padova, Unipress.

Jean-Paul Sansonnet, Nicolas Sabouret and Guillaume Pitel. 2002. *An Agent Design and Query Language dedicated to Natural Language Interaction*. In *Proc. of AAMAS 2002*.

Susanne Salmon-Alt. 2001a. *Référence et dialogue finalisé : de la linguistique à un modèle opérationnel*, PhD Thesis, Université H.Poincaré - Nancy 1, France.

Susanne Salmon-Alt. 2001b. Reference Resolution within the Framework of Cognitive Grammar. In *Proc of the International Colloquium on Cognitive Science*, San Sebastian, Spain

Steffen Staab and Udo Hahn. 1997. "Tall", "Good", "High" – Compared to What? In *Proc. of the 15th International Joint Conference on Artificial Intelligence*, pp. 996-1001.

Geoffrey Simmons. 1993. A tradeoff between compositionality and complexity in the semantics of dimensional adjectives. In *Proc. of the EACL-93*, pp. 348-357.

Terry Winograd. 1973. A Procedural Model of Language Understanding, *Computer Models of Thought and Language*, Roger Schank & Kenneth Colby (eds.), W. H. Freeman Press.

ARQAS - Venice, 23-24 June 2004.

# Reference Resolution as a facilitating process towards robust Multimodal Dialogue Management : A Cognitive Grammar Approach

**Ashwani Kumar** **Susanne Salmon-Alt** **Laurent Romary**
Laboratory LORIA
Campus Scientifique, B.P. 239
54506 Vandoeuvre-lès-Nancy, France
{ashwani.kumar, susanne.alt, laurent.romary}@loria.fr

## ABSTRACT

This paper tries to fit a novel reference resolution mechanism into a multimodal dialogue system framework. Essentially, our aim is to show that a typical multimodal dialogue system can actually benefit from the cognitive grammar approach that we adopt for reference resolution. The central idea is to construct and update reference and context models in a manner that imparts adequate level of underspecificity to multimodal semantics. Context-independent semantic representations are constructed based upon the surface structure of the referring expressions and syntactic constraints within an utterance. The reference resolution algorithm assimilates these semantic representations into a coherent context model, resulting in the profiling of the intended referent. The resolution model is built upon discursive, perceptual and conceptual cues, thus successfully accounting for multiple modalities and a multi-dimensional application domain model.

## 1 Introduction

The complexity of reference resolution is due, in part, to the variety of referring expressions, including indefinites, definite descriptions, pronominal reference and ellipses or *one*-anaphora. The problem is aggravated by the apparent variety of mechanisms required to deal with even one of these types of referring expressions. For example, the referent of a definite description may be linked to a prior discourse entity with the same head, associated to a prior entity from which it can be inferred, or extracted from a larger situation (Poesio and Vieira, 1998). As a result, much current work on reference centers around pronominal reference (cf. Centering Theory: Grosz et al., 1995; McCoy and Strube, 1999). The treatment of other types of referring expressions is often seen as an extension of or variation on the basic co-referential mechanism (DRT and it extensions: Kamp and Reyle, 1993; Bos et al., 1995).

Additionally, the interpretation of referring expressions is based on both discourse and perceptive context. For example, "another one" cannot be understood without previous discourse mention of, let's say "a romantic song". The need of perceptive information is evident for expressions like "the last two song writers" referring to a list displayed on a screen. What we need then is a unified framework to represent and update dynamically the information provided by both discourse and perceptive context and to constrain the access to this information. DRT, for example, provides access to all previously mentioned entities, while Centering Theory considers the previous discourse unit only. On the other hand, within the list of identified potential referents, Centering Theory provides a precise account of relative salience, whereas DRT specifies only general syntactic constraints to narrow the list. Some recent models attempt to apply more precise selectional criteria to global discourse (Asher, 1993; Hahn and Strube, 1997). However, all rely on some prior segmentation, implicitly assuming that discourse structure informs reference resolution, and ignoring the possibility of determining structure based on referential devices or on perceptive information.

Finally, we notice a gap between the predictions made by approaches in analysis and the generation of referring expressions. In an example like "Select a song and play **it** / **the song**", DRT-like models do not predict any difference between pronominal and nominal anaphora whereas a generation model based on Dale and Reiter (Dale, 1992; Dale and Reiter, 1996) would largely prefer the pronoun.

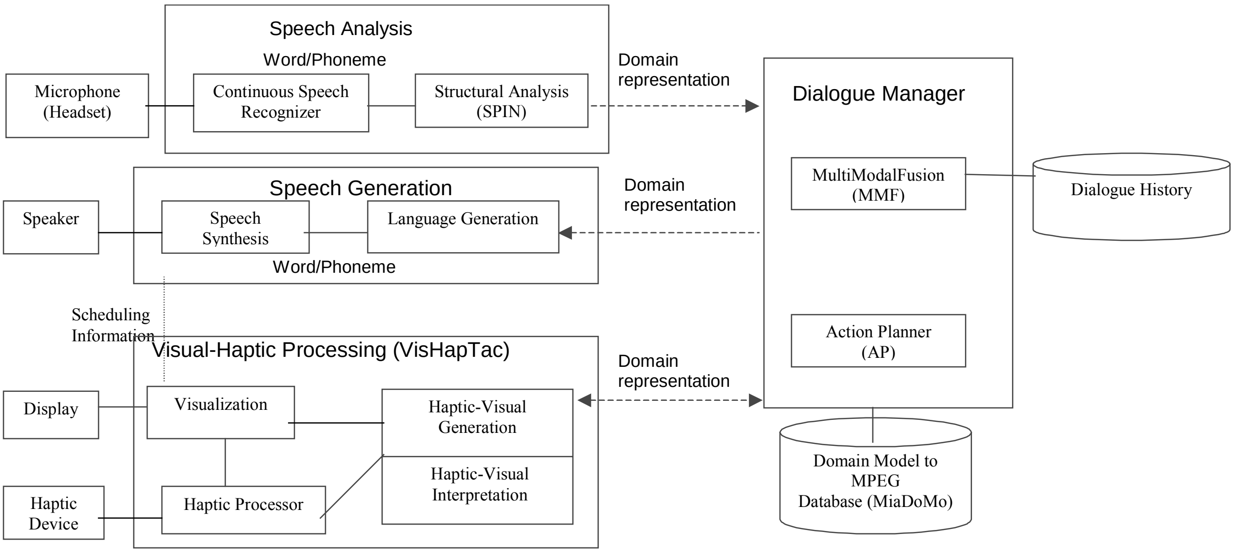


Figure 1: MIAMM System Architecture

For all these reasons, we concentrate on a model for reference resolution that attempts to overcome the diversity of resolution mechanisms. It is based on the fundamental assumption that all reference (independent of the type of referring expression) is accomplished via access to and restructuring of *Reference Domains (RD)* rather than by direct linkage to the entities themselves. It includes the same updating mechanisms for both discourse and perceptive information and is intended to be predictive in both language analysis and language generation – a particularly important feature for a model to be integrated into a dialogue system framework.

There have been efforts towards characterizing relationship between referential and discourse structures (Schauer, 2000; Seville, 1999). However, there does not seem to be much work on how the reference model could be used for robust multimodal dialogue processing. In this paper, we advocate a model which closely hinges the dialogue model on the reference model. The rest of the paper is structured as follows. The MIAMM framework is briefly described in Section 2. Section 3 describes the basic cognitive grammar hypothesis that we incorporate in our reference model. Section 4 describes our reference and dialogue framework in detail.

## 2 MIAMM Framework

The main objective of the MIAMM[1] project (Reithinger et al., 2002) is to provide an integrated and comprehensive framework (cf. Figure 1) for the design of modular multimodal dialogue systems that allow fast and natural access to multidimensional application databases. The MIAMM emulator integrates a speech interface with a graphic interface, which consists of haptic-tactile buttons and a visual display. The user can interact with the device using speech and/or the haptic buttons to search, select and play tunes from an underlying musical database

The application domain model, MiaDoMo realizes an intelligent uniform interface between the dialogue module and the various musical databases to be accessed for content selection. Requests could be as simple as “some country music” or as complex as “the soprano-alto duet piece by Charpentier” or “some Mozart-style happy orchestral music”. Essentially, the domain model is multidimensional in the sense that the application objects can have associated attributes along multiple discrete, as well as continuous dimensions. For example, information related to a musical band can be stored in the form of discrete dimensions such as band name, member artist names, genre objects. A main task of the visualisation functionality is to make it easy for the user to navigate in the visualisation using speech, pointing and haptic interaction., various albums produced, etc. and/or in the form of continuous dimensions such as temporal duration. Therefore, a query to MiaDoMo results in an information matrix which resides in the dynamic memory of the Visual-Haptic (VisHapTac) processor. The various dimensions of the data model as represented in the information matrix define the visualisation space

[1] www.miamm.org

in which the users can navigate. The restrictions of the display require for condensation and concentration of the visible objects. A main task of the visualization functionality is to make it easy for the user to navigate in the visualization using speech, pointing and haptic interaction.

The MIAMM framework allows vague and incomplete multimodal inputs as well as information aggravation and re-structuring along various dimensions. Such a complex scenario entails heavy usage of referring expressions, anaphoric as well as deictic. Apart from the common indefinites, definites and demonstratives, bridging expressions such as "the swing", referring to the musical city currently focussed on the map visualization, are quite frequent. At the dialogue level, the multimodal utterances are terse and potentially ambiguous in nature. However, for the sake of simplicity and naturalness, we are mostly concerned with task-oriented mixed-initiative dialogues.

## 3 From Cognitive Grammar to Reference Resolution

Cognitive Grammar (Langacker, 1986, 1991) situates linguistic competence within a more general framework of cognitive faculties by assuming that language is neither self-contained, nor describable without reference to cognitive processing. As a fundamental assumption of Cognitive Grammar, sense cannot be represented by logical forms. The first reason is that semantic structures are characterised relative to open-ended knowledge systems. The second reason is that an expression's meaning cannot be reduced to an objective characterization of the situation described: equally important for semantics is how the speaker chooses to construe the situation. Therefore, Cognitive Grammar assumes conceptual rather than truth-conditional semantics, considering that meaning consists of a process of conceptualisation, i.e. activation of conceptions in a hearer's mind.

More precisely, the conceptualization of an expression is said to impose a particular image on its domain, where *domain* is defined as a cognitive structure which is presupposed by the semantics of an expression. As an example, the definite in "Select **the first song** and play it" presupposes as its domain an ordered set of songs. In MIAMM, this domain could be a visual representation for a list of songs, displayed on the screen.

The particular image imposed by the expression results in profiling a substructure of the domain, namely that substructure which the expression designates. In the aforementioned example, this would be the part of the list representing the intended referent. As a result of the interpretation process, the profiled subpart of a domain is hypothesized to be more prominent or more activated than the rest of the domain.

