Natural Language Engineering **9** (3): 281–306. © 2003 Cambridge University Press
DOI: 10.1017/S135132490300319X Printed in the United Kingdom

Evaluation-driven design of a robust coreference resolution system

ANDREI POPESCU-BELIS
ISSCO/TIM/ETI, University of Geneva,
40, bvd. du Pont d'Arve, CH-1211 Geneva 4, Switzerland
(*e-mail*: andrei.popescu-belis@issco.unige.ch)

(*Received 26 January 1999; revised 30 September 2000; revised 22 July 2002; accepted 28 November 2002*)

Abstract

In this paper, we describe a system for coreference resolution and emphasize the role of evaluation for its design. The goal of the system is to group referring expressions (identified beforehand in narrative texts) into sets of coreferring expressions that correspond to discourse entities. Several knowledge sources are distinguished, such as referential compatibility between a referring expression and a discourse entity, activation factors for discourse entities, size of working memory, or meta-rules for the creation of discourse entities. For each of them, the theoretical analysis of its relevance is compared to scores obtained through evaluation. After looping through all knowledge sources, an optimal behavior is chosen, then evaluated on test data. The paper also discusses evaluation measures as well as data annotation, and compares the present approach to others in the field.

1 Introduction

The understanding of a discourse by a human or by a computer requires the correct computation of the *discourse entities* that are mentioned – persons, material and conceptual objects, and so on. Although many language engineering tasks may be performed at reasonable levels without taking into account references to discourse entities, their correct identification is often crucial for tasks such as information extraction or question answering.

Reference, the link between linguistic expressions and real or conceptualized entities, is still a matter of vivid debate for the philosophy of language. Nevertheless, numerous programs for reference resolution have already been designed, often as *coreference* or *anaphora* resolution devices (the difference between these terms will be explained later). As more and more language resources become available to computers, the numeric evaluation of coreference/anaphora resolution on unrestricted texts becomes possible, but perfect scores are still far from reach.

In this paper, we show how the extensive evaluation of a coreference resolution program helps developers to make coherent and efficient choices concerning the algorithms and knowledge sources that are used. Numeric evaluation of components validates or invalidates theoretical analyses on various points. The application of such an *evaluation-driven design* requires two important elements: texts on which the resolution has already been done by human judges, as well as quantitative methods to compare the answers of the program with the desired ones. We will illustrate these points in our paper, which is organized as follows. Section 2 sets the framework for coreference resolution and its evaluation; section 3 describes our coreference resolution algorithm as a structured set of knowledge sources that can be optimized; section 4 defines the optimization techniques, and section 5 describes the training/test data. Sections 6 and 7 outline the results obtained using various knowledge sources, while their relevance is measured in section 8. The global results are given in section 9, and our approach is compared to others in section 10.

2 Reference to entities in texts and their computational resolution

2.1 Referring expressions and discourse entities

One of the functions of language utterances is to assert properties about *entities.* Certain fragments of an utterance serve particularly to evoke entities, that is, they function as *Referring Expressions* (RE). Philosophers of language define *reference* as a link between linguistic expressions and, either real-world entities, or – from a mentalist point of view – conceptual structures mapped to real-world entities (Devitt and Sterelny 1999; Jackendoff 2002).

We use the term *Discourse Entity* (DE) to denote the conceptual structures that referring expressions in a discourse refer to[1]. Discourse entities correspond generally to physical or mental objects, such as persons, things, ideas, works of art, etc., but also to "reified" events, relations or properties. In the Discourse Representation Theory, for instance, DEs are represented as first-order logic variables specified by predicates within a Discourse Representation Structure (Kamp and Reyle 1993).

The relation between referring expressions and discourse entities is best named *specification*, following Sidner (1983) among others, leaving the term *reference* for the relation to external-world entities. Specification occurs in the minds of both the speaker and the hearer of a discourse, but in opposite directions: DE → RE for the speaker vs. RE → DE for the hearer.

2.2 Resolution of reference, coreference and anaphora

2.2.1 Coreference vs. *anaphora*

Two important linguistic concepts related to reference are *coreference* and *anaphora.* Coreference is the relation that holds between two REs that specify the same DE (also called co-specification for that reason). Anaphora is a relation between an antecedent RE and an anaphoric RE, and holds, in its broadest sense, when the

[1] This choice follows Grosz, Joshi and Weinstein (1995), and Cristea *et al.* (2002) among others. Other terms are: 'referent', focusing on real-world entities; 'coreference set/chain', focusing on the collection of REs; or 'discourse peg' (Luperfoy 1992).

anaphor cannot be fully interpreted from the point of view of reference without making use of the antecedent[2].

Coreference and anaphora are two different concepts, but in reality instances of anaphora and coreference most often co-occur. This is not always the case, though. Anaphora can occur without coreference: many examples of bridging – also called associative anaphora – involve an anaphoric RE that refers to an entity which is related to the antecedent RE but is not identical to it. Conversely, coreference may occur without anaphora: consecutive uses of the same proper name represent REs that are coreferent but not anaphoric, since each of them refers independently to the same entity, especially in cross-document coreference. Further analyses are found, for example in Van Deemter and Kibble (2000), Vieira and Poesio (2000) and Mitkov (2002).

Pronouns form a special class of anaphors: by virtue of their empty semantic structure, their interpretation almost always requires the use of antecedent REs. Still, not all pronouns are anaphoric: e.g., deictic pronouns such as 'I' and 'you' are not. Some pronouns bear strong syntactic constraints that require a particular kind of processing, e.g.bound anaphors, such as defined by the Government and Binding theory (Chomsky 1981; Lasnik 1989). Some theories do not separate pronouns from other REs, placing them at one end of a gradual scale of semantic content (Ariel 1990); nevertheless, in this case specific knowledge (for instance pragmatic) is needed for their processing (Reboul 1994).

2.2.2 Aspects of reference resolution

The human receiver of a discourse is supposed to locate referring expressions and activate the correct discourse entity for each of them. A *reference resolution program* must also, as a minimum, create and store DEs, and assemble the REs into DEs, for instance the way the DRT does (Kamp and Reyle 1993).

Quite often, computer programs have more limited ambitions, and attempt only to solve one form or another of coreference and/or anaphora. Following the definitions given above, a *coreference resolution program* constructs all the coreference links between REs, whether the REs are anaphoric or not[3]. The transitive closure of all the coreference links generates the RE sets from which the DEs can be abstracted. The goal of our system is to construct directly the sets of coreferent REs in narrative texts, regardless of their anaphoric relations; however, these relations play of course an important role in the identification of coreference relations.

Anaphora resolution programs are normally concerned with the construction of asymmetric links between anaphors and their antecedents. Depending on the defini-

[2] To be even more general, anaphora can occur between expressions that are not nominal REs, such as verbs (one may think of verbs as referring to actions), or expressions whose referring status is unclear, such as negative quantifications.

[3] Most approaches focus only on the 'identity' coreference, i.e. REs that specify the *same* DE. Other referential relations are 'whole/part', 'class/instance', 'name/function', but they are quite difficult to identify consistently in a discourse (Hirschman 1997; Van Deemter and Kibble 2000).

tion of anaphora that is chosen, such programs may or may not attempt to link REs that are coreferent without being strictly anaphoric. For instance, some programs are also interested in indirect anaphors, such as bridging REs (Vieira and Poesio 2000), while others focus on pronominal anaphora (Mitkov 1998). More systems will be discussed in section 10.

2.3 Evaluating reference resolution

The definition of a language engineering task should always be accompanied by the definition of evaluation measures that assess the performance level. Evaluation-driven design, defined in section 4 below, requires an automatic evaluation tool along with desired output in sufficient quantities. Since manual evaluation is too slow to be used frequently, a deterministic evaluation algorithm should be implemented.

Several evaluation methods have been proposed for coreference resolution, and we analyzed them elsewhere (Popescu-Belis 2000). Regarding pronominal anaphora resolution, the asymmetric links between antecedents and anaphors are scored in quite a different way (Mitkov 2001).

Let us consider that the correct REs are given to a program for coreference resolution. There are at least three ways to compare the program's *response* with the correct solution or *key* as defined by human judges: compare the coreference links; compare the two partitions of the RE set into DEs; compare the average correlation between the sender DE and the receiver DE activated by each RE. Each of these views leads to relevant measures (see appendix A for more details). Most of them use two numeric scores inspired from information retrieval, namely recall and precision (Salton and McGill 1983). Intuitively speaking, *recall* is the proportion of correct links in the response with respect to all correct links in the key, while *precision* is the proportion of correct links in the response with respect to all the links in the response. The five measures that we will use are:

MUC measure, $\mathcal{M}$ – the MUC-6 (1995) and MUC-7 (1998) campaigns (the Message Understanding Conferences) used the first scoring algorithm that counted links based only on the *sets* of coreferent REs (Vilain *et al.* 1995);

B^3 measure, $\mathcal{B}$ – to overcome the indulgence of $\mathcal{M}$, Bagga and Baldwin (1998) defined recall and precision *per* RE, then averaged these values;

Kappa-measure, κ – Passonneau (1997) used the *kappa* factor (Krippendorff 1980) to measure agreement above chance between two annotators, and this technique can also be extended to compare an annotator and a program;

Core-DE measure, $\mathcal{C}$ – the concept of 'core-DE' corresponds to the program's view of each correct DE (Popescu-Belis 1998); once DEs are determined, they can be used to compute recall and precision ;

Mutual information measure, $\mathcal{H}$ – understanding reference means that every time a given DE_i is activated in the speaker's mind, the same corresponding DE_j is activated in the hearer's mind (Popescu-Belis 1999); mutual information between the correct and the hypothesized partition of REs yields informational recall and precision scores.

Recall and precision for $\mathcal{M}, \mathcal{B}, \mathcal{C}, \mathcal{H}$ vary from 0 (worst) to 1 (best), while κ varies from -1 (perfect disagreement, i.e. negative statistical correlation) to $+1$ (perfect agreement), 0 denoting random agreement. The harmonic mean of recall and precision is called *f-measure*.

Each measure has its own advantages and drawbacks, one of the most frequent problems being indulgence or leniency, that is, rather high scores for rather poor answers. We use here all five measures, mainly to compare the quality of two responses and find the best one, rather than to estimate the intrinsic quality of a single response. Therefore, we rely on the *concordant variation of all five measures* to draw conclusions on the improvement or degradation of our system's performances.

3 A parametrable algorithm for reference resolution

3.1 Data structures: DEs and REs

The goal of the algorithm proposed here is to build sets of coreferent REs from narrative texts, regardless of the anaphoric relations that they may entertain. The algorithm uses prior markup of REs which excludes non referring noun phrases. The algorithm processes definite, indefinite, and demonstrative noun phrases, as well as third person pronouns. No attempt is made to derive a set for an event, relation, or property that is not reified in the discourse itself by the use of a nominal or pronominal RE. The information structure of a discourse entity DE is (Popescu-Belis, Robba and Sabah 1998):

DE.Activation – a number which models the contextual salience or activation of the RE to the memory of the hearer – in the sense of Alshawi (1987) or Lappin and Leass (1994);

DE.LRE – the list of REs that specify the DE;

DE.LCRE – the list of *characteristic* REs, a subset of the former list which stores the REs that are considered characteristic of the DE and that will be used for comparisons later, in the order of their appearance in discourse.

We adopted a knowledge-poor implementation of DEs to handle unrestricted texts in French, since high-coverage semantic tools were not available to us for this language. DEs do not possess, in our system, abstract semantic representations such as for instance in LaSIE (Gaizauskas *et al.* 1995; Humphreys *et al.* 1998), or characteristic names such as DE.NAME in the AR-Engine (Cristea *et al.* 2002), but they are simply equated to the collection of REs that are used to specify them, and the list of characteristic REs. The semantic content of a DE can be approximated from this collection, that is, from the way it is built (cf. section 6.2) and utilized (cf. section 6.1).

The information stored for each RE includes its position, tokens, and corresponding parse tree in Lexical-Functional Grammar form – we use a parser by Vapillon *et al.* (1997). As no semantic representation is available for an RE, this is inferred from the parse tree and its labels, above all from the head noun. To compare these labels, we built a small network involving the main concepts of our first narrative text (*VA*,

```
⋆ For each $RE_i$ of the text:
    • For each $DE_x$ of the working memory
        — For each characteristic $RE_j$ of $DE_x$ ($RE_j \in DE_x.LCRE$)
            – Find compatibility between $RE_i$ and $RE_j$ — $\boxed{Co}$
        — Infer compatibility between $RE_i$ and $DE_x$ — $\boxed{H}$
    • Decide if a new DE must be created from $RE_i$ — $\boxed{Cr}$
    • If not, decide to which $DE_y$ the $RE_i$ is attached — $\boxed{At}$
    • Update the list of characteristic REs of $DE_y$ ($DE_y.LCRE$) — $\boxed{L, P}$
    • Update the activation of all the DEs — $\boxed{A}$
    • Update the list of DEs in the working memory — $\boxed{M}$
```

Fig. 1. The resolution algorithm with the names of the knowledge sources (boxed)

described below), hence about 50 concepts mainly related to persons. Other programs have used WordNet (for English) to derive such a semantic network (Cardie and Wagstaff 1999; Vieira and Poesio 2000; Soon, Ng and Lim 2001).

3.2 The resolution algorithm

The resolution algorithm draws inspiration from two cognitive constraints on reference resolution. First, the algorithm processes the REs sequentially, from the beginning of the text, and not globally at the end, or even in reverse order, as in Cardie and Wagstaff (1999). Secondly, the algorithm stores the most salient/active REs in a limited-size working memory.

For each RE, the algorithm attempts to determine whether the RE evokes an existing DE or a new DE (cf. figure 1). The decision depends upon the compatibility between the current RE and the existing DEs, and on their activation. As a result, either the current RE is attached to a DE from the working memory, or a new DE is created using the current RE. In both cases, activation is updated.

The algorithm makes use, at each stage, of one or more *knowledge sources*, which are explained hereafter together with their possible *behaviors*. The *minimal behavior* or *baseline* is the minimal contribution that a knowledge source must bring to the algorithm in order to obtain output that can be evaluated. The following knowledge sources are defined:

H, referring compatibility between an RE and a DE: a heuristic to find whether an RE may specify or not a given DE, using the compatibility of the RE with the REs that characterize the DE (from DE.LCRE);

Co, referring compatibility between two REs: a set of criteria stating whether two REs may corefer or not;

Cr, creation/attachment rules: rules that decide between the creation of a new DE and the attachment of the current RE to an existing DE, depending on the characteristics of the RE. The baseline is: a DE is created if there are no DEs that are active enough and compatible with the current RE;

At, criterion for attachment: simply attach the current RE to the most recent among the most active and compatible DEs;

L, evolution of the list of characteristic REs (DE.LCRE): several rules for adding an RE to the list or removing it (baseline: store all REs into DE.LCRE);

P, include or not pronouns in the list of characteristic REs of a DE (baseline: include them);

A, activation: rules and factors that determine the activation of REs;

M, working memory, storing DEs for further reactivation: its size is variable (baseline: store all DEs into the working memory).

It is quite visible that A, M, L and P are optional, i.e. they can be set to baseline without ruining *a priori* the performance of the algorithm. The core of the algorithm is the combination of H and Co, as well as Cr, though the baseline behavior of Cr is simple. The order of the knowledge sources according to their importance for reference resolution is *a priori* assessed as: $Co > H > Cr > L > P > M > A$ (At is not considered, since it cannot be modified or turned off).

4 Principles of evaluation-driven design

Evaluation-driven design is a general optimization method that allows us to determine the best option or behavior for each knowledge source (henceforth, KS); to estimate the contribution of each KS to the overall results of the algorithm; and above all to maximize the overall performance of the algorithm by finding a combination of behaviors that is optimal on a given type of data. The method has the following steps:

1. For each KS, identify its minimal or baseline behavior(s), that is, the contribution the KS is required to bring to the whole algorithm so that a valid response can be generated. KSs whose minimal behaviors are null are called *optional KSs*. The baseline score is obtained when all KSs are set to their baseline behaviors.
2. Order the KSs according to their estimated importance, optional KSs coming last. Distinguish between bounded KSs, i.e. those that have a finite number of behaviors (e.g.the combinations of a set of rules), and unbounded KSs, i.e. those that have a large or infinite range of behaviors (e.g.the values of one or more numeric parameters).
3. Run the algorithm with the *a priori* best behavior for each KS. Then apply the following optimization operations to each KS, starting with the most important ones.
   (a) For a bounded KS, evaluate all its behaviors and adopt the one leading to the best scores (if the number of behaviors is large, this exploration is best done automatically).
   (b) For an unbounded KS, use a standard optimization method to determine the behaviors (often the parameter values) that generate the best scores. If several behaviors are optimal, pick one according to *a priori* relevance.

Once an optimal combination of behaviors is fixed, it becomes possible to measure for each KS the relevance of the chosen behavior with respect to other behaviors. In particular, if the behavior is a combination of rules, it is possible to quantify the contribution of each rule to the final behavior.