Since an expression is always said to be interpreted within a limited domain, our context model – or *multi-modal dialogue history* – is built upon the notion of *reference domains* (Salmon-Alt, 2001). These domains are identifying representations for subsets of contextual entities to which it is possible to refer, including individual objects and collections of objects. A first important point is that these domains are not primarily linguistic constructs, since they are created and updated dynamically *via* discursive information, visual perception, haptic events and conceptual knowledge. The second important characteristic of our domains is that they present the entities from a particular cognitive viewpoint, for which we assume in the following that it is the most likely to be activated for referential access to the entity. In this sense, our model predicts optimal use of referring expressions, whereas fallback strategies can always be applied to failing interpretations. Taking up again the previous example "Select **the first song** and play it", we will, for instance be able to predict that in this context of an activated domain of *songs*, a one-anaphora such as "Delete **the last one"** has to be interpreted preferentially as referring to a song, even if the visual interface displays at the same time, for example, a list of song authors.

Based on a context modeled by *Reference Domains*, the interpretation process for referring expressions is seen as an extension of the hypotheses of Cognitive Grammar about the representation of grammatical meaning in terms of abstract symbolic schemas. More precisely, we assume that the semantics of a given expression can be represented by a schema which corresponds to an underspecified reference domain. This underspecified domain is calculated by combining abstract schemas for nouns, modifiers and determiners, taken from a lexical knowledge base.

The interpretation process consists of two steps:

1) The underspecified domain has to be matched to suitable *RDs* from the context model;

2) A restructuring operation updates the domain by profiling a substructure of the same domain as the referent.

An interesting point here is the fact that the same mechanism acts for linguistic expressions and for gestures: for example, a pointing gesture to a particular CD cover on the screen highlights this entity as the referent, whereas the other CD covers are considered as the reference domain. In this way, an expression like "Delete **the other ones**" will then be interpreted as referring to the rest of covers, even if there are other visual elements on the screen, (for example, portraits of song writers).

# 4 From Reference Resolution to Dialogue Management

## 4.1 Dialogue Functional Specification

In line with the Cognitive Grammar hypothesis, we assert that the dialogue functional behavior is essentially guided by the underlying processes which a multimodal system undertakes for input interpretation, fusion, fission and output generation. The system should be able to map the communicative behaviors within a multimodal utterance onto the communicative functions and vice versa (Cassell, 2001). This requires specification of how the interpretation or generation of the utterance changes the system's information state such as domain model, discourse model, user model, and task model (Bunt et al., 2002). Essentially, an efficient dialogue management strategy should take motivation from question-answering and text summarization approaches (Hovy et al., 1998).

The interpretation of a multimodal input, such as a spoken utterance combined with a haptic gesture, will often have stages of modality-specific processing, resulting in representations of the semantic content of the interactive behavior in each of the separate modalities involved. Other stages of interpretation combine and integrate these representations, and take contextual information into account, such as information from the domain model, the discourse model and the user model. Therefore functionally, the multimodal dialogue strategy ought to be incremental so as to account for low-level modality processing as well as high-level unified semantics.

## 4.2 Underspecified Semantics

In MIAMM, we adhere to multi-level approach to semantics. Various modalities and their respective functional behaviors vary significantly as regards to the distribution of semantics across the modality channels. Besides, the modularity constraint within the system architecture (cf. Figure 1) provides that every module does not have access to every available static or dynamic knowledge resource. Essentially, due to these reasons any conventional ambiguity resolution/multiple hypothesis algorithm (Alexandersson, 2001) will lead to various possible readings, resulting in the combinatorial explosion problem. Therefore, we resort to the Underspecification (van Deemter and Peters, 1996; Pinkal 1999) approach towards semantic specification. The choice fits perfectly with our integrated framework of reference resolution and incremental dialogue processing as our approach tries to specify the multimodal semantics in a context, which builds up incrementally.

The context-independent syntactic-semantic representations from SPIN (cf. Figure 1) and visualization representations from VisHapTac are encoded in MMIL (MultiModal Interface Language, Romary 2002). MMIL serves as the central representation format within the MIAMM architecture as it accounts for the transmission of data between the dialogue manager and both the multi-modal inputs and outputs and the application. It also forms the basis for the content of the dialogue history in MIAMM, both from the point of view of the objects being manipulated and the various events occurring during a dialogue session. MMIL incorporates FOL-type binary predicate-based semantics into a flat XML structure, maintaining two primitive levels of representation – events and participants – (Kumar et al., 2003). For example, the underspecified semantic representation for a simple referring expression, "the song" encoded in MMIL, will look like as follows:

```
<participant id = "p1">
        <objType>tune</objType>
        <individuation>singular</individuation>
        <refType>definite</refType>
        <refStatus>pending</refStatus>
</participant>
```

## 4.3 Specifying Semantics, Incorporating Pragmatics and Resolving References

There have been various attempts towards characterizing reference resolution models such as coreference model (Tetreault, 2001), sense model (Hobbs, 1988), extensions model (Allen et al., 1996). However, none of these models account for the complete range of reference phenomena found in conversational language. Byron (2002) construes the ideal resolution model as a mapping from *initial* referring expression (RE) descriptions to *final* logical term descriptions. The underspecified representations as described in the previous section resemble the context independent description structures. However, an important distinction is that our aim is not to construct *final* logical term descriptions. What we are aiming at instead is the *maximum*[2] possible resolution, which may not always result into *final* logical forms, at the same time maintaining certain degree of underspecificity as it is necessitated for the continuity of the dialogue progress. For example, the user can command, “play me the pop one”, while there exists only jazz tunes in the context. In this case an effort to construct a *final* logical term for the referring expression “the pop song”, would not lead to the desired response from the system. Instead, in such cases of perceptual mismatch, we maintain the level of underspecificity, while informing the Action Planner (AP) about the level of feature mismatch. AP then initiates proper meta-communication with the user, presenting him with the choice to play the jazz tunes. MMIL also has this nice additional feature of percolating lexical information at various levels of processing, which provides for the fall back strategy of lexical semantic specification. Essentially, our algorithm models resolution as a contextual (dynamic) and conceptual (static) mapping from the underspecified RE representations to a maximally specified cognitive description, which in our case are *Reference Domains* (cf. Figure 2).

The semantic representations as introduced in the previous section are assimilated into the type-theoretic models of RDs (Salmon-Alt, 2001). These domains are minimally identified by an *Id*, which serves as a domain index and *type*, which is extracted from the conceptual hierarchy accessible to the dialogue manager. The important features of a reference domain are its partitions, which reflect the cognitive viewpoint towards the domain. More specifically, these partitions in conjunction with focus and salience criteria define the accessibility criteria for appropriate referent profiling. The partition types are discursive cues such as role properties, perceptual information such as Haptic SelectionStatus, and/or conceptual cues such as domain level information.

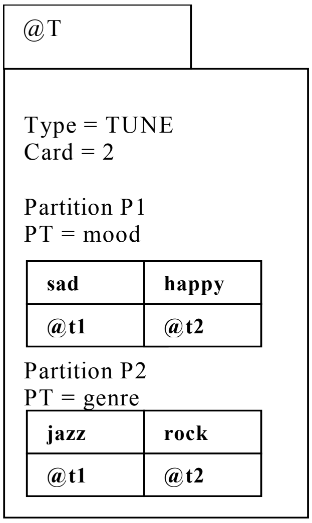


Figure 2 – Reference domain for a group of two tunes (@T)

The data structural representation for RD has the following form:

```
<ParentRefDomainObject>
      domainId
      domainType
      cardinality
      <Partition P1>
            PartitionTypeAttribute1
            <SetOf<PartTypeVal -
            ChildRefDomainObject>>
      <Partition P2>
            PartitionTypeAttribute2
            <SetOf<PartTypeVal -
            ChildRefDomainObject>>
      …
```

It is important to note here that depending upon the current discursive or perceptual state, there might not exist any partition within a RD. This is crucial so as to limit the accessibility of possible referents, as well as to provide fine-grained semantic resolution. During the dialogue progress, if certain partition is rendered out of scope by the resolution algorithm, it is deleted so that it does not lead to wrong extraction of the referent. Similarly, a tune set might not have any partition to begin with. However, by the usage of a discursive trigger like “the one by Madonna”, it can be further resolved at the level of artist resulting in a new partition.

The MultiModalFusion (MMF) component of the Dialogue Module, maintains an incremental dialogue state by constructing underspecified RD representations for the referring expressions within the current utterance and by composing

[2] *maximality* refers to the most basic level of attributes associated with any object.

them with the existing context structure. The typical compositional operations are carried upon in the following stages:

***1) Grouping:*** The underspecified RDs within a multimodal utterance are first evaluated for grouping. Based upon discursive triggers such as prepositions, conjunctions, disjunctions and/or perceptual triggers such as haptic gesture resulting in an item selection, RDs are grouped together if they match type, cardinality and temporal proximity constraints. For example, for the utterance “download the one by Madonna and this one + [haptic selection]”, to begin with, the interpretation process results in 3 underspecified RDs: first for the definite RE, second for the demonstrative RE and third for the haptic event. Using the demonstrative cue and the temporal proximity, the resolution groups 2nd and 3rd RD, resulting in a further-specified RD, which is then composed with the 1st RD owing to the discursive trigger, i.e. the conjunctive *and*. The grouped RD has zero or one partition depending upon whether the 2 RDs have the same or different artists. It is to be noted that in this particular case the demonstrative is resolved at an early stage while the definite is still pending.

***2) Assimilation:*** Depending upon the type of referring expressions, the context model tries to assimilate the underspecified RDs in differing but coherent ways (Salmon-Alt, 2001; Kumar, 2002). Essentially, owing to structural recursive- ness and compositional nature, these RDs lead to a directed acyclic graph like context structure. The leaf RDs are at the level of maximum poss- ible resolution at any stage of the dialogue processing. Firstly, a suitable node in the graph is selected. This selection is usually guided by two algorithms: the first one goes through the contextual domains, according to their activation level and starting with the most activated one, while the second one is intended to test the compatibility depending upon type, individuation, partition types etc. Secondly, the intended referent is extracted by profiling the sub-structure, resulting in re-structuring of the domains. For example, within an existing context of a tune list on the graphic display, the reference interpretation process for the speech utterance, “ play the third song”, would involve finding a node within the context structure representing an RD of tunes and having an index based partition of the member tunes.

In the following section, we provide a sample dialogue processing illustrating how this reference mechanism is useful for the MIAMM multimodal dialogue system framework.

### 4.4 Facilitating Dialogue Management

The following is a typical mixed-initiative dialogue within our framework:

(1)

U[1]: Play me the list I listened to this morning.

S[2]: Which one do you want to listen? + [displays a list of 2 tune-list items]

U[3]: the first one/ the one by Madonna.

S[4]: [plays the tune list]

U[5]: Save it/ * Save this/ *Save the list.