Evaluation-driven design bears some resemblance with machine learning methods based on annotated corpora, which have been used for coreference resolution – such as probabilistic Bayesian models (Ge, Hale and Charniak 1998) or decision trees for pronominal anaphora (Aone and Bennett 1996), or decision trees for coreference resolution (McCarthy and Lehnert 1995; Vieira and Poesio 2000; Soon, Ng and Lim 2001). Some of these studies make in fact intense use of *a priori* knowledge about the features that are learned and the way the classifiers are used.

In fact, our method exhibits indeed a kind of learning, which is subsequently validated on test data. However, our method strongly relies on analytical, knowledge-based considerations in the definition of the behaviors, in the choice of the initial behaviors, and in the choice among behaviors when several are optimal. Automatic optimization or "learning" is just one of the techniques used in our study.

5 Data for training and test

Evaluation-driven design requires a certain amount of data to select an optimal behavior for the various knowledge sources. The data consists of texts in which (co)references have already been solved and annotated by human judges. The optimization procedure maximizes the scores on this data, while the scoring procedure uses *other* pieces of annotated data to check how well the system scores on data other than the training data.

We chose here narrative texts, as in the MUC approach, rather than series of sentences covering various reference phenomena – there are no such test suites, to our knowledge, despite attempts in the DiET project to cover some uses of pronouns (Lewin *et al.* 1999). But even texts with annotated coreferences are still scarce. The MUC corpora contain dozens of short articles in English. As we work on French, the only pre-existing resource was the first chapter of a novel by Balzac (*Le Père Goriot*), in which references corresponding to human story characters were marked for 3078 REs (all REs were marked for the first 184 REs). We annotated, together with I. Robba at LIMSI-CNRS, another French text, namely the beginning of a short story by Stendhal (*Vittoria Accoramboni*), with all the REs and references. These two resources are described in table 1[4].

To annotate the correct discourse entitities (sets of coreferring REs) on the *VA*

[4] Information for obtaining the MUC corpora can be found at `http://www.itl.nist.gov/iad/894.02/related_projects/muc/`. The chapter from *Le Père Goriot* is available on the Silfide webserver at `http://www.loria.fr/projets/Silfide/`. Other coreference resources for French are under development within the ANANAS project, cf. `http://www.inalf.fr/ananas/`. For a resource in French with annotated pronominal anaphors, see Tutin, Trouilleux, Clouzot, Gaussier, Zaenen, Rayot and Antoniadis (2000).

Table 1. *Characteristics of the annotated texts*

Text	Words	REs	Noun phrases	Pronouns	Key DEs	RE to DE ratio
VA	2630	612	510 (83%)	102 (17%)	372	1·64
PG	28576	3262	1864 (57%)	1398 (43%)	480	6·80

Table 2. *Names and characteristics of the twelve fragments*

Number of REs	All REs are annotated (T)	Only persons are annotated (P)
∼200	*VA2T1, VA2T2, VA2T3, PG2T*	*PG2P1, PG2P2, PG2P3*
∼600	*VA6T*	*PG6P1, PG6P2, PG6P3*
∼3100	—	*PG31P*

text, we designed a user-friendly annotation interface and defined three annotation levels (Popescu-Belis 1998), which parallel those used for the MUC-7 data (Hirschman 1997):

Level I – annotation of minimal discourse structures that are useful for reference resolution (sentences, paragraphs, sections);

Level II – annotation of boundaries of REs, which is done here semi-automatically, by validating candidate REs proposed by an LFG parser (Vapillon *et al.* 1997);

Level III – annotation of groups of REs corresponding to DEs – for each RE, the annotator selects between 'DE creation' or 'attachment to an existing DE'.

For evaluation-driven design, level III annotated data is used. For testing, the system processes data at level II, since the identification of REs is not considered part of the coreference task. Then, the system's response is scored automatically against level III human annotation.

The interface exports annotations using SGML/XML tags in the source text, following either the MUC-7 syntax or the syntax proposed by Bruneseaux and Romary (1997) – more recent guidelines by Salmon-Alt (2001) being under study. Automatic indexing and tag generation guarantee a correct tag structure, and can also be applied to check imported resources. The interface is designed for efficient coreference annotation: for instance, a text with about 600 REs and 400 DEs is annotated in about one hour. The interfaces, the scoring module, and the reference solver constitute a Reference Resolution Workbench, on the model of other tools such as Alembic (Day *et al.* 1997), GATE (Cunningham *et al.* 1997), or the proposals of Mitkov *et al.* (2000).

To be able to use several training/test sets, the *VA* and *PG* texts were fragmented into several blocks, respecting narrative coherence boundaries (cf. Table 2). Only one block of ca. 200 REs with all referents annotated was available from *PG*, and three other blocks were created from *VA*. Three blocks of ca. 200 REs, in which only human referents were annotated, were extracted from *PG*, as well as three blocks of ca. 600 REs and one of ca. 3100 REs (total). The whole *VA* text represents a block of ca. 600 REs with all references annotated. The use of twelve fragments ensures that the results of evaluation-driven design are not particular to a certain fragment (though they certainly depend on the type of texts), by optimizing on one data set and testing on another one.

6 Optimization of bounded KSs

6.1 Referring compatibility between a RE and a DE: H

The H knowledge source selects among the existing DEs those that the current RE may specify, and rejects those that it certainly cannot specify. The referring compatibility between the RE and a DE is computed by averaging the compatibility of the current RE with the REs of the DE, stored in the list of characteristic REs (DE.LCRE). The referring compatibility between two REs is determined using another, more elementary KS, namely *Co*, discussed in section 8.1. H has four possible behaviors (ways of averaging), named $H_1 \dots H_4$:

– RE_i may specifyDE_a *iff*. . .
H_1 : RE_i can be coreferent with **all the REs** of $DE_a.LCRE$.
H_2 : RE_i can be coreferent with **at least one RE** of $DE_a.LCRE$.
H_3 : RE_i can be coreferent with **the first RE** of $DE_a.LCRE$.
H_4 : RE_i can be coreferent with **at least** $S\%$ **of the REs** of $DE_a.LCRE$.

Aside from $H_1 \dots H_4$, there are two minimal behaviors, noted H_{all} and respectively H_{none}: either all DEs are deemed to be compatible with the current RE, or none is compatible. Behavior H_{all} generates a response in which all REs are grouped into one DE, if no creation rules *Cr* override this (cf. section 8.2). Behavior H_{none} generates a response in which no REs are grouped, if no attachment rules *Cr* override this. The score variations from H_{none} to H_{all} on our longest text (*PG31P*) are shown in Table 3 (note that *Cr* attachment rules used here lead to non-zero scores). As expected, H_{all} favors recall and H_{none} precision. But $\mathcal{M}$ precision also increases from H_{none} ro H_{all}, which reflects the indulgence of the $\mathcal{M}$ measure when the key has a high coreference rate (RE to DE ratio) and the response conflates too many REs. Indeed, $\mathcal{M}$ f-measure for H_{all} reaches 0.9 on *PG31P*.

Turning now to $H_1 \dots H_4$, H_1 is a test that requires compatibility of the current RE with all REs from a DE to conclude that the RE may evoke the DE. This seems quite restrictive as it does not allow much variation in the naming of a referent. Moreover, sometimes REs introduce new information about an entity, so compatibility between *all* REs should not be required. Conversely, H_2 requires compatibility with at least one RE, being thus too indulgent. Though extremely basic, H_3 embodies the idea that the first RE introducing a DE is characteristic. Finally, H_4 requires that the current RE is compatible with $S\%$ of the REs in a DE. Depending on S, H_4 (better noted $H_4(S)$) exhibits a range of behaviors, from H_1, for $S = 100\%$, to H_2, for S just above zero. The values of S slightly under 100% generate a more flexible version of H_1, allowing for some exceptions in compatibility.

Table 3. *Score variation from H_{none} to H_{all} on PG31P text*

	$\mathcal{M}$	$\mathcal{B}$	κ	$\mathcal{C}$	$\mathcal{H}$
Recall	+200%	+500%		+1700%	+17%
Precision	+45%	−79%		−68%	−84%
f-measure	+120%	+17%	−129%	+256%	−62%

Since H depends on the list of characteristic REs of each DE, we optimized at the same time the three KSs: H, L and P. Rather than finding an optimal behavior for H, with L and P fixed, then optimizing L and P, then the others, then looping again through H, L, P, we optimized directly $H \times L \times P$ by automatically exploring the possible combinations. Results are given in section 6.3.

6.2 The list of characteristic REs: L *and* P

The list of characteristic REs (DE.LCRE field) is used according to the H knowledge source, but it is constructed using L and P. An elaborate construction criterion would store only "definition REs" in this list, thus approximating a semantic representation of the DE. In the absence of a criterion to find these REs, we consider two KSs: L, the size of the DE.LCRE list and P, inclusion or not of pronouns in it.

The various behaviors of L are simply the various sizes, the baseline being unlimited size. REs are added to a given DE.LCRE by discarding identical REs, until maximum size is reached. Then, when a new RE is appended to the list, the shortest among the stored REs is deleted. Behaviors of L have the following consequences. A smaller list size makes the algorithm faster. A size of 1 RE has effects similar to H_3, except that the longest RE is used, instead of the first one. In fact, H and L are closely coupled, which is another reason to jointly optimize $H \times L \times P$.

P has two behaviors: either pronouns are stored in DE.LCRE (baseline behavior) or they are not. The *a priori* analysis raises contradictory arguments. On the one hand, pronouns should be treated like all the other REs, all the more that they provide gender and number information about the DE (and possibly also predicate and argument information when better parsers are available). Pronominal reference contributes to activation, and positional information from pronouns is relevant too. On the other hand, pronominal REs have no semantic content, being thus semantically compatible with all other REs; therefore, they are useless or even detrimental to semantic compatibility tests. Moreover, their gender and number information is already contained somewhere else in DE.LCRE, since they were already attached to this DE. We must therefore rely on evaluation to select the best behavior.

6.3 Evaluation of $H \times L \times P$

The combined knowledge source $H \times L \times P$ was optimized separately on each of our data fragments. For H, the four $H_1 \ldots H_4$ behaviors were tested, and so were $+p$ and $-p$ for P. The L factor (size of DE.LCRE) and the parameter S in $H_4(S)$ were sampled, so that $H \times L \times P$ remained a bounded KS. The behaviors leading

Table 4. *Best $H \times L \times P$ behaviors for three fragments ($\pm p$ stands for $+p$ and $-p$, while $\geq x$ stands for $[x, +\infty[$)*

	VA2T2			*VA6T*			*PG6P1*		
	H	*L*	*P*	*H*	*L*	*P*	*H*	*L*	*P*
$\mathcal{M}$	H_2	≥ 1	$-p$	H_2	≥ 15	$-p$	H_2	≥ 50	$+p$
							$H_4(>0)$	≥ 8	$\pm p$
$\mathcal{B}$	$H_4(0.8)$	≥ 6	$\pm p$	$H_4(0.4)$	≥ 15	$-p$	$H_4(0.8)$	≥ 60	$+p$
	$H_4(0.8)$	≥ 4	$-p$						
κ	H_2	≥ 1	$-p$	H_2	≥ 10	$-p$	$H_4(0.8)$	≥ 60	$+p$
	$H_4(0.2)$	≥ 4	$-p$						
$\mathcal{C}$	H_3	≥ 20	$+p$	$H_4(0.2)$	≥ 15	$-p$	$H_4(0.8)$	≥ 60	$+p$
	H_3	≥ 4	$-p$	$H_4(0.4)$	≥ 8	$-p$			
	$H_4(0.6)$	≥ 4	$-p$						
	$H_4(0.8)$	≥ 6	$\pm p$						
$\mathcal{H}$	same	same	same	H_1	≥ 4	$\pm p$	$H_4(0.8)$	≥ 60	$+p$
	as $\mathcal{C}$	as $\mathcal{C}$	as $\mathcal{C}$	$H_4(0.6)$	≥ 4	$-p$			
				$H_4(0.8)$	≥ 4	$\pm p$			

to the best scores on each fragment and with each evaluation measure are shown in Table 4, for three representative fragments only. The corresponding values of the scores for those behaviors are shown in table 5.

As Table 4 shows, behaviors leading to best scores sometimes vary even between different measures on the same fragment. The intersection of the sets of optimal behaviors for all five measures is empty on almost every fragment. The overlapping between certain optimal behaviors is sufficient to increase the confidence that they would generate good responses on any text.

Behavior H_1 is infrequent among optimal behaviors, but $H_4(0.8)$ appears quite frequently (recall that $H_4(0.8)$ is a more flexible behavior, akin to H_1, requiring compatibility of the current RE with at least 80% of the REs). At the opposite range, $H_4(S)$ with a small value of S often appears at the same time as H_2, though far less frequently than $H_4(0.8)$. Even H_3 is not completely absent from local optima. Regarding L, the interval of optimal values is generally $[x, +\infty[$, which means that keeping all the REs ($L = +\infty$) is the most widespread optimal behavior. Increasing the algorithm's speed would be the only reason to reduce L. As for P, to include the pronouns in DE.LCRE is most of the time an optimal behavior, though cases in which this is indifferent are not infrequent.

After a survey of all the results, the most frequent optimal behavior appears to be $H_4(0.8)$, accompanied by the storage of all REs, including pronouns, in DE.LCRE. We select these settings henceforth as a *joint optimal behavior* for $H \times L \times P$. The scores with these settings are sometimes lower than the best scores that can be obtained on a given fragment (they certainly cannot be higher), but these differences

Table 5. *Best score values for three fragments*

	VA2T2	*VA6T*	*PG6P1*
$\mathcal{M}$	0.78	0.76	0.86
$\mathcal{B}$	0.79	0.78	0.52
κ	0.51	0.58	0.18
$\mathcal{C}$	0.71	0.59	0.54
$\mathcal{H}$	0.89	0.91	0.60

Table 6. *Decrease of f-measure between the local optimal scores and the scores with* $H_4(0.8)$ *and no restrictions on DE.LCRE*

	VA2T1	*VA2T2*	*VA6T*	*PG2T*	*PG2P1*	*PG2P2*	*PG6P1*	*PG6P2*	*PG31P*
$\mathcal{M}$	1.5%	1.3%	2.6%	3.7%	6.4%	4.7%	5.8%	5.8%	8.9%
$\mathcal{B}$	1.2%	0	1.3%	10.0%	0	0	0	0	8.5%
κ	1.9%	2.0%	3.4%	16.7%	41.2%	20.8%	27.8%	0	61.5%
$\mathcal{C}$	0	0	3.4%	2.5%	2.4%	1.7%	0	0	10.6%
$\mathcal{H}$	0	0	0	7.5%	12.5%	1.6%	6.2%	4.3%	7.9%

are small, as shown in Table 6. This confirms the optimality of the selected behavior, especially when compared to other differences shown below (Tables 8 and 9).

The choice of $H_4(0.8) \times +\infty \times +p$ is also supported by knowledge-based analysis. This combination makes the best use of the information embedded in the REs: all REs are kept in DE.LCRE, and almost all are used for RE ↔ DE compatibility; but also, up to 20% "non-standard" REs are tolerable in DE.LCRE. A choice based only on optimization would require us to compute the data in Table 6 for each possible behavior, then choose the behavior that globally minimizes the differences with the local optimal scores. Not only is this cumbersome, but the result would only be tuned to the present training set.

7 Optimization of unbounded KSs

7.1 Results on working memory: M

The most active DEs are stored in a working memory of adjustable size (not to be confused with the size of LCRE for each DE). The various sizes, from one DE to unlimited, constitute the behaviors of M. Not only does this cognitive feature affect the speed and physical memory requirements of the program, but it also affects directly the scores. A small memory quite predictably causes recall errors, but prevents precision errors. More precisely, the capacity to "forget DEs" (small memory) avoids merging certain non-anaphoric definite descriptions that are referentially compatible according to *Co*. Up to this point, a memory size of 40 DE has been used.

Memory size values from 1 to 150 DE have been evaluated, with a focus on the intervals that show greater variation of the scores. On almost all fragments, the five scores $\mathcal{M}, \mathcal{B}, \kappa, \mathcal{C}$, and $\mathcal{H}$ increase or decrease *together* between two responses, and

Fig. 2. Influence of memory size on *VA2T1* scores. The graph shows $\mathcal{M}$ recall and precision for memory sizes of 1, 3, 4, 8, 9, 10, 14, 15–18, 21–29, 30–47, 48–57, 68–69 and 70–80 (from left to right). The curves corresponding to 0.34 and 0.71 f-measure are dashed.

the best scores are reached for a memory size between 15 and 30 DEs. Sometimes a second maximum is reached for sizes greater than 70 DEs. On all fragments, the scores uniformly decrease when the memory size exceeds the first optimum, but the decrease is always under 2%. We thus decide that a *memory size of 30 DEs* is the most convenient in terms of: score, knowledge-based analysis, and program speed. This value leads to scores attaining or exceeding some of the previous scores with $H_4(0.8) \times +\infty \times +p$.