S[6]: [Saves]

To begin with, there exists a multimodal dialogue history for the system’s perusal in the form of a stack of context structures, while the discursive and perceptual current context is empty. The definite RE in U[1], “the list” gives rise to an underspecified RD of type /entity-list/ (say, @L1). As @L1 holds predicative relationship with a past event, the reference interpretation algorithm evaluates existing context structures within the dialogue history to *assimilate* @L1, subject to the identification of a unique RD matching the type and predicative constraints. In this case, the system is able to locate 2 RDs (say @tL1 and @tL2) of type /tune-list/, which is subsumed by the type /entity-list/ in the conceptual hierarchy. Also, these 2 RDs match the predicative relational constraint as imposed on @L1 by U[1]. However, the possible referent is not unique, as it should be for identifying the target referent for a definite RE.

It is important to note that even though a referring action is intended to accomplish the referential communicative goal (Dale et al., 1995), i.e., to help the hearer in identifying the target referent, it might not always lead to the hearer identifying the referent as conceived by the speaker (Poesio et al., 2000). This is partly, because each agent involved in a dialogue can have potentially disparate knowledge resources and cognitive descriptions at his disposal. Goodman (1986) characterizes various possible causes of miscommunication leading to an inappropriate or sub-optimal usage of referring expressions.

Within a multimodal setting, it is quite natural that miscommunications are frequent as it is strongly coupled to affordances (or rather, mis-affordances) of various modalities, as well as to the complexity of the multimodal context. Therefore, in order to impart robustness to any such system, it is imperative that the dialogue progress is incrementally enhanced in a non-monotonic way. In case of dialogue (1), the system retrieves the tune-lists which are in a predicative relation with any past event occurring /this morning/[3]. The RD @L1 thus obtained, is partitioned along the partition type of /event-Type/. The RD within this partition corresponding to the partTypeValue, /played/[4] is profiled as the possible referent and a list of 2 items is displayed along with an information-seeking speech response. This also brings the sub-structured partition under focus, implying that the objects within this partition are most likely to be referred by the user in the subsequent utterances, provided the dialogue continuity is maintained (Brennan, 2000).

In U[3], the user makes the referring action depending upon which attribute is in his perceptual context i.e. either indexicals such as “the first one”, the domain attributes such as “the one by Madonna” or deictic such as [a haptic selection]. While in other scenario, say (2), the user after getting this response from the system, can recognize his mistake, rephrasing his actual request in U[3] as, “No, the one I downloaded”.

Our reference and context model captures these dialogue intricacies in a coherent manner. In the first scenario, the system builds an under-specified RD for the RE, say “the one by Madonna”, having /entity/ as type and /Madonna/ as an absolute modifier – a domain attribute. The activated partition of @L1, contains objects which match in type[5] with the underspecified domain. If there exits any tune-list by Madonna within this partition, the partition is further partitioned into a new partition, profiling the RD having Madonna as an artist, as the identified referent and bringing it under focus. In the other scenario, the RD corresponding to the partTypeValue, /downloaded/ is profiled and focussed. Besides, it is also evident that in U[5], the usage of pronominal is the most optimal one, as a pronominal RE marks monotonic dialogue continuity.

Thus structurally, the notion of reference domains allows transversal as well as horizontal access and update mechanisms. This enables the reference model to mimic the non-monotonic nature of dialogues, resulting into a unified description as provided by multiple modalities at the same time maintaining unified multimodal semantics.

## 5 Conclusions

We have outlined a reference resolution mechanism based on the cognitive grammar approach. The discussion is by no means exhaustive and complete owing to space limitations. Besides, our main objective here is to illustrate how this mechanism can be seamlessly integrated into a dialogue framework especially in a multimodal setting. Also, we argue that the particular choice of reference mechanism does have some important implications for dialogue management. In this light, it is agreeable that the reference model can be used towards building and updating dialogue structure (Seville 1999). Still, it remains to be seen how this model handles further complicated dialogue issues such as *conceptual entrainment* (Brennan 2000), use of absolute vs relative modifiers, mutual grounding etc. As a future activity, we plan to take up these issues by subsequent evaluation of our algorithm with respect to various reference phenomena encountered in a multimodal dialogue system framework.

## 6 References

Alexandersson J. and Becker T. (2001). Overlay as the Basic Operation for Discourse Processing in a Multimodal Dialogue System In: *Proceedings of the IJCAI Workshop ``Knowledge and Reasoning in Practical Dialogue Systems*, Seattle.

Allen J., Miller B., Ringger E., and Sikoski T. (1996). A robust system for natural spoken dialogue. In *Proceedings of the 14th Annual Meeting of the Association for Computational Linguistics* (ACL ‘96), June.

Asher N. (1993). *Reference to Abstract Objects in Discourse.* Kluwer Academic Publishers. Dordrecht, Boston, London.

Bos J., Mineur A-M., and Buitelaar P. (1995). *Bridging as Coercive Accommodation.* Technical Report Number 52, Department of Computational Linguistics, Universität Saarbrücken.

[3] We follow similar mechanism for temporal reference resolution

[4] user request for /listen/ corresponds to system action of /play/

[5] based on the subsumption criteria.

Brennan S. (2000). Processes that Shape Conversation and their Implications for Computational Linguistics. In *38th Annual Meeting of the Association for Computational Linguistics*, Hong Kong, China, 1-8 October 2000.

Bunt H., and Romary L. (2002). Towards Multimodal Content Representation. In *International Standards of Terminology and Language Resources Management*, LREC 2002, Las Palmas (Spain).

Byron K. D. and Allen F.J. (2002). What's a Reference Resolution Module to do? Redefining the Role of Reference in Language Understanding Systems. In *the proceedings of the 4th Discourse Anaphora and Anaphor Resolution Colloquium*.

Cassell J. (2001). "Embodied Conversational Agents: Representation and Intelligence in User Interface" *AI Magazine*, Winter 2001, 22(3): 67-83.

Dale R. (1992). *Generating Referring Expressions: Constructing Descriptions in a Domain of Objects and Processes.* MIT Press.

Dale R. and Reiter E.(1995). Computational Interpretation of the Gricean Maxims in the Generation of Referring Expressions. *Cognitive Science*, 19(2):233—263.

Dale R. and Reiter E. (1996). The Role of Gricean Maxims in the Generation of Referring Expressions. *Proc. of the 1996 AAAI Spring Symposium on Computational Models of Conversational Implicature,* Stanford University, California.

Deemter v. K. and Peters S. (1996). *Semantic Ambiguity and Underspecication*. Stanford: CSLI.

Goodman B.A. (1986). Reference Identification and Reference Identification Failures. *Computational Linguistics*, 12:273-305.

Grosz B.J. and Sidner C. (1986). Attention, Intention and the Structure of Discourse. *Computational Linguistics,* 12, 175-204.

Grosz B.J., Joshi A.K., and Weinstein S. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics,* 12(2), 203-225.

Hahn U. and Strube M. (1997). Centered Segmentation: Scaling Up the Centering Model to Global Discourse Structure. *Proc. of EACL/ACL'97,* Madrid, 104-11.

Hobbs J. (1986). Resolving pronoun reference. In *Readings in Natural Language Processing*, pages 339-352. Morgan Kaufmann.

Hovy E.H. and Lin C.Y. (1998). Automated Text Summarization in SUMMARIST. In M. Maybury and I. Mani (eds), *Advances in Automatic Text Summarization.* Cambridge: MIT Press.

Kamp H. and Reyle U. (1993). *From Discourse to Logic.* Kluwer Academic Publishers. Dordrecht, Boston, London.

Kumar A. (2002). *Dialog Module Technical Specification*. Project MIAMM – Multidimensional Information Access using Multiple Modalities. EU project IST-20000-29487, Deliverable D5.1. LORIA, Nancy.

Kumar A. and Romary L. (2003). A Comprehensive Framework for Multimodal Meaning Representation. In *International Workshop on computational Semantics (IWCS-5)* , Tilburg, Netherlands.

Langacker R.W. (1986). *Foundations of Cognitive Grammar*. Stanford University Press. Stanford.

Langacker R.W. (1991). *Concept, image, and symbol : the cognitive basis of grammar*. Mouton de Gruyter.

McCoy K.F. and Strube M. (1999). Taking Time to Structure Discourse : Pronoun Generation Beyond Accessibility. *Proc. of the 21th Annual Conference of the Cognitive Science Society.* Vancouver, Canada, Aug. 19-21, 1999.

Pinkal M. (1999). On Semantic Underspecification. In: Bunt, H./Muskens, R. (Eds.). *Proceedings of the 2nd International Workshop on Computational Linguistics (IWCS 2).*

Poesio M. and Reyle U. (2000). Underspecification in Reference: Some Evidence from Corpora, *Proc. of the KR-2000 Workshop on Semantic Approximation, Granularity, and Vagueness*, Breckenridge, April.

Reithinger N., Lauer C., and Romary L. (2002). MIAMM: Multidimensional Information Access using multiple modalities, In *Proceedings of the International CLASS Workshop on Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems*, 28-29 June, Copenhagen, Denmark, 2002.

Romary L. (2002). *MMIL technical specification*. Project MIAMM – *Multidimensional Information Access using Multiple Modalities.* EU project IST-20000-29487, Deliverable D6.3. LORIA, Nancy.

Salmon-Alt S. (2001). Reference Resolution within the Framework of Cognitive Grammar. *International Colloquium on Cognitive Science*, San Sebastian, Spain, May 2001.

Schauer H. (2000). Referential Structure and Coherence Structure in *Proceedings of the TALN 2000, 7e conférence annuelle sur le traitement automatique des langues naturelles*, 16-18 October, Lausanne, Switzerland, p.327-336.

Seville H. and Ramsay A. (1999). Reference-based Discourse Structure for Reference Resolution. *ACL'99 Workshop on Discourse Structure and Reference*. University of Maryland, June.

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

Vieira R. and Poesio M. (2000). An Empirically-Based System for Processing Definite Descriptions, *Computational Linguistics*, 26/4. 525-579.

# Anaphora Resolution in ExtrAns

**Diego Mollá, Rolf Schwitter**
Centre for Language Technology,
Macquarie University,
Sydney, Australia
{diego,rolfs}@ics.mq.edu.au

**Fabio Rinaldi, James Dowdall, Michael Hess**
Institute of Computational Linguistics,
University of Zurich,
Zurich, Switzerland
{rinaldi,dowdall,hess}@cl.unizh.ch

## Abstract

The true power of anaphora resolution algorithms can only be gauged when embedded into specific Natural Language Processing (NLP) applications. In this paper we describe the anaphora resolution module from ExtrAns, an answer extraction system. The anaphora resolution module is based on Lappin and Leass' original algorithm, which used McCord's Slot Grammar as the inherent parser. We report how to port Lappin and Leass' algorithm to Link Grammar, a freely available dependency-based parsing system that is used in a range of NLP applications. Finally, we report on how the equivalence classes that result from the anaphora resolution algorithm are incorporated into the logical forms used by ExtrAns.

## 1 Introduction

Research in anaphora resolution has been very intense during several periods in the past and present. Several anaphora resolution modules have been implemented in the past (Mitkov et al., 2002; Grosz et al., 1995; Kennedy and Boguraev, 1996), and a number of NLP applications use anaphora resolution components. In particular, several question-answering systems have implemented anaphora resolution in several ways (Jong-Hoon et al., 2001; Vicedo and Ferrández, 2000).