Typical values are shown in figure 2, for memory sizes ranging from 1 to 80 DEs, 1 DE corresponding to the lowest f-measure (left part of the curve). Several sizes may generate the same response, hence the same score; that is why there are less data points than memory size values. For small memory sizes (up to 7 DEs), both recall and precision increase with the size, but afterwards recall increases and precision decreases. A closer view indicates more clearly that as recall increases and precision decreases, the f-measure has two local maximums, the first one reaching 0.71 for 11–15 DEs in this example.

7.2 Activation of the DEs: A

The values of the activation/salience factors can also be tuned by evaluation, and their relevance confirmed. The activation mechanism works as follows. Upon creation, a DE receives an initial activation (a number), which increases on subsequent evocations by an RE. Activation decreases with the number of words, sentences and paragraphs elapsed as the algorithm proceeds through the text. The list of activa-

Table 7. *Chosen values for the activation factors*

Factor	Value
Additional activation for DE creation	15
Activation for a noun phrase RE	20
Activation for a pronominal RE	13
Activation decrease per word	1
Activation decrease per sentence	3
Activation decrease per paragraph	5
Additional activation for a proper noun RE	15

tion factors is shown in Table 7. The possible values (or behaviors) are integers between -5 and 30 for each of the factors. Negative values seem rather counterintuitive, but have been tested nevertheless.

The optimization procedure is based on gradient ascent on the most sensitive measure, κ (complete exploration of all values is intractable). At each optimization cycle, an activation factor is randomly chosen, and its value is increased, decreased, or left unchanged, attempting to increase the current score, with random choice of the change in case of a tie. This method guarantees a score increase, but may get blocked in a local optimum. The method being time-consuming, we used only the shortest fragments, *VA2T1, VA2T2, PG2T.*

To test A's relevance, we started from a minimal behavior with no activation factors (null values) and ran the gradient ascent procedure. After about 700 cycles, all the $\mathcal{M}, \mathcal{B}, \kappa, \mathcal{C}$, and $\mathcal{H}$ scores increased significantly on each fragment, respectively: for *VA2T1*, (+49%, +13%, +175%, +97%, +6%); for *VA2T2*, (+55%, +23%, +1325%, +90%, +14%); for *PG2T*, (+10%, +2%, +42%, +15%, +1%).

Starting now with values that are initially set through knowledge-based analysis, numerous intervals of values lead to a local maximum. Finding the whole set of optimal values on each text, then intersecting all these sets, would be excessively long. That is why knowledge-based analysis is once again necessary to choose a near-optimal parameter set, valid for all texts and measures. Such a parameter set is shown in Table 7. Some constraints observed in the optimization process are: activation decrease per word should be a little above zero; activation decrease per sentence and per paragraph should be a little higher; specification of a DE by a noun phrase RE should activate the DE more than specification by a pronominal RE, and even more when the RE is a proper name.

8 Assessing the relevance of *Co* and *Cr* behaviors

Regarding the two remaining KSs – *Co*, compatibility between REs, and *Cr*, creation of DEs – we will show that the behaviors set by *a priori* analysis are indeed the best behaviors, and we will quantify the contribution of each of their component rules to the final scores. Starting from the score of the *a priori* best combination of components, we compute the score variation when each of them is removed, and the score variation when each of them is used alone, then rank the behaviors.

8.1 Referring compatibility between REs: Co

The compatibility KS is central to our algorithm, as it is used by H. Three compatibility rules between REs are defined:

R_G – two REs may evoke the same DE if they have the same *gender*;
R_N – two REs may evoke the same DE if they have the same *number*;
R_S – two REs may evoke the same DE if they have compatible *semantic contents*, that is, if the head of the RE coming last is a duplicate, synonym, or hypernym of the head of the RE coming first (but not a hyponym, as this is uncommon)[5]. The same rule applies to the modifiers of the heads, if any.

The *a priori* behavior of this KS requires *all* three rules to be satisfied for two REs to be compatible (unparsed REs are never compatible). To evaluate the relevance of the rules, we discard them alternatively, then conserve alternatively only one of them, and study the average score variation (recall/precision is averaged on $\mathcal{M}, \mathcal{B}, \mathcal{C}, \mathcal{H}$, then f-measure is averaged on all five measures). The scores in Table 8 (upper half) show that removal of any of the rules drives f-measure down. Though recall expectedly increases, as fewer coreferences are ruled out, precision decreases even more. This proves that all three rules are relevant. The scores tend to decrease even more when only one rule is left, as shown in the lower half of the table.

It appears that R_S has the strongest influence on the results: by far the lowest scores occur when R_S is removed, and by far the highest scores appear when it is the only rule. The semantic compatibility rule R_S is thus the most important of the three. It is not clear which is the second most important rule, since it is R_G for *VA6T* ($-4\% < -2\%$ and $-52\% > -55\%$), and R_N for *PG6P1* ($-3\% > -15\%$ and $-53\% < -47\%$), but there is disagreement for *PG6P2*. The importance of R_G and R_N seems similar on these texts, but in general, it depends on the proportion of singular/plural references in the texts, and on the availability of gender information in a given language.

8.2 Creation / attachment rules: Cr

The decision to create a new DE from the current RE or to attach the current RE to an existing DE depends first on *Cr* and only afterwards on RE ↔ DE compatibility. Three rules compose *Cr*:

R_{def}**:** force the attachment of each definite RE to an existing DE;
C_{indef}**:** force the creation of a DE for each indefinite RE;
C_{NP}**:** force the creation of a DE for each non-pronominal RE.

In the initial behavior of *Cr*, the three rules are discarded, as it is unclear from *a priori* analysis whether they might increase the scores or not. Rule C_{NP} could be activated only as a baseline, or to study pronoun resolution, but R_{def} and C_{indef}

[5] Similar rules are also used by others (Cardie and Wagstaff 1999; Vieira and Poesio 2000; Soon, Ng and Lim 2001), together with semantic networks often derived from WordNet.

Table 8. *Average score variation when each Co rule is alternatively disabled, then when each Co rule is used alone*

		VA6T	*PG6P1*	*PG6P2*
Without R_G	Recall	−4%	+4%	−2%
	Precision	−4%	−9%	−7%
	f-measure	−4%	−3%	−2%
Without R_N	Recall	+1%	−4%	−1%
	Precision	−4%	−7%	−6%
	f-measure	−2%	−15%	−7%
Without R_S	Recall	+26%	+18%	+21%
	Precision	−61%	−30%	−39%
	f-measure	−50%	−58%	−48%
Only R_G	Recall	+27%	+18%	+21%
	Precision	−64%	−35%	−43%
	f-measure	−52%	−53%	−43%
Only R_N	Recall	+32%	+29%	+27%
	Precision	−67%	−42%	−48%
	f-measure	−55%	−47%	−44%
Only R_S	Recall	+1%	+2%	−3%
	Precision	−8%	−14%	−9%
	f-measure	−4%	−10%	−5%

Table 9. *Average score variation when only one of the Cr rules is active*

		VA6T	*PG6P1*	*PG6P2*
With R_{def}	Recall	−10%	−3%	−1%
	Precision	−34%	−11%	−11%
	f-measure	−38%	−70%	−54%
With C_{indef}	Recall	−5%	−10%	−7%
	Precision	+4%	−6%	+4%
	f-measure	−2%	−12%	−0·3%
With C_{NP}	Recall	−32%	−26%	−31%
	Precision	−3%	+6%	+8%
	f-measure	−29%	−21%	−26%

correspond to certain linguistic intuitions, such as: “definite REs generally specify existing DEs” – but see Poesio and Vieira (1998) for corpus evidence against this, or “indefinite REs generally introduce new DEs”. Numeric evaluation of the relevance of each rule is thus needed.

Table 9 shows the variations of recall and precision (averaged on $\mathcal{M}, \mathcal{B}, \mathcal{C}, \mathcal{H}$), and of f-measure (averaged on $\mathcal{M}, \mathcal{B}, \kappa, \mathcal{C}, \mathcal{H}$) when these rules are added. Obviously, none of the three rules brings any significant improvement: all f-measures decrease when a rule is activated. R_{def} favors RE attachment, therefore it drives down precision more than recall, and f-measure even more, because κ decreases severely. C_{indef} favors DE creation, hence a lower recall and a slightly better precision, but not enough to increase f-measure. A similar result holds for C_{NP}, with a dramatic decrease in recall and a slight increase in precision, while f-measure clearly shows that C_{NP} is not beneficial either. Overall, these scores confirm that the links between definiteness and reference are more subtle than the R_{def} or C_{indef} rules, which are useless in their present formulation.