In this paper we consider the role of Anaphora Resolution in ExtrAns, a Question Answering system targeted specifically at technical documentation. After an initial application to the Unix manpages (Mollá et al., 2000), ExtrAns was used in the Aircraft Maintenance Manual (AMM) of the Airbus A320 (Rinaldi et al., 2002), and currently we are targeting the Linux HowTos. ExtrAns translates documents and questions into a flat semantic representation using a comprehensive linguistic analysis. The system resolves pronominal references, disambiguates ambiguous structures, and includes modules capable of dealing with peculiarities of technical terminology (Rinaldi et al., 2003; Dowdall et al., 2002). ExtrAns derives the answers to questions by logical proofs from the document collection. A schematic representation of the architecture of the system can be seen in Figure 1.

Document sentences and questions are syntactically processed by Link Grammar, a parsing system that consists of a robust dependency-based parser and a wide-coverage grammar for English (Sleator and Temperley, 1993). In the current version of ExtrAns, anaphora resolution is restricted exclusively to pronominal cases since it is less clear how the explicit resolution of definite noun phrases and especially associative references might improve the answer extraction process without making use of complex external resources (e.g. domain/world knowledge, ontologies).

The resolution algorithm we are going to present in this paper is an adaptation of a purely syntactic approach. The theory behind the anaphora resolution module of ExtrAns is based on (Lappin and Leass, 1994), but it has been fine-tuned in several ways for the answer extraction task. Its major advantage

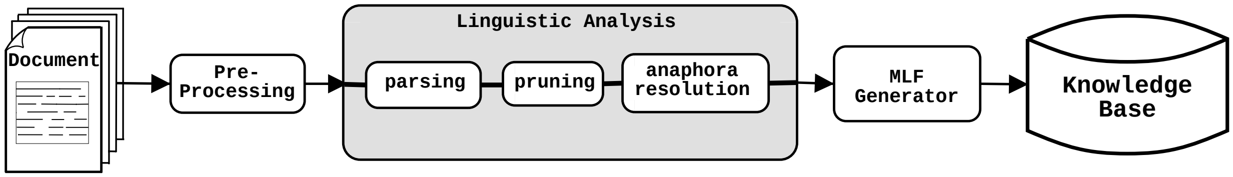


Figure 1: Schematic view of the Architecture of ExtrAns

against other algorithms is that it can be adapted to use the resources that are available in a state-of-the-art answer-extraction system such as ExtrAns. In particular it uses syntactic information, such as that produced by Link Grammar, rather than resorting to parser-free approaches like (Kennedy and Boguraev, 1996). We will see in the following sections that the type of syntactic information required for the anaphora resolution algorithm is different from the one provided by Link Grammar, but it is possible to largely re-create the necessary information.

Another advantage of the use of Lappin and Leass' algorithm is that it does not use semantic information, in contrast with (van der Sandt and Geurts, 1991; Pinkal, 1991), nor real-world knowledge, in contrast with (Hobbs, 1978). Nor does it model intentional or global discourse structure, in contrast with (Grosz, 1981). Thus, the result is computed in relatively short time and uses less resources.

Lappin and Leass' anaphora resolution model has the following components (Lappin and Leass, 1994, 536):

- An intrasentential syntactic filter for ruling out anaphoric dependence of a pronoun on an NP on syntactic grounds.
- A morphological filter for ruling out anaphoric dependence of a pronoun on an NP due to non-agreement of person, number, or gender features.
- A procedure for identifying pleonastic (semantically empty) pronouns.
- An anaphor binding algorithm for identifying the possible antecedent binder of a reflexive or reciprocal pronoun within the same sentence.
- A procedure for assigning values to several salience parameters for an NP.
- A procedure for identifying anaphorically linked NPs as an equivalence class.
- A decision procedure for selecting the preferred element of a list of antecedent candidates for a pronoun.

Each one of these steps has been adapted to ExtrAns, as we will discuss in the following sections.

## 2 Anaphora Resolution in ExtrAns

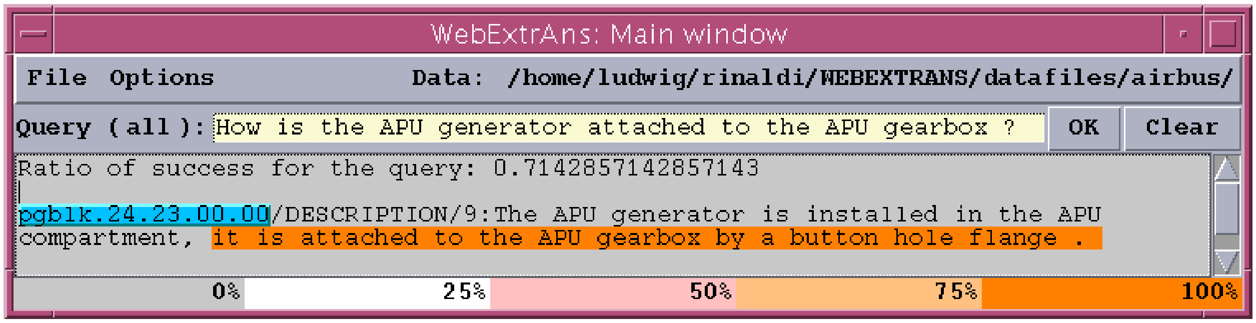
### 2.1 Emulation of the Slot Grammar

Lappin and Leass' algorithm relies heavily on the output of the parser, a Prolog clausal implementation of Slot Grammar (McCord et al., 1992). The resulting syntactic analysis includes the head-argument and head-adjunct relations of the phrase structure that the Slot Grammar assigns to the sentence or phrase. These relations (also called **slots**) include "subject", "agent", "object", "indirect object", and "prepositional object".

Since Slot Grammar is dependency-based, it is possible to approximate its behaviour by means of the dependency structures returned by Link Grammar. However, Link Grammar does not show the direction of dependencies explicitly. This information is easily recovered by examining the link types and occasionally some specific local arrangements of the link structures, and it has been added in a post-processing module of ExtrAns' parser (Mollá et al., 2000).

Given a dependency structure such as the one provided by ExtrAns' parser module, we can explore the labels of the links to compute the relations listed above: [1]

[1] It is possible that the algorithm tries to find the value of the slot of a particular word several times (if the word is a candidate for coreference for several pronouns, for example). To speed up processing, the algorithm first checks if the information has already been computed. If not, the slot is computed and the result is stored in the noun-related data structure and returned.



Data: /home/ludwig/rinaldi/WEBEXTRANS/datafiles/airbus/

Query (all): How is the APU generator attached to the APU gearbox ?

```
Ratio of success for the query: 0.7142857142857143

pgblk.24.23.00.00/DESCRIPTION/9:The APU generator is installed in the APU
compartment, it is attached to the APU gearbox by a button hole flange .
```

Figure 2: An example of Anaphora Resolution in ExtrAns

**Subject (`subj`).** Check if the link is `S`[2] [subject] or `SI` [inverted subject]. Alternatively, check if the word is actually the subject of a relative clause (links `B` [external subject/object] and `RS` [subject relative pronoun] in a specific pattern).

**Agent (`agent`).** Check if the word is the main noun of a PP headed by *by*, and the sentence is passive.

**Direct (`obj`) and indirect object (`iobj`).** This is decided by the link `O` [object]. The object closest to the verb is the direct object, and the rest are indirect objects.

**Prepositional object (`pobj`).** The object of a PP can be found by checking if the link is `J` [object of a PP] or `U` [idiomatic noun].

### 2.2 The syntactic filter on pronoun-NP coreference

The filter uses a set of syntactic rules to prune and remove anaphoric dependencies which are syntactically impossible. Lappin and Leass use the following terminology:

**Immediate containment.** A phrase $P$ is immediately contained in a phrase $Q$ iff $P$ is either an argument or an adjunct of $Q$. In terms of ExtrAns, $P$ is directly dependent on $Q$. Note that the algorithm must also check if $P$ or $Q$ is a member of a coordination.

**Containment.** A phrase $P$ is contained in a phrase $Q$ iff:

1. $P$ is immediately contained in $Q$, or
2. $P$ is immediately contained in some phrase $R$, and $R$ is contained in $Q$.

**Argument domain.** $P$ is in the argument domain of a phrase $N$ iff $P$ and $N$ are both arguments of the same head. Since Link Grammar does not differentiate between arguments and adjuncts, the rule is implemented so that $P$ and $N$ are both immediately contained in the same head.

**Adjunct domain.** $P$ is in the adjunct domain of $N$ iff $N$ is an argument of a head $H$, $P$ is the object of a preposition $PREP$, and $PREP$ is an adjunct of $H$. In terms of Link Grammar, this means that $N$ and $PREP$ are both immediately contained in $H$.

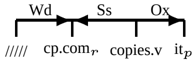
**NP domain.** $P$ is in the NP domain of $N$ iff $N$ is the determiner of a noun $Q$ and:

1. $P$ is an argument of $Q$, or
2. $P$ is the object of a preposition $PREP$ and $PREP$ is an adjunct of $Q$.

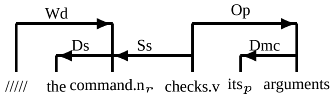
In terms of Link Grammar, the definitions of "argument" and "adjunct" merely mean immediate containment.

Now, we can say that a pronoun $P$ cannot corefer with a non-reflexive, non-reciprocal noun phrase $R$ if any of the following conditions hold:

1. $P$ is in the argument domain of $R$:

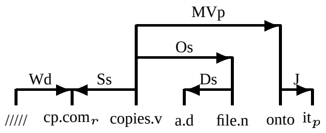


This rule does not apply when $P$ is a determiner:

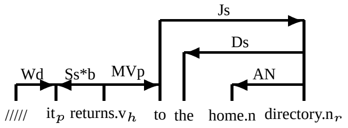


[2] See http://www.link.cs.cmu.edu/link/ for a summary of the link types used by Link Grammar.

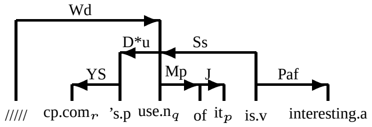
2. $P$ is in the adjunct domain of $R$:



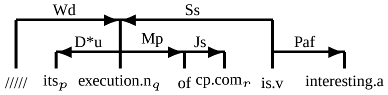
3. $P$ is an argument of a head $H$, $R$ is not a pronoun, and $R$ is contained in $H$:



4. $P$ is in the NP domain of $R$:



5. $P$ is the determiner of a noun $Q$, and $R$ is contained in $Q$, or $R$ is $Q$:



   Note that the example above is ruled out by rule 3 because of our lax definition of argument.

Apart from the original rules, we have added the following rules. A pronoun $P$ cannot corefer with a noun $R$ if:

1. $P$ is $R$. This rule is pretty obvious.

2. $P$ is contained in $R$. This rule handles some of the cases that were not ruled out by the rules above because of our particular definition of argument and adjunct dependency.