9 Final results

The behaviors of all our KSs have been evaluated at this point. Our measures indicated which are the best behaviors and provided an experimental basis against which knowledge-based considerations were tested. When these considerations did not provide solid conclusions, evaluation was the only means to choose among behaviors.

Using all the final behaviors, we attempted to cycle through the evaluation sequence and optimize again the KSs in the same order. This did not lead to significant changes. The figures were comparable to the uncertainty on the annotation of the correct answer (Passonneau 1997), and they did not single out a particular behavior among the preceding ones. We adopt thus the following joint optimal behaviors:

For H – $H_4(0.8)$, i.e. compatibility with at least 80% of the REs in a DE;
For Co – R_S, R_N, R_G, i.e. gender, number and semantic agreement;
For Cr – no creation/attachment constraint;
For L **and** P – all the REs ($L = +\infty$), including pronouns ($+p$), are kept in the list of characteristic REs of a DE (DE.LCRE);
For A – the activation factors have the values in table 7;
For M – the working memory stores 30 DEs.

The final scores are shown in Table 10 for three texts: the two original ones (*VA6T* and *PG31P*) and a fragment of *PG* having the same size as *VA* (i.e. *PG6P1*, about 600 REs). The scores are of course slightly below the local optimal score for each text and measure, but well above the various baseline scores that can be obtained by choosing baseline behaviors for each KS. Several combinations of behaviors can count as a baseline: for instance, the decrease in performance due to baseline behaviors for *Co* and *Cr* is shown respectively in Tables 8 and 9, while the variations due to H are shown in Table 3. More important, the score obtained by the joint optimal behaviors on each fragment is very close to the local optimal score for each fragment – compare Table 10 with table 5. The various fragments used to choose the joint optimal behaviors ensure that they represent a suitable option for our domain of narrative texts.

The differences in score between the three texts are notable. The κ score decreases severely from *VA6T* to *PG6P1*, and further on to *PG31P*. A similar decrease is

Table 10. *Final scores of the algorithm with the optimal behaviors*

Measure		*VA6T*	*PG6P1*	*PG31P*
$\mathcal{M}$	Recall	0.72	0.72	0.75
	Precision	0.76	0.87	0.89
	f-measure	0.74	0.79	0.82
$\mathcal{B}$	Recall	0.76	0.50	0.38
	Precision	0.79	0.54	0.48
	f-measure	0.77	0.52	0.42
κ		0.56	0.13	0.05
$\mathcal{C}$	Recall	0.56	0.57	0.45
	Precision	0.58	0.51	0.39
	f-measure	0.57	0.54	0.42
$\mathcal{H}$	Recall	0.90	0.60	0.58
	Precision	0.91	0.61	0.58
	f-measure	0.91	0.60	0.58

observed for $\mathcal{H}$. In addition, the $\mathcal{B}$ and $\mathcal{C}$ scores decrease from *VA6T* to *PG6P1* and *PG31P*, however less dramatically. The $\mathcal{M}$ scores vary in the opposite way, but the differences are lower, probably due to the indulgence of this measure on texts with high coreference rates (Popescu-Belis 2000). The low value of κ for *PG31P* does not necessarily signify a very poor response, since the other scores reach reasonable levels. Indeed, if we use H_2 on *PG31P* we obtain the following scores for $\mathcal{M}, \mathcal{B}, \kappa, \mathcal{C}, \mathcal{H}$ respectively: 0.90, 0.30, 0.13, 0.32, 0.25. Here, all the $\mathcal{B}, \mathcal{C}$, and $\mathcal{H}$ values are low, despite a κ score higher than the one with $H_4(0.8)$ shown in table 10; therefore, H_2 should not be preferred to $H_4(0.8)$. Once again, all score values must be used together to estimate the quality of a response.

On the whole, the final response on *VA6T* is better than responses on *PG6T* and *PG31P*. This is quite certainly due to the availability of a small semantic network for the main characters in *VA*, used by the R_S semantic compatibility rule. The response on *PG6P1* is better than that on *PG31P* as the first text is shorter, and confusions probably accumulate on longer texts, thus merging coreference chains. There is no clear indication in general that recall errors far exceed precision errors or vice-versa, a sign that our system is well-balanced and maximizes f-measure. The scores of our system are globally in the same range as those obtained at the MUC-6 and MUC-7 campaigns (50–70%). Our task here is however a little easier, since the correct REs are available to our system. The quantification of these differences is uneasy, so an exact comparison to the MUC scores is probably not relevant here.

10 Other studies of coreference resolution

Karttunen (1976) was one of the first to suggest that a reference resolution program has to manipulate some kind of representations of entities, which he called ‘object files'; his work focused on formal constraints mostly for intra-sentential coreference. Appelt and Kronfeld (1987) introduced the notion of individuating sets of features for each discourse entity, within a computational model of referring acts. The Discourse Representation Theory (Kamp and Reyle 1993) makes extensive use of a logic-form representation of entities, and so does work in natural language generation (Appelt 1985; Dale 1992). All these models provided useful ideas for our work, but since we aimed at implementing a robust system, the knowledge sources that these models presuppose were not available to us.

Grosz and Sidner (Grosz 1981; Sidner 1983) studied the evolution of the focus of attention on referents through the consecutive utterances of a discourse, and proposed an application to dialog, including human-computer dialog (Grosz and Sidner 1986). The Centering Theory advocated a more fine-grained view of the focus stack and explored its relation to coherence betweeen utterances (Grosz, Joshi and Weinstein 1995). The computation of the focus stack or of the centers requires more knowledge sources than those that were available to us – though simplified versions have been implemented by Azzam, Humphreys and Gaizauskas (1998) or Cristea *et al.* (2002) for instance. We drew inspiration for our activation model from Alshawi's (1987) model of memory and his salience factors, adapted for pronominal anaphora resolution by Lappin and Leass (1994).

A great variety of systems have been proposed for coreference resolution. Several of them illustrate knowledge-poor approaches, such as Harabagiu and Maiorano (1999), following pioneering work on pronoun resolution by Mitkov (1996, 1998). Kennedy and Boguraev (1996) enhanced their algorithm for pronoun resolution by solving elementary coreference relations between potential antecedents (such as a company's full name and its acronym). We retained from these approaches the emphasis on evaluation, but we tried here to single out and organize our knowledge sources following a more theoretical approach to coreference resolution.

The Message Understanding Conferences (MUC-6 1995; MUC-7 1998) have proposed a series of competitive tasks aimed at information extraction from short articles in English into predefined templates; coreference resolution between 'markables' (REs) was evaluated as an intermediate phase. Various coreference techniques were used by the participating systems. For instance, LaSIE (Gaizauskas *et al.* 1995; Humphreys *et al.* 1998) constructed a simple semantic representation of the discourse entities using substantial domain-dependent knowledge (Gaizauskas and Humphreys 1997). Such an approach does not seem easy to extend to unrestricted texts, since in-depth semantic knowledge is not readily available yet.

The CAMP system presented at MUC-6 and MUC-7 (Baldwin *et al.* 1995; Baldwin *et al.* 1998) relied for coreference resolution on Baldwin's (1997) CogNIAC (initially developed for pronoun resolution), together with other specialized modules (proper names, pleonastic 'it', etc.). The system was tuned for high precision and made use of an original mechanism for DE activation, but its official results were globally in the same range as for other systems. Another interesting system, PIE (Lin 1995), used constraint solving on sets of coreference links to construct sets of coreferent REs. The method is appealing, since it makes use of all possible coreference information extracted from the comparison of REs, but does not seem to work incrementally (RE by RE in their order of appearance), thus rendering the use of activation cues impossible.

Turning towards more empirical approaches, RESOLVE (McCarthy and Lehnert 1995), used in the UMass system for MUC (Fisher *et al.* 1995), relied on pattern matching between potentially coreferent phrases; the criteria to ascertain or to rule out coreference were learned using decision trees. Several other techniques, not present at MUC, used machine learning for coreference resolution, namely: for pronominal anaphora, Bayesian models (Ge, Hale and Charniak 1998), co-occurrence statistics (Dagan and Itai 1990), or decision trees (Aone and Bennett 1996); for nominal coreference, clustering (Cardie and Wagstaff 1999), or decision trees (Vieira and Poesio 2000; Soon, Ng and Lim 2001). The method proposed by Soon, Ng and Lim (2001) defined twelve features for coreference, and used them in a decision tree built through machine learning. The decision tree classifier was used in the following coreference algorithm: for each RE, scan backwards for an antecedent, and stop at the first one that is positive for the classifier. Therefore, unlike our algorithm, the procedure does not consider discourse entities, but attempts to build coreference links from which the DEs can be *a posteriori* derived. It would be interesting to combine this decision tree with an approach such as Lin's (1995), to directly build reliable DEs.

We have outlined at the end of section 4 our position on machine learning techniques. First, the proportion of learning, in the previous examples, is quite small compared to the linguistic analysis that is necessary to define the features that are learned, as some authors also acknowledge: Cardie and Wagstaff (1999), note 4, or Vieira and Poesio (2000), section 6.5. In our work, learning is almost never the only factor that guides the choice of a system's behavior against another one. Learning or optimization is used here in parallel to conceptual analysis, quite often to corroborate hypotheses about the factors that are relevant to coreference resolution.

11 Conclusion

This paper shows how the distinction of various knowledge sources in a reference resolution program enhances its performances thanks to evaluation-driven design. Our main point is not the algorithm *per se*, but the design technique, which grounds in numeric evaluation the choice of the best behaviors of the KSs. The design technique is a mixture of optimization (somewhat close to machine learning) and of knowledge-based analysis. These two factors are complementary, and in general, their agreement is always sought for. The results emphasize the importance of the semantic compatibility rule, and that of an activation mechanism, over the creation/attachment rules, i.e. over a direct link from reference to linguistic expression. Further work should thus be aimed at a more elaborate yet robust semantic representation of discourse entities.

Acknowledgments

The author would like to thank Anne Reboul, Isabelle Robba and Gérard Sabah for their kind support; the three anonymous reviewers for their valuable and detailed comments; and Sandrine Zufferey for final remarks on the article. This work was initiated at LIMSI (Orsay, France), and was funded by the French Ministry for Research through the graduate program, then the post-graduate 'Lavoisier' program.

A Appendix: definitions of the evaluation methods

This appendix provides details on the measures used for the evaluation of reference resolution (Popescu-Belis 2000). Inspiration from human communication is limited, since the evaluation of reference understanding by a human receiver happens infrequently, and in most cases erroneous references are detected and corrected through dialog. The program's task is the construction of the correct DEs from a given discourse, that is, the correct sets of coreferent REs. As the sets are built during the completion of the task itself, there is no predefined correct DE that each RE should be attached to.

We limit here the task to the identification of identity coreference relations between REs, and do not include the identification of REs (called 'markables' in MUC-6 and MUC-7). We believe that RE identification, though an unavoidable step in text processing, is a separate task that uses a different kind of knowledge - e.g., spotting impersonal pronouns, idiomatic expressions, etc.

Evaluation consists in either comparing the coreference links, or comparing the two partitions of the total RE set which are derived from the DEs, or comparing the average correlation between the sender DE and the receiver DE activated by each RE. An intuitive view of recall and precision is the following: suppose that the response contains correct and wrong links, and misses some correct links. Recall is the proportion of correct links in the response with respect to all correct links in the key, whereas precision is the proportion of correct links in the response with respect to all the links in the response. This is, however, only an intuitive view: as many link configurations can correspond in fact to the same DEs, formal definitions must abstract from the particular links, as proposed in Popescu-Belis (2000).

MUC measure, $\mathcal{M}$ – MUC-6 and MUC-7 campaigns used an algorithm proposed by Vilain *et al.* (1995).This method counts links depending only on the sets of coreferent REs, not on the particular links that constitute them. The count is indulgent, as it computes, by definition, the minimal number of missing and wrong links.

B^3 measure, $\mathcal{B}$ – Bagga and Baldwin (1998), aware of the indulgence of the MUC algorithm, defined recall and precision per RE, then averaged these values to obtain global scores. The scores are lower than the MUC scores when many REs are unduly grouped, but we proved that they are always well above 0%.

Kappa measure, κ – Passonneau (1997) used the κ factor (Krippendorff 1980) to measure the agreement between two annotators of a given text, based on the probability of agreement by chance. For the distance between key and response, κ is especially relevant when these are very close. The score is computed using MUC recall and precision, and although it is less indulgent, it also conveys less information than the MUC couple. However, Carletta (1996) strongly advocates the use of the κ statistic against more *ad-hoc* measures of similarity between human annotators of discourse phenomena.

We proposed also two new measures that attempt to overcome certain limitations of the previous ones, based on new theoretical grounds and using the idea of DE (Popescu-Belis 1998; Popescu-Belis 2000).

Core-DE measure, $\mathcal{C}$ – the notion of 'core-DE' tries to grasp the program's view of each correct DE. For each key DE, an associated core-DE in the response is computed. Then, all the REs in the key DE that do not belong to the corresponding core-DE count as recall errors. Precision is computed symmetrically. We have proved that for every response, its $\mathcal{C}$-score is lower than its $\mathcal{M}$-score.

Mutual information measure, $\mathcal{H}$ – applying the model of a communication channel (Ash 1965) to reference transmission, we view the proper understanding of reference as follows: every time a given DE_i is activated in the sender's mind, the same DE_j is activated in the receiver's mind. Intuitively, once DE_1 and DE_a have been activated simultaneously (sender vs. receiver), a further co-activation of DE_1 and of DE_b ($DE_b \neq DE_a$) represents an $\mathcal{H}$-recall error, while a further co-activation of DE_2 ($DE_2 \neq DE_1$) and of DE_a represents an $\mathcal{H}$-precision error. The scores are computed using the entropy and the conditional entropy of the RE distributions in the key and the response. The $\mathcal{H}$ measure possesses a sound theoretical base, but seems often more indulgent than the MUC measure.

References

Alshawi, H. (1987) *Memory and Context For Language Interpretation.* Cambridge University Press.

Aone, C. and Bennett, S. W. (1996) Applying machine learning to anaphora resolution. In: Wermter, S., Riloff, E. and Scheler, G. (eds.), *Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing*, pp. 302–314. Springer-Verlag.

Appelt, D. (1985) Planning English referring expressions. *Artificial Intelligence* **26**(1): 1–33.

Appelt, D. and Kronfeld, A. (1987) A Computational model of referring. *Proceedings IJCAI'87*, vol. 2, pp. 640–647. Milan, Italy.

Ariel, M. (1990) *Accessing Noun-Phrase Antecedents.* Routlege.

Ash, R. B. (1965) *Information Theory.* Interscience.

Azzam, S., Humphreys, K. and Gaizauskas, R. (1998) Evaluating a focus-based approach to anaphora resolution. *Proceedings COLING-ACL'98*, vol. 1, pp. 74–78. Montreal, Canada.

Bagga, A. and Baldwin, B. (1998) Algorithms for scoring coreference chains. In *Proceedings LREC'98 Workshop on Linguistic Coreference.* Granada, Spain.

Baldwin, B. (1997) CogNIAC: high precision coreference with limited knowledge and linguistic resources. *Proceedings ACL'97 Workshop on Operational Factors in Practical, Robust, Anaphora Resolution*, pp. 38–45. Madrid, Spain.

Baldwin, B., Morton, T., Bagga, A., Baldridge, J., Chandraseker, R., Dimitriadis, A., Snyder, K. and Wolska, M. (1998) Description of the UPENN CAMP system as used for coreference. *Proceedings MUC-7.* Washington, DC.

Baldwin, B., Reynar, J., Collins, M., Eisner, J., Ratnaparkhi, A., Rosenzweig, J. and Srinivas, A. S. (1995) University of Pennsylvania: description of the University of Pennsylvania system used for MUC-6. *Proceedings MUC-6*, pp. 177–191. Morgan Kaufmann.

Bruneseaux, F. and Romary, L. (1997) Codage des références et coréférences dans les dialogues homme-machine. *Proceedings ACH-ALLC'97.* Kingston, Canada.

Cardie, C. and Wagstaff, K. (1999) Noun phrase coreference as clustering. *Proceedings Joint SIGDAT EMNLP and VLC*, pp. 82–89. University of Maryland.

Carletta, J. (1996) Assessing agreement on classification tasks: the kappa statistic. *Computational Linguistics* **22**(2): 249–254.

Chomsky, N. (1981) *Lectures on Government and Binding.* Foris.

Cristea, D., Postolache, O.-D., Dima, G.-E. and Barbu, C. (2002) AR-Engine – a framework for unrestricted coreference resolution. *Proceedings LREC 2002*, pp. 2000–2007. Las Palmas de Gran Canaria, Spain.

Cunningham, H., Humphreys, K., Gaizauskas, R. and Wilks, Y. (1997) Software infrastructure for NLP. *Proceedings ANLP'97.* Washington, DC.