### 2.3 Tests for agreement

In the original algorithm, the agreement features of an NP are number, person, and gender. In the technical domain of ExtrAns, the pronoun resolution has been implemented only for neuter pronouns, and therefore gender and person are irrelevant. Thus, we use the number only.

To compute the number agreement of a word, we use the label of the link of the word to its head. The algorithm checks whether the link has a suffix `s` or `m` (indicating singular), or whether the suffix is `p` (indicating plural). [3]

### 2.4 Identifying pleonastic pronouns

Pleonastic pronouns are pronouns that do not carry any meaning, like in the phrase *it is likely that*. To identify them, we use Link Grammar's link labels. The following link labels indicate the use of a pleonastic pronoun:

**SF** is a special connector used to connect "filler" subjects like *it* and *there* to finite verbs: *THERE IS a problem, IT IS likely that ....*

**SFI** connects "filler" subjects like *it* and *there* to verbs in cases with subject-verb inversion: *IS THERE a problem?, IS IT likely that ...?*

One stage in the process of anaphora resolution consists in identifying all the nouns in the sentence, including pronouns. The algorithm intentionally ignores both `SF` and `SFI`, as a result pleonastic pronouns are not selected.

### 2.5 The treatment of lexical anaphors

According to Lappin and Leass, a lexical anaphor is either a reciprocal or a reflexive pronoun. The syntactic rules of coreference of lexical anaphors do not check if the coreferent nouns are incompatible. Instead, the rules check if the noun can corefer with the lexical anaphor. Two assumptions are that the noun and the anaphor appear in the same sentence, and the noun must appear before the anaphor.

To compute the possibility of coreference, it is necessary to rank the slot information of the noun and the anaphor, as follows:

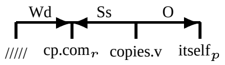
$$\text{subj} > \text{agent} > \text{obj} > (\text{iobj—pobj})$$

In other words, a subject occupies a higher argument slot than an object, and the slot position of an indirect object and a prepositional object are equivalent.

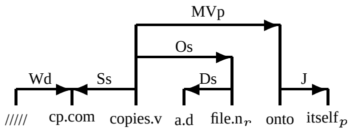
A noun $R$ is a possible antecedent binder for a lexical anaphor $P$ iff $R$ and $P$ do not have incompatible agreement features, and one of the following conditions holds:

[3] In some cases, the link label does not provide the information. When this happens, the number of the word is unspecified (using an unbound Prolog variable).

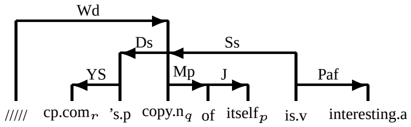
1. $P$ is in the argument domain of $R$, and $R$ fills a higher argument slot than $P$:



2. $P$ is in the adjunct domain of $R$:



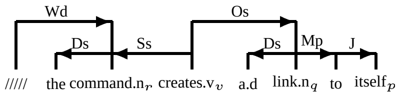
3. $P$ is in the NP domain of $R$:



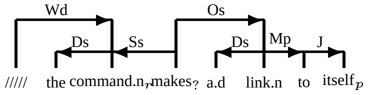
4. $R$ is an argument of a verb $V$, and there is an NP $Q$ in the argument domain or in the adjunct domain of $R$ such that:

   (a) $P$ is an argument of $Q$, or

   (b) $P$ is an argument of a preposition $PREP$ and $PREP$ is an adjunct of $Q$:



In this rule, we need to determine that $V$ is a verb. We do that by checking the tag assigned by the parser ('.v' in the examples above). If the parser gives a wrong tag or no tag at all, the verb is not recognised. For that reason, the anaphora in the following sentence is not recognised:



5. $P$ is a determiner of a noun $Q$, and:

   (a) $Q$ is in the argument domain of $R$ and $R$ fills a higher argument slot than $Q$, or

   (b) $Q$ is in the adjunct domain of $R$

### 2.6 The salience parameters

Salience is a measure that indicates how likely a particular noun is to corefer with a specific pronoun or lexical anaphor. The salience of a noun is a combination of several factors. Lappin and Leass' salience factors are based on (Alshawi, 1987), though the actual factors and values are more specific to the task of pronominal anaphora resolution.

Salience factors can be classified into two types: independent factors and dependent factors. The salience of a noun will be the sum of the weights of all the salience factors that apply to it.

### 2.7 Independent salience factors

Independent salience factors are those that do not depend on the syntactic relation between the pronoun and the noun. The following salience factors used by Lappin and Leass have been implemented: [4]

**Subject.** The noun is the head of a subject.

**Agent.** The noun is an agent in a passive sentence.

**Existential emphasis.** The noun is in an existential construction, such as in "*there are only* a few restrictions *on LQL query construction for WordSmith*".[5]

**Accusative.** The noun is the object.

**Indirect object.** The noun fills indirect object slot.

**Oblique complement.** The noun fills the prepositional object slot.

**Head noun.** The noun is not contained in another noun, using the Slot Grammar notion of "containment within a phrase" (it is either an argument or an adjunct of a noun).

**Non-adverbial.** The noun is not contained in an adverbial PP demarcated by a separator. [6]

### 2.8 Dependent salience factors

Dependent salience factors may or may not apply to a specific noun, or they may apply with a different value, depending on the syntactic relation with the pronoun. The factors implemented in ExtrAns are:

**Cataphora penalty.** Cataphora (when the pronoun appears in the sentence *before* the noun) is to be discouraged, by adding a rather large negative value to the accumulated score.

[4] Table 1 lists the weight values selected for the independent salience factors.

[5] This is checked by exploring the label of the link that connects the word to its head: it should match `O?t`, where `?` matches any link label suffix.

[6] ExtrAns uses the link labels `J` and `U` to determine if a noun is in a PP, and `Xc` and `Xd` to determine the existence of a separator.

| Factor type | Weight |
|---|---|
| Subject | 80 |
| Agent | 80 |
| Existential emphasis | 70 |
| Accusative | 50 |
| Indirect object | 40 |
| Oblique complement | 40 |
| Head noun | 80 |
| Non-adverbial | 50 |

Table 1: Weight values for independent salience factors

**Parallel roles reward.** If both the noun and the pronoun fill the same slot, the probability of being coreferent is higher.

**Recency reward.** Intrasentential coreference is to be encouraged.

The actual values of the dependent salience factors are listed in Table 2. The values are the same as in the original algorithm (Lappin and Leass, 1994) except for the case of cataphora, where we decided to increase the penalty by 100 units (up from -175).

| Factor type | Weight |
|---|---|
| Cataphora | -275 |
| Parallel roles | 35 |
| Recency | 100 |

Table 2: Weight values for dependent salience factors

### 2.9 Equivalence classes

Coreference between the pronoun and the noun is signaled by classifying both words as belonging to the same equivalence class. An equivalence class represents the set of the words that point to the same instance in the world of the application domain. [7]

Lappin & Leass' algorithm assigns a salience to the equivalence classes. How they do it, though, is not clear. In ExtrAns, the equivalence class salience is that of the class representative (for the time being, the most recent element of the equivalence class). An interesting possibility worth considering is the computation of the equivalence class salience based on when the last new word was introduced to the equivalence class. In other words, we can update the equivalence class salience on the basis of the current focus.

### 2.10 Decision procedure

With the modifications detailed in the previous sections, the implementation of Lappin and Leass' anaphora resolution algorithm is straightforward:

1. Create the list of antecedents:
   (a) Create an initial list of IDs for all the NPs in the sentence.
   (b) Compute the independent salience factors of every ID.
   (c) Group the antecedents in equivalence classes according to their coreference (for the obvious cases, like names or command arguments).
2. For every pronoun in the sentence:
   (a) Compute the list of possible antecedent candidates (the most recent of each equivalence class) $\mathcal{A}$.
   (b) Compute the list of incompatible references $\mathcal{B}$, according to syntactic and agreement grounds.
   (c) Compute the list of possible references of reflexive pronouns $\mathcal{C}$, according to syntactic and agreement grounds.
   (d) The final list of candidates is $(\mathcal{A} - \mathcal{B}) \cap \mathcal{C}$.
   (e) Compute the dependent salience factors of the final candidates.
   (f) Select the candidate with higher salience (the sum of independent and dependent salience factors). If there are several with the same salience, choose the candidate closest to the anaphor.
   (g) Add the pronoun to the equivalence class of the selected candidate.

[7] Membership of the same equivalence class is expressed in the original algorithm by means of the predicate `coref(u,y)`, which is inserted in the logical form of the sentence. For ExtrAns, however, we decided not to use additional predicates to express coreference chains. Instead, variable substitution is made. Thus, if `u` corefers with `y`, then all the predicates in the sentence that have `y` as an argument will replace `y` with `u`.

## 3 Evaluation

In order to evaluate the effectiveness of the reference resolution algorithm we selected arbitrarily 50 sentences of the Aircraft Maintenance Manual corpus which contained at least one pronoun. The total number of pronouns to be resolved was 60. We then manually examined the logical forms generated by ExtrAns to verify if the pronouns had been resolved correctly: we found that it was only 26 (43%) of the pronouns. However, the reference resolution algorithm is only responsible for some of the failures, other processing factors produced the rest. For instance, the most frequent cause of an error is a wrong parse (in a few cases the parser failed altogether). We therefore re-examined the cases were the pronoun was not resolved correctly and could filter out 18 sentences were the failure was clearly caused by reasons external to the reference resolution algorithm.

This left us with 32 sentences and 36 pronouns, of which 26 were resolved correctly. After manual examination 3 cases were then excluded because they were judged genuinely ambiguous, like the following: “Do not take the repair kit to the repair area until it is ready to use”.

So we have 26 pronouns correctly resolved out of 33, which results in an accuracy of 79%. This is lower than the result reported by (Lappin and Leass, 1994) of 86%. But, of course, our data is too small to be representative.

We then examined in detail what happened for each pronoun type:

**It:** Out of 28 anaphoric *it*, 21 were correctly solved. Thus, the result is of 75% correct resolutions.

**Its:** Of 3 cases, only 1 was correctly solved.

**Itself:** Out of 3 cases, 2 were correctly solved.

**Them:** There were 2 cases, all of them solved correctly.

Of interest is the fact that the pronoun *its* had a rather low success ratio. Lappin and Leass did not report on the success ratio of individual pronouns, and our data is too small to draw any conclusion. Further tests are necessary, and if considered necessary, the algorithm should be modified to enhance the results.

## 4 Discussion

The original algorithm can be applied to intersentential anaphora in the same way as for intrasentential. We only need to consider a few additional points. First of all, syntactic restrictions do not cross sentence boundaries. Thus, in theory, any noun in the previous sentence can corefer with a non-reflexive pronoun in the current sentence. Also, reflexives cannot corefer with nouns in previous sentences. Finally, the more sentence boundaries between the pronoun and the noun, the less likely the noun corefers with the pronoun. This is implemented by degrading (halving) the salience of the noun for every sentence boundary that is crossed.

The implementation of the anaphora resolution algorithm in ExtrAns allows for the possibility of intersentential anaphora only with the previous sentence. The reason for this restriction is twofold. First, it is very rare for a pronoun to corefer with a noun more than one sentence away. We did not find any case in the test corpus. Second, to compute the salience of a noun, it is necessary to know the syntactic structure of the sentence where the word appears, e.g. for the parallel roles reward. This means that we need to keep that information available together with other information regarding the noun. Since this information is very unlikely to be used at all, it is not practical to keep the information in the system.

ExtrAns’ semantic interpreter uses the information from the anaphora resolution algorithm to merge the variables of the logical form predicates that correspond to words belonging to the same equivalence class.

For example, without information about equivalence classes, the semantic interpreter would produce the following logical form for the sentence *“The APU Generator is installed in the APU compartment, it is attached to the APU gearbox by a button hole flange”*:[8]

```
object(APU_generator,o1,[x2]),
evt(install,e4,[a4,x2]),
object(anonym_object,o5,[a4]),
in(e4,x8),object(APU_compartment,o2,[x8]),
object(it,o3,[x1]),evt(attach,e3,[x12,x1]),
object(button_hole_flange,o4,[x12]),
to(e3,x7), object(APU_gearbox,o6,[x7]).
```

[8] See (Mollá et al., 2000) for details about the logical forms

Since the anaphora resolution algorithm groups *APU_generator* and *it* into the same equivalence class, the semantic interpreter replaces x1 with x2 and produces the following logical form:

```
object(APU_generator,o1,[x2]),
evt(install,e4,[a4,x2]),
object(anonym_object,o5,[a4]),
in(e4,x8),object(APU_compartment,o2,[x8]),
object(it,o3,[x2]),evt(attach,e3,[x12,x2]),
object(button_hole_flange,o4,[x12]),
to(e3,x7),object(APU_gearbox,o6,[x7]).
```

This way, a question like *"How is the APU generator attached to the APU gearbox?"* prompts ExtrAns to return the answer shown in Figure 2.

## 5 Conclusion

In this paper we have presented the approach adopted for Anaphora Resolution in ExtrAns, a Question Answering System specifically developed to target technical documentation. The particular nature of the domain constraints the types of anaphoras that need to be targeted.

We think that technical documentation provides an important and interesting application for real-world Question Answering systems and certainly Anaphora Resolution has an important role to play.

## References

Hiyan Alshawi. 1987. *Memory and Context for Language Interpretation*. Cambridge University Press, Cambridge.

James Dowdall, Michael Hess, Neeme Kahusk, Kaarel Kaljurand, Mare Koit, Fabio Rinaldi, and Kadri Vider. 2002. Technical terminology as a critical resource. In *International Conference on Language Resources and Evaluations (LREC-2002), Las Palmas*, 29–31 May. [9]

B. J. Grosz, K. J. Aravind, and S. Weinstein. 1995. Centering: A Framework for Modeling the Local Coherence of Discourse. *Computational Linguistics*, 21(2):203–255.

Barbara J. Grosz. 1981. Focusing and description in natural language dialogues. In Aravind K. Joshi, Bonnie L. Webber, and Ivan A. Sag, editors, *Elements of Discourse Understanding*, chapter 3, pages 84–105. Cambridge University Press, Cambridge.

Jerry R. Hobbs. 1978. Resolving pronoun references. *Lingua*, 44:311–338. Also in Grosz, Barbara J. & Karen Sparck Jones & Bonnie Lynn Webber (1986), *Readings in Natural Language Processing*, Kaufmann, Los Altos, CA.

O. Jong-Hoon, L. Kyung-Soon, C. Du-Seong, W. S. Chung, and C. Key-Sun. 2001. TREC-10 Experiments at KAIST: Batch Filtering and Question Answering. In *Proc. of TREC-10*, Gaithersburg, Maryland.

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*, Copenhagen, Denmark.

Shalom Lappin and Herbert J. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.

Michael McCord, Arendse Bernth, Shalom Lappin, and Wlodek Zadrozny. 1992. Natural language processing within a slot grammar framework. *International Journal on Artificial Intelligence Tools*, 1(2):229–277.

R. Mitkov, R. Evans, and C. Orasan. 2002. A new, fully automatic version of Mitkov's knowledge poor pronoun resolution method. In *Proc. of CICLing 2002*, pages 168 – 186, Mexico City, Februrary.

Diego Mollá, Rolf Schwitter, Michael Hess, and Rachel Fournier. 2000. Extrans, an answer extraction system. *Traitment Automatique des Langues*, 41(2):495–522.

Manfred Pinkal. 1991. On the syntactic-semantic analysis of bound anaphora. In *Proc. of the Fifth Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Berlin.

Fabio Rinaldi, James Dowdall, Michael Hess, Diego Mollá, and Rolf Schwitter. 2002. Towards Answer Extraction: an application to Technical Domains. In *ECAI2002, European Conference on Artificial Intelligence, Lyon*, 21–26 July. [9]

Fabio Rinaldi, James Dowdall, Michael Hess, Kaarel Kaljurand, and Magnus Karlsson. 2003. The Role of Technical Terminology in Question Answering. In *Proceedings of TIA-2003, Terminologie et Intelligence Artificielle*, pages 156–165, Strasbourg, April. [9]

Daniel D. Sleator and Davy Temperley. 1993. Parsing English with a link grammar. In *Proc. Third International Workshop on Parsing Technologies*, pages 277–292.

Rob A. van der Sandt and Bart Geurts. 1991. Presupposition, anaphora, and lexical content. In Otthein Herzog and Claus-Rainer Rollinger, editors, *Text Understanding in LILOG: Integrating Computational Linguistics and Artificial Intelligence - final report on the IBM Germany LILOG project*, volume 546 of *Lecture Notes in Computer Science*, pages 259–296. Springer-Verlag, Berlin.

J. L. Vicedo and A. Ferrández. 2000. A Semantic Approach to Question Answering Systems. In *Proc. of TREC-9*, Gaithersburg, Maryland.

[9] `http://www.cl.unizh.ch/CLpublications.html`

# The Contribution of Domain-independent Robust Pronominal Anaphora Resolution to Open-Domain Question-Answering

**Rebecca Watson, Judita Preiss & Ted Briscoe**
Computer Laboratory
University of Cambridge
JJ Thomson Ave
Cambridge
CB3 OFD, UK
firstname.lastname@cl.cam.ac.uk

## Abstract

We explore the performance increase that results from the use of robust domain-independent pronoun resolution as a component of an open-domain question-answering (QA) system. We describe a baseline system based on robust parsing, named entity recognition, and matching of underspecified (rMRS) semantic structures between question and putative answer sentences, and its performance on the TREC8 QA task. We derive an experimental upper bound for improvement on this task through use of a 2 sentence context window around putative answer sentences. We describe our pronoun resolver, and its integration with the baseline system. Finally, we assess the potential and actual improvement in QA given the performance of anaphora resolution on this data and methods for integration with the QA system.

## 1 Introduction

Open-domain question-answering (QA) involves matching a representation obtained from a user's question against a document collection in order to find appropriate short answers. It is intuitively clear, at least for questions which do not contain many equivalent correct answers in the collection, that intelligent use of the context around sentences which match elements of the question should improve performance. To give one example from the TREC-8 QA dataset, the question, *What country is the biggest producer of tungsten?* can be given the answer *China* using the following passage:

> The 15 countries attending the three-day annual market review, which ended yesterday, account for about 90 per cent of world trade in tungsten products. They include China, the biggest producer, which represents over 60 per cent of world trade...

However this requires the inference that China is a country and the biggest producer *of tungsten* to achieve a convincing match with the question. The first inference can be assisted by resolving the pronominal anaphor *They* to its antecedent *The 15 countries* and the second via a more complex inference from *producer* back to *tungsten products*.

We focus on the ability of *pronominal* anaphora resolution technology to provide enriched representations of sentences containing target answers which can be used as the basis for the first type of inference. We describe a baseline QA system based on robust parsing, named entity recognition, and matching of underspecified (rMRS) semantic structures between question and putative answer sentences, and its performance on the TREC8 QA task. We derive an experimental upper bound for improvement on this task through use of a 2 sentence context window around putative answer sentences. We describe our pronoun resolver and its integration with the baseline system. Finally, we assess the likely improvement in QA given the performance of anaphora resolution and method of integration into the QA system.

## 2 The Baseline QA System

Our baseline system utilses the LingoERG grammar and the LKB parser (Copestake and Flickinger, 2000) to analyse questions and construct compositional semantic representations of them using minimal recursion semantics (MRS) (Copestake et al., 1999). Robust MRS (rMRS) is a syntactic variant of MRS which supports factorization and systematic underspecification of most aspects of this semantic representation (Briscoe et al., 2002). rMRS representations are factored into a set of elementary predications (e.g. **tungsten(x1)**) and a set of (sorted) variable equality statements (e.g. **x1=x2**). Variables can be event or object arguments of predicates which can be underspecified in cases where, say, it is unclear whether a PP is adverbial or adjectival, or can be handles used to specify scope. For our experiments we used 163 of the TREC 8 questions which the LKB analysed and assigned rMRS representations.

The 1000 top-ranked documents for the TREC 8 QA task were parsed using the RASP (robust accurate statistical parsing) system (Briscoe and Carroll, 2002). The highest-ranked analysis returned by RASP for each sentence was converted to a rMRS. Many of these rMRS representations are severely underspecified because of the impoverished information about lexical semantic types yielded by this system. In addition, about 23% of grammatical relations recovered by the RASP system in highest-ranked analyses are incorrect on test data. Therefore, we can assume that a similar proportion of the elements of the rMRS recovered from the syntactic analysis are likely to be incorrect. For a small fraction of very long and complex sentences, lists, and so forth rMRSs are based entirely on the part-of-speech tag of each word as the parser timed out before returning any syntactic analysis. An example of the output of RASP is given in Figure 1. The resulting (simplified) rMRS is underspecified for the type of the (event) variable (u4), the object (ARG2) of *include*, and the coreference between *China* and the following appositive NP, etc. The baseline QA system is based on checking the graded compatibility of rMRS elements between questions

```
(|T/txt-sc1/---|
 (|T/leta_s|
  (|S/s_co_np1|
   (|S/np_vp| |They_PPHS2|
    (|V/np| |include_VV0|
      (|NP/n1_name/-|
          (|N1/n| |China_NP1|))))
   |,_,|
   (|NP/det_n| |the_AT|
    (|N1/ap_n1/-|
      (|AP/a1| (|A1/a| |biggest_JJT|))
     (|N1/n| |producer_NN1|))))
  (|Tacl/comma-e| |,_,|
   (|S/whnp_vp| |which_DDQ|
    (|V/np| |represent+s_VVZ|
     (|NP/ap2_np| (|A1/a| |over_RP|)
      (|NP/plu3|
       (|N1/num2_nms|
          (|NP/num| (|N1/n| 60_MC))
        (|N1/nms_nms| |per_NNU|
         (|N1/n_of| |cent_NNU|
          (|PP/p1|
           (|P1/p_n1| |of_IO|
            (|N1/n1_nm| |world_NN1|
                (|N1/n| |trade_NN1|
                     ))))))))))))))
GRs:
(ncsubj represent+s_VVZ which_DDQ _)
(dobj represent+s_VVZ cent_NNU _)
(ncsubj include_VV0 They_PPHS2 _)
(dobj include_VV0 China_NP1 _)
(ncmod _ producer_NN1 biggest_JJT)
(detmod _ producer_NN1 the_AT)
(ncmod _ include_VV0 producer_NN1)
(ncmod _ trade_NN1 world_NN1)
(ncmod of_IO cent_NNU trade_NN1)
(ncmod _ cent_NNU per_NNU)
(ncmod _ cent_NNU 60_MC)
(mod _ cent_NNU over_RP)
(cmod _ include_VV0 represent+s_VVZ)

rMRS:
they_rel u2, include_rel u4
ARG1  u4 u2, ARG2  u4 u7
china_rel x6, the_rel x12
biggest_rel x12, producer_rel x12
which_rel x27, represent_rel e29
over_rel e29, 60_rel u33
per_rel x35, cent_rel x37
of_rel e39, ARG2 e39 x41
world_rel x41, trade_rel x50
```