Dagan, I. and Itai, A. (1990) Automatic processing of large corpora for the resolution of anaphora references. *Proceedings COLING'90*, vol. 3, pp. 330–332. Helsinki, Finland.

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

Day, D., Aberdeen, J., Hirschman, L., Kozierok, R., Robinson, P. and Vilain, M. (1997) Mixed-initiative development of language processing systems. *Proceedings ANLP'97.* Washington, DC.

Devitt, M. and Sterelny, K. (1999) *Language and Reality: an Introduction to the Philosophy of Language.* MIT Press.

Fisher, D., Soderland, S., McCarthy, J., Feng, F. and Lehnert, W. (1995) Description of the UMass system as used for MUC-6. *Proceedings MUC-6*, pp. 127–140. Morgan Kaufmann.

Gaizauskas, R. and Humphreys, K. (1997) Using a semantic network for information extraction. *Natural Lang. Eng.* **3**(2–3): 147–169.

Gaizauskas, R., Wakao, T., Humphreys, K., Cunningham, H. and Wilks, Y. (1995) University of Sheffield: description of the LaSIE system as used for MUC-6. *Proceedings MUC-6*, pp. 207–220. Morgan Kaufmann.

Ge, N., Hale, J. and Charniak, E. (1998) A statistical approach to anaphora resolution. *Proceedings Workshop on Very Large Corpora*, pp. 161–170. Montreal, Canada.

Grosz, Ba. (1981) Focusing and description in natural language dialogues. In: Joshi, A. K., Sag, I. A. and Webber, B. L. (eds.), *Elements of Discourse Understanding*, pp. 84–105. MIT Press.

Grosz, B., Joshi, A. and Weinstein, S. (1995) Centering: a framework for modeling the local coherence of discourse. *Computational Linguistics* **21**(2): 203–225.

Grosz, B. and Sidner, C. (1986) Attentions, intentions and the structure of discourse. *Computational Linguistics* **12**(3): 175–204.

Harabagiu, S. and Maiorano, S. (1999) Knowledge-lean coreference resolution and its relation to textual cohesion and coherence. *Proceedings ACL'99 Workshop on the relation of discourse structure and reference*, pp. 29–38. University of Maryland.

Hirschman, L. (1997) *MUC-7 Coreference Task Definition.* The MITRE Corporation.

Humphreys, K., Gaizauskas, R., Azzam, S., Huyck, C., Mitchell, B., Cunningham, H. and Wilks, Y. (1998) University of Sheffield: description of the LaSIE-II system as used for MUC-7. *Proceedings MUC-7.* Washington, DC.

Jackendoff, R. (2002) *Foundations of Language.* Oxford University Press.

Kamp, H. and Reyle, U. (1993) *From Discourse to Logic: Introduction to Model-Theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory.* Kluwer Academic.

Karttunen, L. (1976) Discourse referents. In: McCawley, J. D. (ed.), *Syntax and Semantics 7*, pp. 363–385. Academic Press.

Kennedy, C. and Boguraev, B. (1996) Anaphora in a wider context: tracking discourse referents. *Proceedings ECAI'96*, pp. 582–586. Budapest, Hungary.

Krippendorff, K. (1980) *Content Analysis: An Introduction to Its Methodology.* Sage Publications.

Lappin, S. and Leass, H. J. (1994) An Algorithm for pronominal anaphora resolution. *Computational Linguistics* **20**(4): 535–561.

Lasnik, H. (1989) *Essays on Anaphora.* Kluwer.

Lewin, I., Bouillon, P., Lehmann, S., Milward, D. and Tanguy, L. (1999) Discourse data in DiET. *Proceedings EACL'99 Workshop on Linguistically Interpreted Corpora.* Bergen, Sweden.

Lin, D. (1995) University of Manitoba: description of the PIE system used for MUC-6. *Proceedings MUC-6*, pp. 113–126. Morgan Kaufmann.

Luperfoy, S. (1992) The representation of multimodal user interface dialogues using discourse pegs. *Proceedings ACL'92*, pp. 22–31. Delaware, Newark.

McCarthy, J. F. and Lehnert, W. G. (1995) Using decision trees for coreference resolution. *Proceedings IJCAI'95*, pp. 1050–1055. Montreal, Canada.

Mitkov, R. (1996) Pronoun resolution: the practical alternative. *Proceedings DAARC 1996.* Lancaster, UK. (Also appeared in S. Botley and A. McEnery (eds.), *Corpus-based and computational approaches to discourse anaphora*, 2000, pp. 189–212. John Benjamins.)

Mitkov, R. (1998) Robust pronoun resolution with limited knowledge. *Proceedings COLING-ACL'98*, vol. 2, pp. 869–875. Montreal, Canada.

Mitkov, R. (2001) Towards a more consistent evaluation and comprehensive evaluation of anaphora resolution algorithms and systems. *Appl. Artif. Intell.* **15**(3): 253–276.

Mitkov, R. (2002) *Anaphora Resolution.* Studies in Language and Linguistics. Longman.

Mitkov, R., Evans, R., Orăşan, C., Barbu, C., Jones, L. and Sotirova, V. (2000) Coreference and anaphora: developing annotating tools, annotated resources and annotation strategies. *Proceedings DAARC 2000*, pp. 49–58. Lancaster, UK.

MUC-6 (1995) *Proceedings Sixth Message Understanding Conference.* Morgan Kaufmann.

MUC-7 (1998) *Proceedings Seventh Message Understanding Conference.* Available from `http://www.itl.nist.gov/iad/894.02/related_projects/muc/`.

Passonneau, R. J. (1997) Applying reliability metrics to co-reference annotation. Technical Report CUCS-017-97, Department of Computer Science, Columbia University.

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

Popescu-Belis, A. (1998) How corpora with annotated coreference links improve anaphora and reference resolution. *Proceedings LREC'98*, vol. 1, pp. 567–572. Granada, Spain.

Popescu-Belis, A. (1999) Modélisation multi-agent des échanges langagiers: application au problème de la référence et à son évaluation. Doctoral Dissertation, Departement of Computer Science, University of Paris XI / LIMSI-CNRS. (Available from `http://andreipb.free.fr/these.en.html`.)

Popescu-Belis, A. (2000) Évaluation numérique de la résolution de la référence critiques et propositions. *T.A.L. (Traitement automatique des langues)* **40**(2): 117–146.

Popescu-Belis, A., Robba, I. and Sabah, G. (1998) Reference resolution beyond coreference: a conceptual frame and its application. *Proceedings COLING-ACL'98*, vol. 2, pp. 1046–1052. Montreal, Canada.

Reboul, A. (1994) L'anaphore pronominale: le problème de l'attribution des référents. In: Moeschler, J., Reboul, A., Luscher, J.-M. and Jayez, J. (eds.), *Langage et Pertinence*, pp. 105–173. Presses Universitaires de Nancy.

Salmon-Alt, S. (2001) Entre corpus et théorie: l'annotation (co-)référentielle. *T.A.L. (Traitement automatique des langues)* **42**(2): 459–486.

Salton, G. and McGill, M. J. (1983) *Introduction to Modern Information Retrieval*. McGraw-Hill.

Sidner, C. L. (1983) Focusing in the comprehension of definite anaphora. In: Brady, M. and Berwick, R. (eds.), *Computational Models of Discourse*, pp. 267–330. MIT Press.

Soon, W. M., Ng, H. T. and Lim, D. C. Y. (2001) A Machine learning approach to coreference resolution of noun phrases. *Computational Linguistics* **27**(4): 521–544.

Tutin, A., Trouilleux, F., Clouzot, C., Gaussier, E., Zaenen, A., Rayot, S. and Antoniadis, G. (2000) Annotating a large corpus with anaphoric links. *Proceedings DAARC 2000*, pp. 28–38. Lancaster, UK.

VanDeemter, K. and Kibble, R. (2000) On Coreferring: coreference in MUC and related annotation schemes. *Computational Linguistics* **26**(4): 629–637.

Vapillon, J., Briffault, X., Sabah, G. and Chibout, K. (1997) An object-oriented linguistic engineering environment using LFG (Lexical Functional Grammar) and CG (Conceptual Graphs). *Proceedings ACL'97 Workshop on Computational Environments for Grammar Development and Linguistic Engineering*. Madrid, Spain.

Vieira, R. and Poesio, M. (2000) An Empirically based system for processing definite descriptions. *Computational Linguistics* **26**(4): 629–637.

Vilain, M., Burger, J., Aberdeen, J., Connolly, D. and Hirschman, L. (1995) A Model-Theoretic coreference scoring scheme *Proceedings of MUC-6*, pp. 45–52. Morgan Kaufmann.