Figure 1: System Outputs

| | |
|---|---|
| rMRS | 0.472 |
| +Morph | 0.476 |
| +WordNet+NE | 0.484 |
| rMRS+Context | 0.619 |

Table 1: Baseline System(s) MRR Performance

| | |
|---|---|
| rMRS | 0.150 |
| +Morph | 0.178 |
| +WordNet+NE | 0.270 |
| +Context | 0.470 |

Table 2: Performance on Unseen Sentences

and putative answer sentences. The TREC 8 QA dataset and evaluation system was used to optimize the weights and matching criteria which determine the overall level of compatibility when matching different components of the rMRS structure. This resulted in a system which ignored elementary predications for closed class items, abstracted over subtypes of major parts-of-speech, used morphological analysis to match morphologically-related predicates (e.g. **producer** and **production**), WordNet hyponym and synonym variants of predicates (e.g. **inhabitant** ⇝ **soul**), and named entity (NE) recognition to filter out entire rMRSs which did not contain a NP of the same sort as the question. The novel element to this baseline QA system is that it utilizes two extant parsers to generate rMRS representations for questions and document sentences and uses rMRS as the prime representation level upon which all assessments of compatibility are made.

## 3 Performance on the TREC 8 QA Task

Since our baseline system was optimized on the TREC 8 data, its performance on this data is not a realistic assessment of its general utility. Table 1 gives the mean reciprocal rank (MRR) for the baseline rMRS matching system, the baseline system with morphological analysis, and the baseline system with WordNet and NE filtering. The final column gives the results for our simplest baseline system with the addition of two sentence contexts to either side of the matching sentence, where all 5 sentences are submitted to the evaluation system[1]. It is clear from these results that, whilst the use of morphological and semantic relationships (from WordNet) and semantic filtering via NE sorts does improve performance, a far more dramatic improvement is possible by inclusion of further context. Note that submitting the entire context to the evaluation code is a crude way of assessing an upper (experimentally-derived) bound on correct exploitation of this context in the 50byte task. Assuming 100% exploitation of context in this way reduces the number of unanswered questions by about 33%.

We have also tested various variants of our baseline system on a small sample of unseen data from the TREC 9 QA track and results for these 10 sentences do support the conclusion that the matching criteria and weights selected are valid – see Table 2. If we could effectively exploit context to derive the correct short answer, then this would improve the performance of the system significantly and this would produce a bigger increase in performance (potentially) than focussing on variants of predicates in matching.

## 4 The Utility to QA

Analysis of the top 5 matching contexts returned by the baseline system revealed that there are 1041 third person pronouns in the contexts (roughly 1.2% of the total number of words). This suggests that anaphora resolution might be able to enrich the representation of the highest-matching sentence so that the correct short answer could be directly extracted via this sentence. For instance, returning to the example in section 1, resolving *They* to *The 15 countries* would license the addition of an equality statement between the rMRS variables associated with these two NPs, and this would yield an enriched rMRS for the sentence containing *China* pointing to an elementary predication for **countries**. Accurate anaphora resolution will improve the match, but there remain further inferences to be made before we can guarantee that this sentence will become the highest-ranked match to the question rMRS.

We evaluated all the matching contexts, given the baseline rMRS system, to determine whether anaphora resolution would potentially improve performance by increasing the compatibility of ques-

[1] Morton (2000) reports that a look-back of 2 sentences makes the antecedent of a pronoun accessible 98.7% of the time.

| | |
|---|---|
| intraP | 0.11 |
| interP | 0.04 |
| interD | 0.13 |
| contx+ | 0.14 |
| contx- | 0.10 |

Table 3: Proportions of Contexts Improvable

tion rMRS and the enriched rMRS of the sentence containing the answer in the context. Table 3 classifies the proportion of contexts in which resolution of intrasential pronominal anaphora (intraP), intersentential pronominal anaphora (interP), and intersentential definite description (interD) anaphora could improve QA performance in this sense.[2] It also indicates where more open-ended contextual inference could, in principle, work (contx+) and where the context is not sufficient (contx-, i.e. contexts where the match to the answer is effectively spurious). The remaining proportion (0.48) of contexts not classified in Table 3 contained correct answers consisting of phrases, often appositives NPs, in the matching target sentence which also included all the information required to match the question and generate the correct short answer. This high proportion may reflect the artificial method in which the TREC 8 QA track questions were created. However, if the remaining data are representative, they suggest that anaphora resolution has a significant role to play in the exploitation of context. Ignoring the 10% of spurious contexts, anaphora resolution is potentially beneficial in two thirds of the remaining contextual cases, and about 30% of the TREC 8 questions overall. This result does not imply that perfect anaphora resolution would improve performance by one third or better – only a small number of sentences containing correct short answers would be sufficently enriched by resolving anaphors to support straightforward generation of the short answer by rMRS matching. Nevertheless, it does suggest that anaphora resolution is a necessary and potentially significant component for the effective exploitation of context in a QA system.

## 5 Robust Domain-Independent Anaphora Resolution

Accurate anaphora resolution requires access to syntactic information in order to compute non-coreference constraints and to compute the structural salience of potentially coreferential antecedents. Work by Lappin and McCord (1990) and Lappin and Leass (1994) demonstrated that reasonable performance could be achieved, given an accurate and detailed enough syntactic analysis from which predicate argument structure (or deep grammatical relations including 'understood' control relations) could be extracted. Kennedy and Boguraev (1996) demonstrated that robust anaphora resolution could be achieved, using tags encoding lexical syntactic category and surface grammatical relations to compute salience but not non-coreference constraints. Castano et al. (2002) demonstrated that an even more syntactically impovershed system only utilizing part-of-speech tags could achieve reasonable performance, particularly on anaphoric definite descriptions, if supplemented with rich domain information in the form of sortal constraints. Ge et al. (1998) developed a probabilistic approach which integrated both types of syntactic factor with many others, such as distance and mention rate of the antecedent, as well as domain-dependent lexical information. All these systems achieved accuracy rates of 70% or better. However, meaningful comparison is hard because of the slightly different evaluation schemes employed and very different datasets used.

Preiss and Briscoe (2003) describe a reimplementation of the system of Lappin and Leass (1994) (hereafter LL) using the grammatical relations (GR) output from the RASP system. The RASP system makes very little use of lexical information (unlike most statistical parsers) in an attempt to remain (initially) as domain-independent as possible. As the system returns ranked analyses, this does not preclude reranking utilizing domain-dependent (lexical) information where this is available, but our goal is to achieve useful levels of accuracy without relying on the availability of domain-specific lexical resources. We have pursued a similar approach to anaphora resolution, attempting to exploit structural information fully in order to produce as ac-

[2] We are assuming that cases of intrasentential definite description anaphora, such as in appositives, will be handled correctly by the parser and rMRS pattern matcher.

| Factor | Weight |
|---|---|
| Sentence recency | 100 |
| Subject emphasis | 80 |
| Existential emphasis | 70 |
| Accusative emphasis | 50 |
| Indirect object/oblique | 40 |
| Head noun emphasis | 80 |
| Non-adverbial emphasis | 50 |
| Parallelism | 35 |
| Cataphora | 175 |

Table 4: LL Salience Weights

curate a domain-independent ranking of potential antecedents as possible. As yet, our implementation does not cover definite descriptions, although it would be possible, in principle, to do so by utilizing WordNet or another relatively domain-independent lexical resource to capture the synonymy and hyponymy relations required to evaluate sortal compatibility between antecedents and anaphoric definite descriptions.

The RASP GR scheme utilizes 20 GRs organized into an inheritance network (see Carroll et al. (2003)). An example of GR output is given in Figure 1 above. LL's non-coreference constraints can be captured effectively in terms of this GR scheme. For example, RASP output for *Kim seems to want to see him* includes the GRs:

```
(ncsubj see_VV0 Kim_NP1 _)
(dobj see_VV0 he_PPHO1 _)
```

and LL's argument domain filter can be succinctly encoded as:

```
(arg - X N -)
(arg - X P -)
```

where arg $\in \{ncsubj, dobj, iobj, obj2\}$, X is a variable over predicates, and N and P are nominal and pronominal dependents of X respectively. Thus *Kim* and *him* are predicted to be non-coreferential because they are both arguments (dependents) of the same (head) predicate *see*. All of LL's non-coreference and agreement filters can be implemented in terms of such patterns over RASP GR output (see Preiss and Briscoe (2003)).

LL's salience factors and weights are given in Table 4. The factors are straightforwardly computed from RASP GR output and the overall salience of a potential antecedent is a weighted sum of these factors. The original LL weights were manually set on the basis of heuristic experiments with their data.

|  | BC | BU | CH | C1 | C2 |
|---|---|---|---|---|---|
| 1 | 60 | 63 | 63 | 63 | 61 |
| 2 | 51 | 53 | 54 | 55 | 54 |
| 3 | 70 | 70 | 69 | 67 | 69 |
| 4 | 67 | 65 | 70 | 64 | 67 |
| 5 | 55 | 53 | 50 | 52 | 52 |
| $\mu$ | 61 | 61 | 62 | 61 | 61 |

Table 5: Anaphora Results

Preiss and Briscoe (2003) and Preiss (2002) report experiments on a new annotated corpus, drawn from the BNC (Leech, 1992), containing 2400 manually-resolved pronominal anaphors. The corpus was divided into 5 sections containing roughly equal numbers of anaphors. GR output was obtained for the anaphora corpus from five statistical parsers: RASP (BC), Buchholz (2002) (BU), Charniak (2000) (CH), Collins (1997) model 1 (C1) and model 2 (C2). For the latter three parsers, we implemented GR extraction rules for Penn Treebank analyses. In Table 5, we present the results (precision only as all pronouns are always attempted, mean $\mu$) obtained from our implementation of the LL algorithm using the output from all five parsers. These results revealed no significant parser differences in performance on anaphora resolution. However, given that the RASP system is the least lexicalized of the five evaluated, this result suggests that state-of-the-art and domain-independent anaphora resolution can be achieved this way. Error analysis does, however, reveal that the LL algorithm is not optimal. For example, the argument-contained filter, used to prevent *He* and *the man* coreferring in examples like *He believes that the man is amusing*, removed the correct antecedent in 12 sentences with intrasentential anaphors, despite the correctness of the GR output, and similarly 8 antecedents were ruled out by hard agreement constraints.

## 6 Performance on the QA Contexts

In order to evaluate the utility of incorporating the RASP-GR+LL algorithm described above into our baseline QA system, we parsed the contexts identified in Table 3 which contained pronominal anaphors and ran the LL algorithm on the GR output. Manual evaluation of the output revealed that 73.2% of these anaphors were correctly resolved by the system. Error analysis revealed that 36% of the errors were caused by misidentification of the head of the antecedent rather than misidentification of the antecedent (e.g. *El* instead of *El Nino*). Several errors were a consequence of chain effects where an initial anaphor was incorrectly resolved to a full NP antecedent and subsequent anaphors, though correctly resolved to the initial anaphor, then 'inherited' the incorrect full antecedent through this chain[3].

## 7 The Contribution to QA

To assess the contribution to performance that a domain-independent anaphora resolution component would make to a QA system, we augmented the rMRS provided to our baseline QA system by adding variable equality statements binding antecedent and anaphor, as found by the resolution component, in the rMRS representations, which were then fed to the compatibility matching component of the QA system.

In the cases where the correct antecedent is found the potential effects are two-fold. Firstly, the target sentence may receive a higher ranking because of the higher degree of resultant compatibility between question rMRS and that of the target sentence. Secondly, elementary predications obtained via the antecedent may enable the QA system to directly construct an appropriate short answer from the target sentence. Conversely, ranking may decline if antecedents are incorrect, or if variable linking results in irrelevant rMRSs being added.

We assessed the effect on ranking by providing several enriched rMRS representations to the compatibility matching component (along with all the rMRSs for the rest of the top-ranked documents). We assessed the ability of the system to extract a short answer from the highest-ranked sentences by submitting these sentences together with the textual forms associated with the additional rMRSs to the evaluation system.

| | |
|---|---|
| Baseline | 0.491 |
| +antecedent | 0.510 |
| +direct-subst | 0.499 |
| +partial-rMRS | 0.483 |
| +full-rMRS | 0.459 |
| +context | 0.619 |

Table 6: MRR with Anaphora

In Table 6 row 'rMRS' gives the MRR score for a slightly optimized baseline system (as described in section 3) and rMRS+context restates the experimental upper bound achievable through optimal exploitation of the context window (with the caveat that it is an overestimate due to spurious answer matching, see section 4). Row +antecedent reports the MRR score obtained by manually substituting the antecedent found by the anaphora resolution system for the pronoun(s) in the text of the target submission sentence. This provides an upper bound on performance increase, modelling optimal integration of information from the antecedent constituent. Row +direct-subst reports the MRR obtained by automatically enriching the rMRS for the target sentences with the elementary predications corresponding to the head (nouns) of antecedents of anaphors in the context window. Row +partial-rMRS gives results when the rMRS for the target sentence is automatically enriched not only with the rMRS corresponding to the antecedent head but also any other elementary predications and argument constraints on the variable introduced by the antecedent head elementary predication. Row +full-rMRS gives the MRR for a system which integrates the entire rMRS for a context sentence containing an antecedent, and that of its preceeding sentence if it is also anaphorically linked to it or the target sentence.

The improvement in MRR for manual integration of the antecedent illustrates that pronoun resolution has the potential to improve ranking (and identification) of potential answers that would otherwise not be detected. Returning to our previous example, the baseline QA system found the optimal sentence:

> Tungsten producing and consuming coun-

[3] We have not directly tested the alternative architecture in which anaphora resolution is applied to the entire document collection (as with parsing) prior to any matching. Our assumption is that, given the level of performance of the anaphora resolution component, this would result in additional noise and degraded performance over its focussed application in shorter matching contexts.

> tries have been meeting this week in Geneva...

The approach correctly resolved the pronoun *They* in the correct sentence to *country* in the passage. Integrating these rMRS structures increases the ranking (match score) for the correct sentence: *They* (country) *include China, the biggest producer...* above that of the previous (incorrect) sentence. Further analysis demonstrated that 71% of the submissions are improved by this method. In 90% of the cases, this improvement was not reflected in terms of MRR score as the anaphora resolution occurred for sentences in which the resolution referred to an antecedent from the same (correct) submission sentence. For QA in general, however, this result is still important as the improved context can be used during NE recognition and short answer construction. This example also highlights that integration of further information is potentially beneficial. The ranking of the correct sentence would further improve if the integration cintext and target sentences also included the 'tungsten' and 'products' predications from the context sentence.

The automatic methods we explored for the addition of rMRS structures corresponding to the antecedent and increasingly larger amounts of the context sentence underperformed manual integration of the textual antecedent. For +full-rMRS we expected a lower MRR because often the merging of irrelevant rMRSs from the context causes the correct target sentence to be ranked lower or not selected. The optimal amount of rMRS from an anaphorically linked sentence to add to that of the target sentence lies somewhere between the full rMRS and the elementary predication for the head of the antecedent. The main factor in the poor performance of partial rMRS integration is a consequence of the high degree of underspecification in the rMRSs output by our current QA system.

## 8 Conclusions

We have demonstrated experimentally that anaphora resolution is highly-relevant to open-domain QA (both in theory and in practice). However, the accuracy and manner of integration of domain-independent anaphora resolution is critical to effective deployment in open-domain QA. Using an extant resolver to conservatively enrich the rMRS of a target answer sentence containing a pronoun leads to an improvement in MRR, but there is still room for improvement as the approach fails to effectively enrich the representation in 29% of contexts. We have noted failures in the LL algorithm that could be addressed and in addition, the RASP system and other robust parsers (e.g. Buchholz (2002); Clark et al. (2002)) continue to be developed and improve, yielding more accurate starting points for anaphora resolution. Optimally defining the rMRS substructure from an anapohrically-linked context sentence to integrate with the target sentence rMRS is difficult in our current QA system. Extraction of more informative rMRS from the RASP system output should ameliorate this problem.

It is difficult to assess how potentially useful or genuinely effective domain-independent anaphora resolution would be for QA on different data.[4] Newspaper articles, as used in many extant TREC QA competitions, contain many anaphors. However, in different genres, such as scientific articles where we might expect QA systems to find genuinely useful application (e.g. Zweigenbaum (2003)), current anaphora resolution technology may not be so helpful. In this genre, sentences tend to be longer and a higher proportion of anaphors are definite descriptions, requiring more domain-dependent lexical information for high precision resolution.

Morton (2000) reports a small improvement when adding a coreference component to a QA system on the TREC 8 QA dataset. However, his results don't quantify the effect of coreference resolution effectively as his baseline system heuristically includes terms from surrounding sentences, being based on passage retrieval rather than sentence parsing and matching. Morton utilizes a supervised approach to pronominal anaphora resolution which, unlike ours, requires labelled training data and is arguably less domain-independent. He also resolves proper noun coreference and some def-

[4] It is easy to show such an effect in principle: Katz and Lin (2003) demonstrate that there are combinations of questions and document sets containing candidate answers that require syntactic analysis to recover GRs for accurate QA. With 16 carefully chosen questions which exhibit high semantic confusability between subjects and objects of the relevant predicates, system precision leapt from 29% for keyword matching to 84% for GR extraction.

inite description anaphora. Our future work will explore a more graded approach to non-coreference constraints by adopting a probabilistic framework of the type utilized by Ge et al. (1998), incorporating other domain-independent factors such as distance and mention in the resolution decision as well as extension of the approach to definite descriptions.

## Acknowledgements

We thank Ann Copestake and Simone Teufel for their work setting up the task, evaluation and rMRS output from the parsers, and one anonymous referee for useful feedback which helped improve the final version.

## References

E. J. Briscoe and J. Carroll. 2002. Robust accurate statistical annotation of general text. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 1499–1504.

E. J. Briscoe, J. Carroll, J. Graham, and A. Copestake. 2002. Relational evaluation schemes. In *Proceedings of the beyond PARSEVAL Workshop at the 3rd International Conference on Language Resources and Evaluation*, pages 4–8.

S. Buchholz. 2002. *Memory-Based Grammatical Relation Finding*. Ph.D. thesis, University of Tilburg.

J. Carroll, G. Minnen, and E. J. Briscoe. 2003. Parser evaluation using a grammatical relation annotation scheme. In A. Abeille, editor, *Treebanks: Building and Using Syntactically Annotated Corpora*. Dordrecht:Kluwer.

J. Castano, J. Zhang, and J. Pustejovsky. 2002. Anaphora resolution in biomedial literature. In *Proceedings of the International Symposium on Reference Resolution*.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of NAACL-2000*, pages 132–139.

S. Clark, J. Hockenmaier, and M. Steedman. 2002. Building deep dependency structures with a wide-coverage CCG parser. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 327–334.

M. Collins. 1997. Three generative, lexicalized models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the ACL (jointly with the 8th Conference of the EACL)*, pages 16–23.

A. Copestake and D. Flickinger. 2000. An open-source grammar development environment and broad-coverage English grammar using HPSG. In *Proceedings of the Second conference on Language Resources and Evaluation*.

A. Copestake, D. Flickinger, C. J. Pollard, and I. A. Sag. 1999. Minimal recursion semantics: An introduction.

N. Ge, J. Hale, and E. Charniak. 1998. A statistical approach to anaphora resolution. In *Proceedings of the Sixth Workshop on Very Large Corpora*, pages 161–170.

B. Katz and J. Lin. 2003. Selectively using relations to improve precision in question answering. In *Proceedings of the EACL03 Workshop on NLP for QA*, pages 43–50.

C. Kennedy and B. 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.

S. Lappin and H. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.

S. Lappin and M. McCord. 1990. A syntactic filter on pronominal anaphora for slot grammar. In *Proceedings of 28th ACL*, pages 135–142.

G. Leech. 1992. 100 million words of English: the British National Corpus. *Language Research*, 28(1):1–13.

T. Morton. 2000. Coreference for NLP applications. In *Proceedings of 38th ACL*.

J. Preiss and E. Briscoe. 2003. Shallow or full parsing for anaphora resolution? An experiment with the Lappin and Leass algorithm. In *Proceedings of the Workshop on Anaphora Resolution*, pages 1–6.

J. Preiss. 2002. Choosing a parser for anaphora resolution. In *Proceedings of DAARC*, pages 175–180.

P. Zweigenbaum. 2003. Question answering in biomedicine. In *Proceedings of the EACL03 Workshop on NLP for QA*, pages 1–5.